

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

GOOGLE LLC,
Petitioner

v.

SANDPIPER CDN, LLC,
Patent Owner

Case IPR2025-00806
U.S. Patent No. 8,645,517

PATENT OWNER'S RESPONSE

Mail Stop "PATENT BOARD"
Patent Trial and Appeal Board
U.S. Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	PERSON OF ORDINARY SKILL IN THE ART (POSA)	2
III.	CLAIM CONSTRUCTION	2
IV.	OVERVIEW OF THE '517 PATENT	3
V.	PETITIONER'S PRIORITY CHALLENGE	8
VI.	FOUNDATIONS 1 AND 2	8
A.	Hasan is non-analogous to the '517 patent	9
1.	Hasan is not from the same field of endeavor as the '517 patent	9
2.	Hasan is not reasonably pertinent to the problem the '517 patent solves	14
B.	Petitioner's failure to identify any "policy" in Beaumont dooms Foundations 1 and 2	17
1.	The claimed policy is a rule.	17
2.	Petitioner fails to identify the claimed policy in either Beaumont or Hasan.....	19
3.	Beaumont-Hasan does not teach using a GUI to set or obtain policies for direction of network traffic.....	22
4.	Beaumont-Hasan does not teach providing said at least one policy to the name server network.	26
C.	Beaumont-Hasan does not teach the claimed GUI that uses a decision tree.....	27
1.	Beaumont-Hasan's GUI does not support the setting of policies using a decision tree representing rules.....	28
a.	Neither reference has any teaching or suggestion of a decision-tree GUI.	28
b.	The combination of Beaumont and Hasan does not teach or suggest the claimed GUI or decision tree.....	31
2.	Beaumont-Hassan's GUI does not support the setting of policies using a decision tree that includes branch nodes specifying decision criteria	33

a.	Neither Beaumont nor Hasan teaches or suggests a decision tree that includes branch nodes specifying decision criteria.....	33
b.	The combination of Beaumont and Hasan does not teach or suggest a decision tree that includes branch nodes specifying decision criteria.....	36
D.	A POSA would not have been motivated to combine Beaumont and Hasan	39
1.	A POSA would not have combined Beaumont with Hasan.....	39
2.	A POSA would not have combined Beaumont and Hasan as proposed by Petitioner	41
VII.	GROUND 3A AND 3B.....	44
A.	Overview of Ben-Shaul.....	44
1.	Ben-Shaul’s Edge Servers.....	45
2.	Ben-Shaul’s DNS.....	48
3.	Coordination of Ben-Shaul’s Edge Server and Edge DNS	52
B.	The cited prior art fails to teach or suggest the claimed decision tree	52
1.	The cited prior art fails to teach or suggest a decision tree that includes resource nodes specifying answers to be provided in response to a DNS request.....	53
2.	The cited prior art fails to teach or suggest a decision tree that includes branch nodes specifying decision criteria to be applied	59
3.	The cited prior art fails to teach or suggest a decision tree at all.....	62
C.	The cited prior art fails to teach or suggest providing said at least one policy to the name server network	65
D.	Ben-Shaul does not disclose a domain name server adapted to provide adaptive policy-based domain name service	65
E.	A POSA would not have combined Ben-Shaul and Hasan	67
VIII.	CONCLUSION.....	70

TABLE OF AUTHORITIES

Cases

<i>In re Bigio</i> , 381 F.3d 1320 (Fed. Cir. 2004)	9
<i>Sanofi-Aventis Deutschland GmbH v. Mylan Pharms. Inc.</i> , 66 F.4th 1373 (Fed. Cir. 2023)	9
<i>Sisvel Intern. S.A. v. Sierra Wireless, Inc.</i> , 81 F.4th 1231 (Fed. Cir. 2023)	2
<i>Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.</i> , 200 F.3d 795 (Fed. Cir. 1999)	3

Statutes

35 U.S.C. § 102	8, 16
-----------------------	-------

PATENT OWNER'S UPDATED EXHIBIT LIST

Exhibit No.	Description
2001	“Cloud CDN: content delivery network,” <i>Google Cloud</i> , accessed at https://cloud.google.com/cdn (last accessed July 2, 2025)
2002	Prosecution History of U.S. Application No. 10/259,497
2003	Prosecution History of U.S. Application No. 11/932,162
2004	<i>Sandpiper CDN, LLC v. Google LLC</i> , 2:24-cv-03951, ECF No. 1 (C.D. Cal. May 10, 2024)
2005	<i>Sandpiper CDN, LLC v. Google LLC</i> , No. 2:24-cv-03951, Invalidity Contentions (April 22, 2025)
2006	<i>Sandpiper CDN, LLC v. Google LLC</i> , No. 2:24-cv-03951, ECF No. 82 (C.D. Cal. June 24, 2025)
2007	<i>Sandpiper</i> , No. 2:24-cv-03951, ECF No. 28 (C.D. Cal. Sept. 16, 2024)
2008	FAQs for Interim Processes for PTAB Workload Management, accessed at https://www.uspto.gov/patents/ptab/faqs/interim-processes-workload-management
2009	<i>Gestures Tech. Partners LLC v. Apple Inc. et al.</i> , Petition for Writ of Certiorari, Case. No. 24- (Filed June 11, 2025)
2010	<i>Intentionally Left Blank</i>
2011	<i>Intentionally Left Blank</i>
2012	<i>Intentionally Left Blank</i>
2013	Interim Processes for PTAB Workload Management Memorandum (March 26, 2025)
2014	National Judicial Caseload Profile – United States District Courts (March 2025)
2015	Declaration of Dr. Prashant Shenoy

Exhibit No.	Description
2016	Curriculum Vitae of Dr. Prashant Shenoy
2017	Declaration of Dr. Prashant Shenoy in Support of Patent Owner's Response
2018	Deposition Transcript of Dr. Bill Lin (January 26, 2026)

I. INTRODUCTION

U.S. Patent No. 8,645,517 (the '517 patent) provides novel techniques for creating domain name service (DNS) policies with a graphical user interface (GUI) that uses a specifically arranged “decision tree” that includes “nodes” specifying decision criteria and answers to be provided in response to DNS requests.

Petitioner’s obviousness grounds challenging independent claims 1, 9, and 17 fail to teach any GUI for specifying DNS policies, let alone the specifically claimed one.

In Grounds 1 and 2, Petitioner asserts a “combined GUI” allegedly taught by Beaumont in view of Hasan. But neither of these references teaches a GUI for specifying DNS policies. Beaumont does not teach a GUI at all. Hasan is non-analogous art because it is completely unrelated to DNS, and its GUI thus does not provide DNS policies. Thus, the combination of Beaumont and Hasan would not teach a GUI that enables specification of DNS policies, nor would it include nodes specifying decision criteria and answers to be provided in response to DNS requests.

In Grounds 3A and 3B, Petitioner asserts two GUIs taught by Ben-Shaul that provide a “hierarchical map” of edge servers in a content delivery network. But Ben-Shaul’s edge servers do not provide DNS services and its GUIs are not used to write DNS policies—they are only used to write edge server policies that control

operation of the edge servers. Ben-Shaul's GUIs are not used to write policies for its EdgeDNS server, which implements DNS in Ben-Shaul's system, nor are they used to write any policies for responding to DNS requests. Petitioner also asserts a combined GUI allegedly taught by the combination of Ben-Shaul and Hasan, but this combination fails for the same reasons. Again, neither reference teaches a GUI for writing DNS policies, and their combination does not either.

The Board should uphold the patentability of all challenged claims for the reasons provided below. For convenience, the below arguments address elements of independent claim 1. But independent claims 1, 9, and 17 are substantially similar and all arguments herein apply equally to each claim.

II. PERSON OF ORDINARY SKILL IN THE ART (POSA)

For the purposes of this proceeding only, Patent Owner agrees with Petitioner's proposed definition of a POSA.

III. CLAIM CONSTRUCTION

Absent lexicography or disavowal, claim terms in an IPR are given their plain and ordinary meaning in light of the specification and prosecution history. *Sisvel Intern. S.A. v. Sierra Wireless, Inc.*, 81 F.4th 1231, 1236 (Fed. Cir. 2023). In the arguments below, Patent Owner discusses the plain and ordinary meanings of the terms "policy" and "decision tree" as would have been understood by a POSA in the context of the '517 patent. Beyond honoring these plain and ordinary

meanings, no term requires express construction. *See Vivid Techs., Inc. v. Am. Sci. & Eng'g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999).

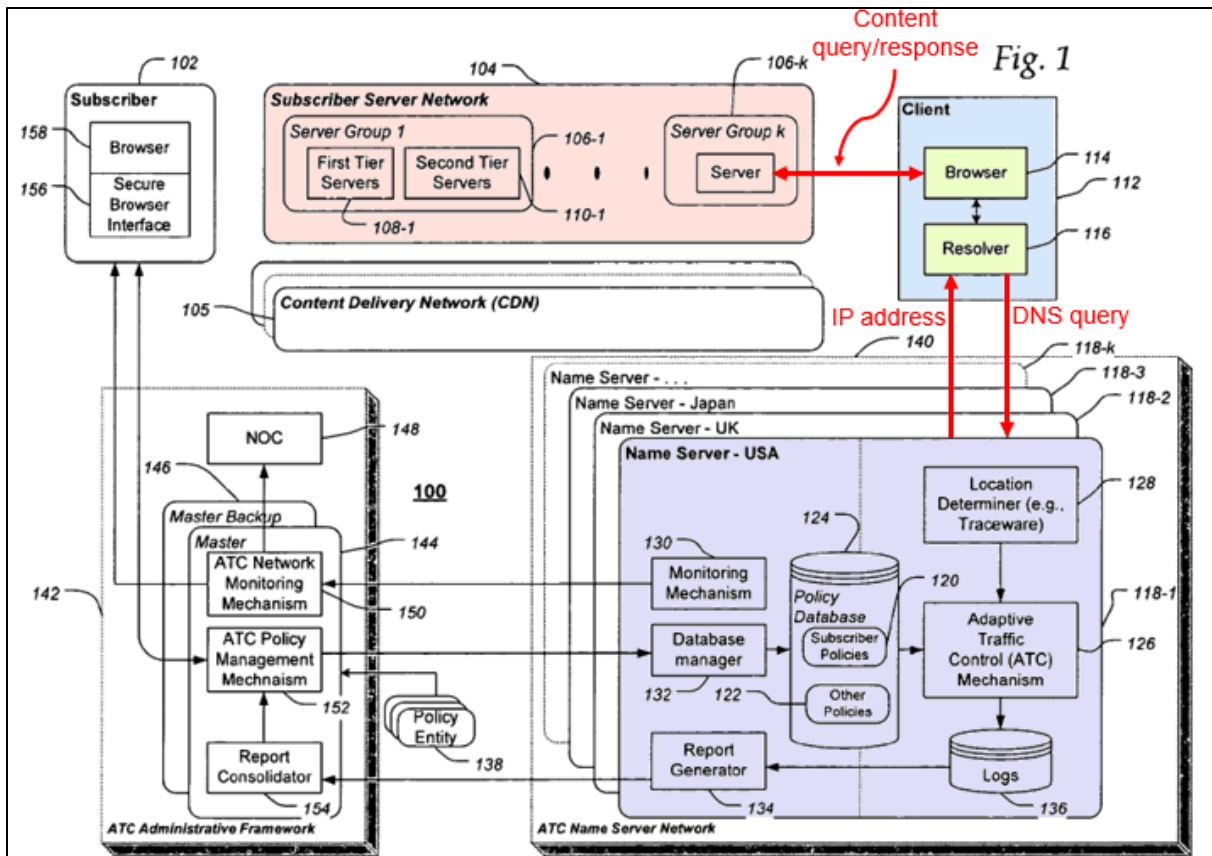
IV. OVERVIEW OF THE '517 PATENT

The '517 patent provides solutions for “configurable, adaptive, global traffic control and management in networks such as the Internet.” EX1001, 1:36-39. Specifically, the '517 patent is directed to “a framework wherein resources of a content provider may be delivered to clients from different domains.” EX1001, Abstract. The specification and claims describe “a scenario in which DNS service is provided... on behalf of a content or applications provider.” EX1001, 3:16-20, claim 1 (reciting a “name server network comprising at least one domain name server” that provides “adaptive policy-based domain name service” and “specif[ies] one or more answers to be provided in response to a Domain Name Service (DNS) request”).

The '517 patent describes the process for delivering network (e.g., internet) data to end users with respect to Figure 1. First, a user at a client device 12 “enters a [URL] into a browser 114 on client 112.” EX1001, 3:60-62. In response, a DNS resolver 116 on the client 112 “queries a particular name server (e.g., name server 118-1)” to determine the IP address that corresponds to the domain associated with that URL. EX1001, 4:35-44. The name server determines and “returns [] the IP

(Internet Protocol) address... of a server... in the subscriber server network” that hosts the requested content associated with the domain. EX1001, 4:35-44.

As the ’517 patent explains, multiple servers may host the requested content. *See* EX1001, 4:39-42, 4:60-67. To address this, the ’517 patent describes and claims policies for determining which of these servers will receive the network requests. For instance, “[t]he determination of the particular IP address returned to the resolver 116 may be based on a number of factors, including the resolver’s location (e.g., as determined from the resolver’s IP address) and various policies (e.g., subscriber policies 120, other policies 122) in a policy database 124.” EX1001, 4:44-49. When the browser receives the returned IP address, it is able to direct its content query to the server having that address “in order to obtain the desired resource.” EX1001, 4:49-51.



EX1001, FIG. 1 (annotated).

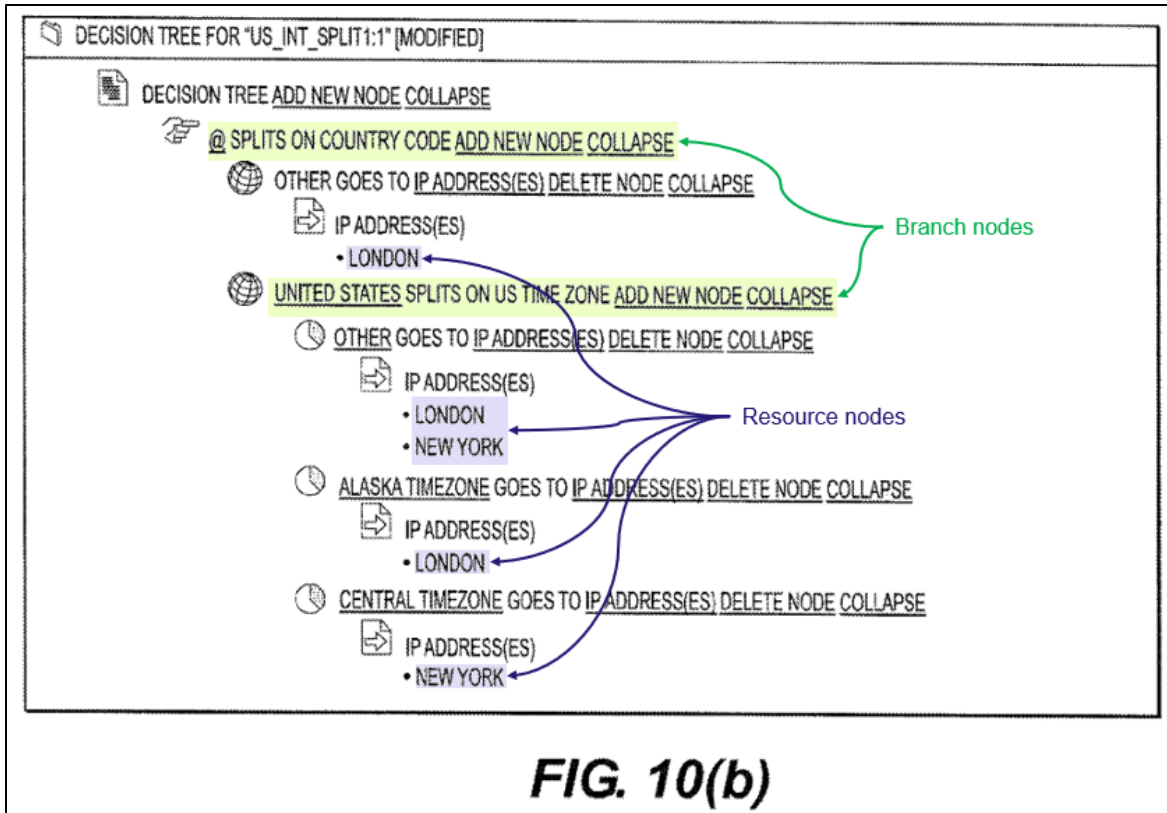
More particularly, the '517 patent discloses policies that are applied in response to receiving DNS requests in order to resolve the requests. It also discloses a graphical user interface (GUI) for setting these policies. The policies “may be hierarchically constructed to form a decision tree,” also referred to as a “policy hierarchy.” EX1001, 8:55-56. The decision tree comprises “rules that the user (content provider/subscriber) wants to apply” in resolving DNS requests. EX1001, 9:1-2. The “leaves” or “resource nodes” of the decision tree “specif[y] the answers (IP addresses or CNAMEs) that the user wants [] to provide in response to a DNS request.” EX1001, 9:3-6. The “branches” or “branch nodes”

“specif[y] the various decision criteria that the user wants to apply.” EX1001, 9:6-

8. Thus, the decision tree represents a policy that is applied in response to DNS requests.

Figure 10(b) “shows an exemplary decision tree embedded in a set of geographical policies that guide how the traffic should be directed to a set of servers.” EX1001, 25:8-10. As shown, Figure 10(b) depicts a GUI that represents a decision tree for setting DNS policies. The GUI enables users to specify rules, including decision criteria (e.g., “Split on Country”) for responding to requests in conjunction with corresponding answers for responding to requests (e.g., London, New York servers). EX1001, 25:8-25. A DNS server applies the policy represented by the decision tree in response to receiving a DNS request from an end-user.

This decision tree includes a branch node that specifies decision criteria for splitting “between the United States and the rest of the world.” EX1001, 25:10-13. That is, upon receiving the DNS request, the DNS server determines whether the requester is located in the United States, and if so, applies one set of policies, and if not, applies another set of policies (i.e., “Splits on Country”). The decision tree includes another branch node specifying decision criteria for splitting based on time zone in the United States. EX1001, 25:16-20. That is, if the requestor is located in the United States, the DNS server further determines the requestor’s time zone and applies different policies accordingly.



EX1001, FIG. 10(b).

The decision tree also includes “resource nodes” specifying answers to be provided in response to the DNS request. The resource nodes include IP addresses of servers named “London” and “New York.” When the DNS server reaches the leaves of the tree, after all decision criteria have been applied, it selects the IP address of the server represented by that resource node. EX1001, 25:14-25. For instance, if the requestor was located in the United States in the Central Time Zone, the DNS server would return the IP address of the New York server, but if the requestor was located in France, the DNS server would return the IP address for the London server. EX1001, 25:14-25.

V. PETITIONER'S PRIORITY CHALLENGE

The Petition first presents a priority date challenge, and then presents every prior-art reference as alternatively prior art under 35 U.S.C. § 102(b) or § 102(e) based on the success of that priority challenge. *See* Pet., 8. But because each reference asserted in the Petition was filed before the earliest possible priority date of September 28, 2001, the Board need not decide the priority date of the '517 patent to uphold the patentability of the challenged claims in this proceeding. Patent Owner does not concede that Petitioner's proffered priority date is correct, and reserves the right to demonstrate the correct priority date in other forums or proceedings.

VI. GROUNDS 1 AND 2

Petitioner's Grounds 1 and 2 rely on Beaumont in combination with Hasan. These grounds fail to render obvious any claim in the '517 patent because: (1) Petitioner fails to identify the claimed policy in the prior art, (2), Hasan does not teach the claimed GUI that uses a decision tree to support the setting of at least one policy, and (3) a POSA would not have combined Beaumont with Hasan in the first place. But most important, no POSA would consider Hasan analogous art to the '517 patent. As the '517 patent's independent claims are substantially similar, these arguments apply to all claims.

A. Hasan is non-analogous to the '517 patent

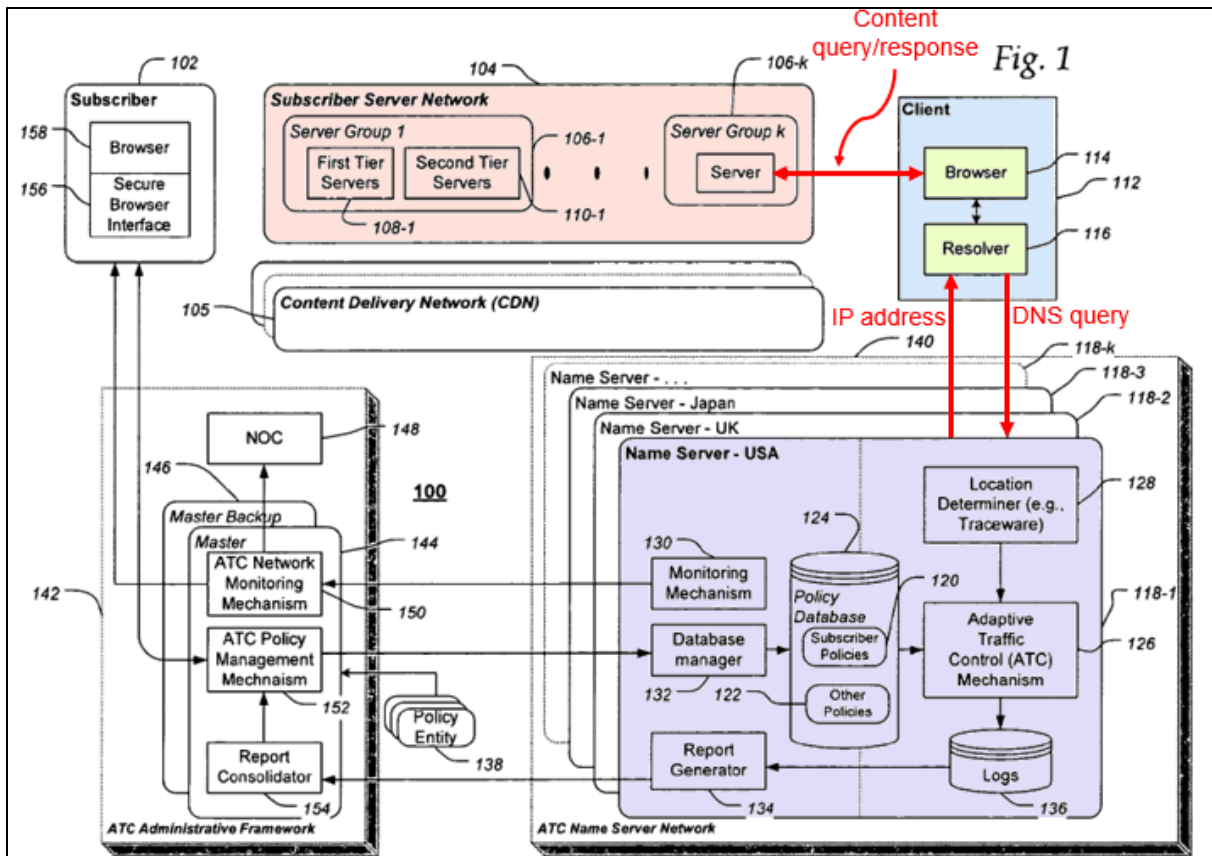
To be considered analogous art, Hasan must be (1) from the same field of endeavor as the '517 patent, or (2) if not from the same field of endeavor, reasonably pertinent to the particular problem with which the '517 patent inventors were involved. *Sanofi-Aventis Deutschland GmbH v. Mylan Pharms. Inc.*, 66 F.4th 1373, 1377 (Fed. Cir. 2023). Hasan is neither, and thus does not qualify as prior art to the '517 patent. *In re Bigio*, 381 F.3d 1320, 1325 (Fed. Cir. 2004). This failure dooms Grounds 1, 2, and 3B equally. *See* EX2017, ¶¶62-79.

1. Hasan is not from the same field of endeavor as the '517 patent

Hasan is not from the same field of endeavor as the '517 patent. EX2017, ¶¶62-72. The '517 patent is directed to “a framework wherein resources of a content provider may be delivered to clients from different domains.” EX1001, Abstract. The specification and claims describe “a scenario in which DNS service is provided... on behalf of a content or applications provider.” EX1001, 3:16-20, claim 1 (reciting a “name server network comprising at least one domain name server” that provides “adaptive policy-based domain name service” and “specif[ies] one or more answers to be provided in response to a Domain Name Service (DNS) request”). The '517 patent's field of endeavor is thus DNS services for directing network traffic. *See In re Bigio* at 1325 (The field of endeavor is determined “by reference to explanations of the invention's subject matter in the

patent application, including the embodiments, function, and structure of the claimed invention.”); EX2017, ¶62.

As explained above, the ’517 patent discloses DNS name servers that resolve DNS requests by determining the IP address of a server corresponding to the domain specified in the request. EX1001, 4:35-44. As the ’517 patent explains, multiple servers may correspond to the requested domain. *See* EX1001, 4:39-42, 4:60-67. To address this, the ’517 patent describes and claims policies for determining which of these servers will receive the network requests. For instance, “[t]he determination of the particular IP address returned to the resolver 116 may be based on a number of factors, including the resolver’s location (e.g., as determined from the resolver’s IP address) and various policies (e.g., subscriber policies 120, other policies 122) in a policy database 124.” EX1001, 4:44-49. When the browser receives the returned IP address, it is able to direct its content query to the server having that address “in order to obtain the desired resource.” EX1001, 4:49-51.



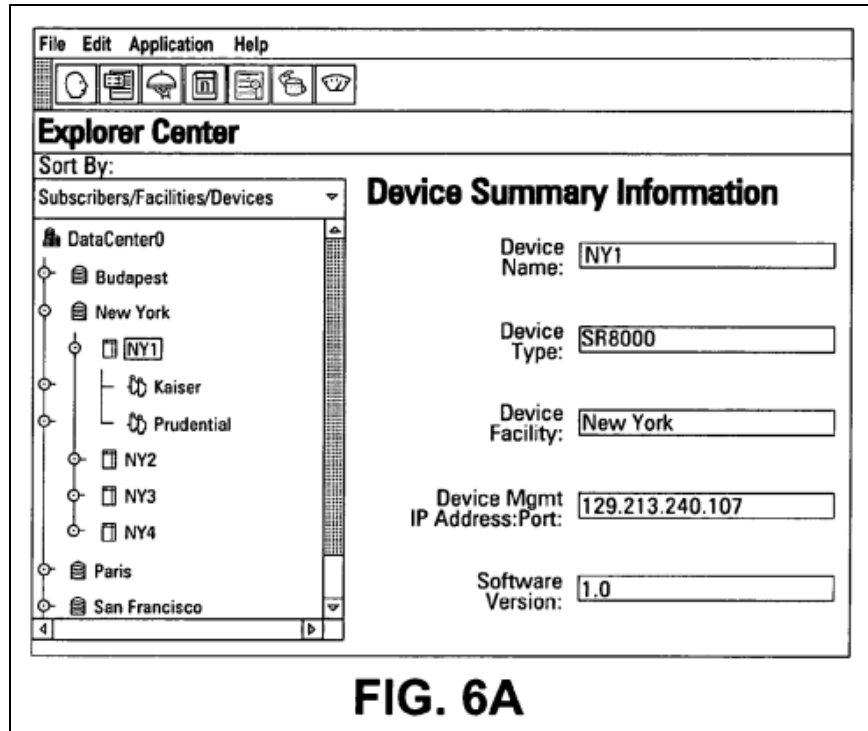
EX1001, FIG. 1 (annotated).

The '517 patent thus discloses content delivery policies for resolving DNS requests to particular servers. EX2017, ¶¶63-65. The policies implement DNS services to manage the direction of network traffic between end user client devices and servers hosting requested content. EX1001, 1:36-39.

Hasan is from a different field of endeavor. EX2017, ¶¶66. It is directed to a “virtual management system for a network facility, such as a data center, or any facility having a plurality of components which can be organized as objects for presentation in a virtualized environment.” EX1006, Abstract. It describes its field of endeavor as the “management of network services in a data center, and in

particular to management of services, subscribers, devices, log servers, and facilities using a common, virtualized management system.” EX1006, 1:5-10 (describing the “Field of the Invention”). Hasan does not disclose a DNS system at all. *See* EX2018, 97:5-98:17. It mentions DNS only one time to explain that “DNS settings” are “implemented by *other* systems.” EX1006, 17:57-63 (emphasis added); *see also* EX2017, 66.

Hasan’s virtual management system “includes a management topology” of a data center. EX1006, 3:46-50. It provides “centralized management to a data center having at least one facility and a plurality of devices configurable to provide network services to subscribers.” EX1006, 4:9-12. It allows “a single system administrator [to] configure individual devices, log servers, or services for a subscriber, the facility, or for the entire data center” via a graphical user interface (GUI). EX1006, 6:28-31. The GUI provides “a hierarchical layout of the data center,” which allows a data center administrator to edit the properties of the data center objects. EX1006, 8:19-22, 8:30-47. An example view is shown in Figure 6A.



EX1006, FIG. 6A.

The GUI of Figure 6A “allows the configuration or viewing of attributes related to a specific device within a single data center facility.” EX1006, 9:1-4. In other words, Hasan’s GUI allows administrators to view the data center layout to perform “various management capabilities” on devices or objects depicted in the view. EX1006, 8:32-47; *see also* EX2017, ¶¶67-68.

Hasan’s virtual management system enables administrators to edit data center policies, *not* DNS policies for directing network traffic. *See* EX1006, 6:62-63, FIG. 4; EX2017, ¶¶69-70. Hasan explicitly states that its system does *not* implement “DNS settings.” EX1006, 17:57-63 (“DNS settings” are “implemented by other systems”). Rather, Hasan merely discloses a number of undefined, generic policies for data center administration, such as, for instance, “administrative

policies,” “security policies,” “policy information,” “data configuration polic[ies],” “subscriber policies,” “VPN policies,” “access control policies,” and “[f]irewall policies.” EX1006, 6:55-57, 7:14-19, 7:52-55, 12:62-63, 13:7-9, 13:14-16, 14:52-55. It does not describe any specific policy; it only lists *types* of policies that *can* be implemented. EX2017, ¶¶70-71.

Hasan is directed to the administration of data center facilities. It is not concerned with DNS services or directing end-user network requests to the appropriate servers. EX2017, ¶¶69, 71. Hasan is not in the same field of endeavor as the claimed invention of the ’517 patent. EX2017, ¶72.

2. Hasan is not reasonably pertinent to the problem the ’517 patent solves

Hasan is also not reasonably pertinent to the particular problem addressed by the ’517 patent. EX2017, ¶¶73-79. The ’517 patent explains the problem it solves: “[a]s the volume of Internet traffic grows, providers of web content and applications increasingly need to deliver content from multiple servers at widely-separated locations in order to sustain a good end-user experience under high traffic loads.” EX1001, 1:42-48. In particular, one of the “difficult challenges” of providing of good end-user experience on the growing internet is “how to control the way in which requests from end-users are distributed to each server according to important content provider policy constraints imposed for economic, contractual or other reasons.” EX1001, 1:52-55. The ’517 patent thus addresses the problem of

how to distribute network requests from end users to the appropriate servers in a manner that provides a good end-user experience. EX2017, ¶73.

As explained above, the '517 patent solves this problem by describing and claiming specific rules, or policies, for directing network requests to the appropriate servers. EX1001, 1:58-60. These policies govern the server selection process based on factors such as user location, server availability, load sharing between servers, the state of the network, etc. EX1001, 1:60-67. The policies implement “an extremely fine-grained level of network Internet traffic control to providers of Internet content and applications, enabling them to dramatically improve the end-user experience (measured by speed of request resolution, associated download time, and the availability of servers) over that provided by conventional web servers and mirrored server farms.” EX1001, 2:1-7.

Hasan, on the other hand, is not concerned with delivering traffic to end user devices. EX2017, ¶¶74-75. As explained above, it is directed to data center administration. The particular problem to which Hasan relates is “[t]he difficulty in administering a data center.” EX1006, 2:59-62. Hasan explains: “the more subscribers one has in the data center, the more resources must be committed to administration.” EX1006, 2:65-67. “This can become exceedingly difficult where changes must physically occur at the cage or via individual devices, especially in multi-facility data centers.” EX1006, 2:67-3:3.

Hasan's "virtual management system" provides solutions "to alleviate some of these management problems." EX1006, 3:22-24, 3:39-43. Hasan explains that, "[o]nce organized into the virtualized management system of the present invention, management of the data center is streamlined and less user intensive." EX1006, 3:43-45. "The users of the virtualized network management system are [] administrators," not end users. EX1006, 15:39-41. Hasan is thus concerned with helping administrators configure and manage data centers. EX2017, ¶¶76-77. It provides an "administrative interface" that enables administrators to view and configure data center objects, such as "devices, subscribers, facilities, log servers and data centers." EX1006, 3:46-56, 5:60-62. It is not concerned with delivering traffic to end users. It leaves "DNS settings" to be "implemented by other systems." EX1006, 17:57-63; *see also* EX2017, ¶78.

Thus, Hasan is not reasonably pertinent to the problem addressed by the '517 patent. EX2017, ¶79.

Hasan is not analogous art to the '517 patent, and thus does not qualify as prior art under 35 U.S.C. § 102. Accordingly, each of Petition Grounds 1, 2, and 3B fail for relying on non-prior art.

B. Petitioner’s failure to identify any “policy” in Beaumont dooms Grounds 1 and 2

Petitioner fails to identify any policy in Beaumont. EX2017, ¶¶80-104. This failure dooms Grounds 1 and 2. The Petition points solely to Beaumont’s “configurable preference table” as representing the claimed “at least one policy.” *See* Pet., 15 (Beaumont’s DNS server “directs requests (traffic) according to a configurable preference table (policies)”). But this configurable preference table is not a policy, as claimed by the ’517 patent, which must be a course, rule, or principle of action. The Petition thus fails to identify the claimed policy in Beaumont or Hasan. Consequently, the Petition also fails to identify (1) a GUI that is used to set or obtain the claimed policy for direction of network traffic, and (2) a policy that is provided to the name server network, as claimed.

1. The claimed policy is a rule.

The independent claims each recite various features involving a “policy.” *See, e.g.*, claim 1 (reciting “policy-based traffic direction”, “policy-based domain name service”, and “at least one policy”). The ’517 patent is clear that a “policy” refers to a course, rule, or principle of action. *See* EX1001, 1:43-2:7 (“This invention solves these and other problems by providing a means to distribute network (e.g., Internet) traffic according to a configurable set of rules.”); EX2015, ¶59. The Board agreed with this interpretation in its Institution Decision. Paper 14, 19 (the “understanding of a policy as a rule is consistent with the disclosure in the

'517 patent that 'a policy may be created by generating a decision tree representing the ATC rules"). Both experts also agree that the claimed policy is a rule. EX2017, ¶¶81-85; EX2018, 33:6-34:21, 146:10-20.

In the context of the claims, a policy is a course, rule, or principle of action used to direct traffic of the underlying servers. EX1001, 1:43-2:7 ("These rules together provide an extremely fine-grained level of network Internet traffic control to providers of Internet content and applications"); EX2017, ¶83. The policy or policy-based features make decisions based on the geographical locations and the capacities of subscriber network servers to manage and direct traffic. *See* EX1001, 6:33-44 ("For example, the subscriber may set up policies to direct traffic with respect to considerations such as the geographical locations and the capacities of the underlying servers in the subscriber's network"), 7:13-21 ("Decisions are based on location of the end-user or an approximate thereof, e.g., using the IP address of an end-user's resolver"), 7:22-28 ("The subscriber can explicitly specify the amount of traffic (load share) to be directed to each of their servers within a defined server set in the subscriber server network").

Importantly, the "policy" or "policy-based" features are structurally and functionally different from the "geographical locations and the capacities of the underlying servers," EX1001, 6:33-36, that the policy features use. EX2017, ¶84. In other words, the policy governs the process, while the geographical locations

and the capacities of the underlying servers are the inputs that the policy uses to determine traffic direction. *See* EX1001, 6:33-44, 15:28-35.

The '517 patent further illustrates what it means by “policy” and “policy-based” course, rule, or principle of action through several examples. *See* EX1001, 7:13-8:25. One example is a “geographic policy,” in which decisions are based on the location of the end-user or an approximate thereof, *e.g.*, using the IP address of an end-user’s resolver. *See* EX1001, 7:13-21. In this example, the policy or policy-based rule is interpreted as: “if the IP address is within the U.S., [then] the request may be directed to a ‘domestic’ group of servers in the subscriber server network 104, otherwise it may be directed to ‘international’ servers in the network.” EX1001, 7:13-21; *see also* EX2017, ¶85.

2. Petitioner fails to identify the claimed policy in either Beaumont or Hasan

Petitioner fails to identify the claimed policy in the cited references, alone or in combination. The Petition points solely to Beaumont’s “configurable preference table” as representing the claimed at least one policy. *See* Pet., 15 (Beaumont’s DNS server “directs requests (traffic) according to a configurable preference table (policies)”). This preference table is not a policy, nor does it contain policies. EX2017, ¶¶87-91. It is merely a data structure that stores data. EX1007, ¶¶23, 32. It has none of the structure of the claimed policy, which is characterized by rules, such as a conditional and an action. *See* EX1001, 8:55-56, 15:28-32, 16:5-7. Nor

does the preference table of Beaumont function like the claimed policy. EX2015, ¶64. Beaumont's preference table merely stores information, such as locations or statuses of a server, delivery server address, destination domain, estimated distance, number of times being selected per hour, health, and time-to-live. *See* EX1007, FIG. 3.

	371	372	373	374	375 (optional)	376 (optional)	377 (optional)	378 (optional)	379 (optional)
310	Delivery Server Address	Destination Domain	Preference	Number of lost packets	Number of connections	Estimated distance	number of times being selected per hour	Health	Time-To-Live (minutes)
320	152.192.16.15	182.192	1	100	1	100	100	Working	1
321	153.192.16.15	182.192	2	200	2	90	90	Down	2
322	154.192.16.15	182.192	9	900	9	10	10	Working	9
330	155.192.17.15	155.192	8	800	8	20	20	Working	8
340	156.192.17.15	156.192	1	100	1	10	10	Working	1
350	157.192.17.15	156.192	5	500	5	50	50	Working	5
360	158.192.17.15	156.192	3	300	3	70	70	Working	3

FIG. 3

EX1007, FIG. 3.

The preference table does not provide the claimed course, rule, or principle of action required for a policy. EX2017, ¶¶88-89. It does not store software or any other component for implementing such course, rule, or principle of action. It does not govern any process or behavior for directing traffic to the content delivery servers. It is a listing of information. This information could be used as input to a policy—but it is not a policy itself.

For instance, Beaumont discloses “rule[s] for selecting the best content delivery server” that are distinct from the information stored in the preference table, such as the preference numbers shown in Figure 3 above. EX1007, ¶¶31-32. To the extent Beaumont discloses policies, they are specified by these rules, not by the preference numbers, which are merely integers input by the user. EX2017, ¶¶89-90.

The rules provide instructions or regulations that govern the DNS’s behavior in selecting content delivery servers. EX2017, ¶¶89-90. For instance, one rule governs DNS behavior for “selecting [a server] based on user preferences.” EX1007, ¶31. This rule is implemented via a software-based “selecting module” that instructs the system to retrieve certain “preference numbers” from the preference table, and to “select[] an address based on the retrieved preference numbers.” EX1007, ¶32. Conversely, the preference numbers themselves do not provide or represent any logic or rule for selecting a server. EX2017, ¶¶89-91. They are merely parameters that serve as inputs to Beaumont’s rules. *See* EX1007, ¶31. Indeed, Dr. Lin acknowledged in deposition that a rule for selecting a server that “has a higher preference number” would be a “policy” and that Beaumont’s preference number itself is a “decision feature” that could be used by that policy. EX2018, 120:3-6; *see also* EX2018, 180:11-18. But the Petition points only to the preference numbers as the claimed policy.

For this reason alone, Grounds 1 and 2 fail.

3. Beaumont-Hasan does not teach using a GUI to set or obtain policies for direction of network traffic.

Petitioner's failure to identify the claimed policy dooms Grounds 1 and 2 for another reason. Neither Beaumont nor Hasan teach a GUI that is used to set or obtain the claimed policy for direction of network traffic. EX2017, ¶¶93-100. Beaumont does not teach a GUI at all. And although Hasan teaches a GUI for editing generic data center administration policies, none of its policies are for directing network traffic in response to DNS requests, as claimed. Petitioner's combination thus fails because neither reference teaches the concept of using a GUI for inputting the claimed type of policies. EX2017, ¶93.

Beaumont does not disclose a GUI for obtaining or setting policies. EX2017, ¶¶94-96. At best, Beaumont discloses an interface for manually inputting “preference numbers”¹ that are stored in its preference table. EX1007, ¶¶31, 34. But again, these preference numbers are not policies, or rules—they are merely integers. EX2017, ¶¶94-96. They are merely parameters that serve as inputs for

¹ As Petitioner acknowledges, Beaumont does not explicitly disclose an interface for these inputs. *See* Pet., 14. It thus does not teach a GUI at all, let alone one for inputting the claimed policies.

Beaumont’s “rules for selecting the best content delivery server.” *See* EX1007,

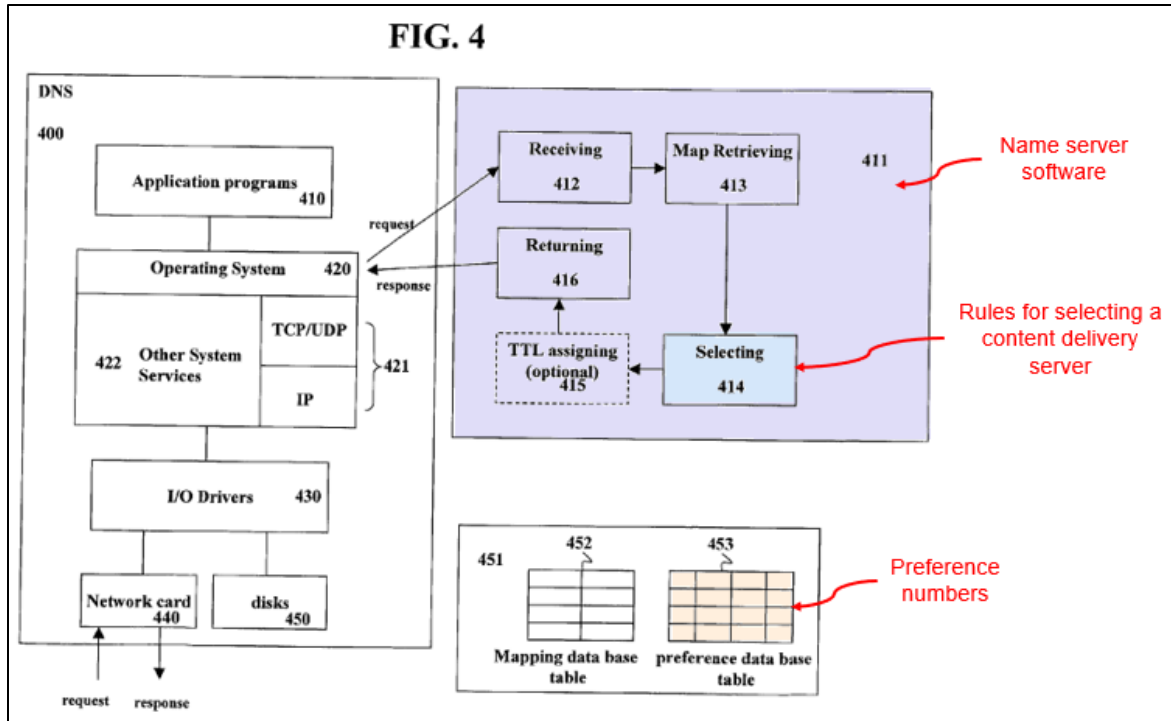
¶31. Beaumont’s rules, however, have no GUI.

	371	372	373	374	375 (optional)	376 (optional)	377 (optional)	378 (optional)	379 (optional)
							number		
310	Delivery Server Address	Destination Domain	Preference	Number of lost packets	Number of connections	Estimated distance	number of times being selected per hour	Health	Time-To- Live (minutes)
320	152.192.16.15	182.192	1	100	1	100	100	Working	1
321	153.192.16.15	182.192	2	200	2	90	90	Down	2
322	154.192.16.15	182.192	9	900	9	10	10	Working	9
330	155.192.17.15	155.192	8	800	8	20	20	Working	8
340	156.192.17.15	156.192	1	100	1	10	10	Working	1
350	157.192.17.15	156.192	5	500	5	50	50	Working	5
360	158.192.17.15	156.192	3	300	3	70	70	Working	3

FIG. 3

EX1007, FIG. 3.

As explained, to the extent Beaumont discloses policies at all, these “rules for selecting the best content delivery server,” are the policies, not the preference numbers. EX2017, ¶¶95-96. The rules are not set or obtained through a user interface. Rather, they are merely implemented as software via the “selecting module 414” that is “recorded in a secondary memory... before being loaded into the main memory.” EX1007, ¶¶32-33. Beaumont does not disclose a user interface for inputting policies. EX2017, ¶¶95-96.



EX1007, FIG. 4 (annotated).

Hasan does not cure Beaumont’s deficiencies. EX2017, ¶¶97-99. The Petition does not rely on Hasan for any disclosure of policies or policy-based features. Nor does Hasan disclose the claimed policy in any event. Although Hasan’s “policy editor” is a GUI for setting policies, it does not set or obtain policies for direction of network traffic, as claimed. EX2017, ¶97. Hasan is not concerned with DNS. It does not disclose DNS policies for directing network traffic. Rather, it merely discloses a number of undefined, generic policies for data center administration, such as, for instance, “administrative policies,” “security policies,” “policy information,” “data configuration polic[ies],” “subscriber policies,” “VPN policies,” “access control policies,” “[f]irewall policies.” EX1006, 6:55-57, 7:14-19, 7:52-55, 12:62-63, 13:7-9, 13:14-16, 14:52-55. It does

not define or otherwise elaborate on these policies. It does not describe their structure or logic. It merely lists undefined types of policies that may be helpful in implementing data center administration tasks. Thus, Hasan does not disclose a GUI for setting or obtaining the claimed policies. EX2017, ¶97.

Petitioner's problem is not merely that neither Beaumont nor Hasan alone teach the claimed GUI—the proposed combination does not teach or suggest this element. EX2017, ¶98. Petitioner's proposed combination is premised on Beaumont's manually entered "preference numbers." Pet., 13-14 ("In view of Beaumont's teaching of manually entered rule preferences [], a POSITA would have been motivated to implement Hasan's [] teaching in Beaumont"). According to Petitioner, the resulting combination allows users to "conveniently configure *preferences for Beaumont's DNS servers.*" Pet., 14. Petitioner alleges that, in the combined GUI, "selecting a server would launch a menu for assigning the preferences shown in Fig. 3 of Beaumont." Pet., 18, *see also* 20 ("Hasan's GUI would enable users to specify [Beaumont's] preferences"). But again, Beaumont's preferences are merely integers that represent user preference. EX2017, ¶99. They are separate and distinct from the underlying *rules* that apply the preferences. At best, Petitioner's combination includes a GUI for inputting these preference numbers. EX2017, ¶99. It does not include a GUI used "to obtain at least one

policy for direction of network traffic,” or that “supports the setting of said at least one policy,” as claimed.

This is consistent with Beaumont, which only teaches manual entry of user preferences, and with Hasan, which only teaches manual entry of generic data center administration policies that are unrelated to the claims. Their combination thus cannot teach more than a GUI for manually inputting preference numbers. EX2017, ¶¶99-100. Indeed, because Beaumont’s rules for selecting a content delivery server are set in software, these policies are obtained and set prior to, and completely independent from, the manual entry of Beaumont’s user preferences. *See* EX1007, ¶¶31-32; EX2017, ¶99. The combined Beaumont-Hasan system thus does not teach or suggest a GUI for setting or obtaining the specifically claimed DNS policies.

4. Beaumont-Hasan does not teach providing said at least one policy to the name server network.

Petitioner’s failure to identify the claimed policy dooms Grounds 1 and 2 for yet another reason. Neither Beaumont nor Hasan teach providing said at least one policy to the name server network, as claimed. EX2017, ¶¶102-104. The independent claims require “using said GUI to obtain at least one policy for direction of network traffic” and “providing [the same] at least one policy to the name server network.” The combination of Beaumont and Hasan does not teach these elements.

Petitioner points to Beaumont’s preference table as the alleged policy that is provided to the name server network. *See* Pet., 18. But again, this preference table is not a policy as claimed. EX2017, ¶103. Thus even to the extent the data stored in this preference table (e.g., the preference values) are provided to Beaumont’s DNS server, this does not satisfy this claim element.

To the extent Beaumont’s “rules” for selecting content delivery servers are considered policies, they do not satisfy this element either. This element requires the at least one policy that was obtained and set using the GUI to be provided to the name server network. But as explained above, Beaumont-Hasan’s rules for selecting content delivery servers are not set or obtained using a GUI. EX2017, ¶104. Beaumont-Hasan does not teach a policy that was both obtained and set using a GUI and also provided to the name server network. Beaumont-Hasan thus fails to teach “providing said at least one policy to the name server network,” as claimed.

C. Beaumont-Hasan does not teach the claimed GUI that uses a decision tree

Petitioner’s combination fails to teach or suggest the claimed GUI that “supports the setting of said at least one policy using a decision tree representing rules.” The decision tree must “represent[] rules,” and it also must comprise “one or more branch nodes” that “specify one or more decision criteria to be applied.”

Neither Beaumont nor Hasan teaches these elements, alone or in combination.

EX2017, ¶¶107-118.

1. Beaumont-Hasan’s GUI does not support the setting of policies using a decision tree representing rules

Neither Beaumont nor Hasan, alone or in combination, discloses a GUI that supports the setting of policies using a decision tree representing rules. Petitioner argues that Beaumont-Hasan’s “combined GUI” teaches this element “because it is presented in a ‘familiar tree structure’ and would allow users to set server preferences and rules.” Pet., 18. Petitioner is wrong. Beaumont-Hasan does not teach a decision tree at all, let alone one that is used by a GUI to and through which a user sets a policy.

a. Neither reference has any teaching or suggestion of a decision-tree GUI.

Neither Beaumont nor Hasan teaches the claimed GUI that “supports the setting of said at least one policy using a decision tree representing rules.” At the outset, Petitioner appears to acknowledge that neither reference teaches the decision tree individually. *See* Pet., 17-18 (arguing only that the “combined GUI” teaches a decision tree). Dr. Lin stated that he had not performed an analysis as to whether Hasan alone teaches a decision tree. EX2018, 94:10-96:4. But combining silence with silence does not produce disclosure here.

The '517 patent describes the decision tree as a “policy hierarchy.” EX1001, 8:56-59; *see also* Pet., 4 (“These policies are constructed as a hierarchical decision tree”). It states that “[adaptive traffic control (ATC)] policies [] may be hierarchically constructed to form a decision tree,” and that “a policy may be created by generating a decision tree representing the ATC rules that the user (content provider/subscriber) wants to apply.” EX1001, 8:55-56, 9:1-3. The independent claims require a GUI that uses a decision tree to support the setting of a policy. An example of such GUI using a decision tree is provided in FIG. 10(b). EX1001, 25:8-25. This limitation is therefore two analytical steps beyond anything in the combination of Beaumont and Hasan.

First, Hasan does not teach a decision tree as claimed. EX2017, ¶¶109-113. Hasan’s GUI merely presents “a hierarchical layout of [a] data center, its facilities, subscriber[s], devices and/or log servers.” EX1006, 8:19-22. Hasan refers to this hierarchical layout as a “view.” EX1006, 8:19-22. The “view” is “normally shown as the familiar tree structure.” EX1006, 8:22-25. It represents “an organization of data objects” and is used to enable an administrator to “perceive[] the virtualization topology” of the data center. EX1006, 6:54-56, 8:19-22.

This view is not a decision tree. EX2017, ¶¶109-110. That it is presented in a “tree structure” does not make it one. It is merely provided as an interface so that “an administrator can view” and configure the data center objects. EX1006, 8:50-

67. Indeed, Petitioner acknowledges that the “tree structure” of the view is implemented merely to “organize information.” Pet., 14.

Hasan’s GUI thus merely represents a hierarchy of data center objects. EX2017, ¶111. It is not a hierarchy of *policies*, as described by the ’517 patent. EX2017, ¶111. In Hasan, a user navigates to and selects an object in the GUI to perform “various management capabilities” *on that object*, such as viewing or editing the properties of the object. EX1006, 8:30-34. In other words, Hasan’s GUI at most allows users to edit policies for respective data center objects. The policies themselves are not hierarchical. They are not related nor do they depend on one another. Indeed, as explained in Section VI.B.3 *supra*, Hasan does not describe the structure or underlying logic of any policy. It merely lists broad categories of policies that could be implemented for the various data center objects (e.g., “security policies”).

Second, Hasan’s GUI does not represent rules, nor does it have any underlying logic for implementing a decision tree. EX2017, ¶¶112-113. The structure of the topology (e.g., the tree structure) does not itself represent a policy hierarchy, nor does it represent any rules, as required by the claims. EX2017, ¶113. Rather, the tree merely provides a convenient “view” of the data center objects.

Beaumont suffers similar shortcomings, and thus does not fill the gaps in Hasan. EX2017, ¶114. It does not disclose a GUI at all. Although Beaumont

teaches manual entry of preference values, Petitioner acknowledges that it “does not describe a specific interface” for doing so. Pet., 14. Beaumont also does not disclose the claimed decision tree. Although its preference table can be implemented as a “tree,” EX1007, ¶23, this is not the decision tree claimed. EX2017, ¶114. Nor does Petitioner assert that it is. *See, e.g.*, Pet., 17. Rather, the tree is a “data structure” that represents a different arrangement of the preference table data. Just as the preference table does not represent rules as claimed, the preference tree would also not represent such rules, nor would it represent a policy hierarchy. *See* Sections VI.B.2-3 *supra*.

Thus, Beaumont does not teach a GUI that “supports the setting of said at least one policy using a decision tree representing rules,” as claimed.

b. The combination of Beaumont and Hasan does not teach or suggest the claimed GUI or decision tree

Because Beaumont and Hasan are both deficient in critical ways, their combination does not teach more than the sum of their parts. EX2017, ¶¶115-118. Petitioner argues that the “combined GUI” of Beaumont-Hasan teaches the claimed decision tree “because it is presented in a ‘familiar tree structure’ and would allow users to set server preferences and rules.” Pet., 18. This hypothetical extrapolation of the teachings of the references is not supported by their actual respective disclosures. EX2017, ¶115. Petitioner offers expert opinion in an attempt to bridge the gap in the combination, but Dr. Lin merely parrots the

Petition verbatim. EX1003, ¶77. The claimed decision tree requires more than the ability to set user preferences—it requires a policy hierarchy. EX2017, ¶115.

Petitioner makes no argument that the combined GUI represents such a policy hierarchy. Nor can it.

As explained above, Petitioner’s combination amounts to using Hasan’s GUI to generate a topology of content delivery servers and to input Beaumont’s preference values for the respective servers. *See* Pet., 14, 18, 21. Specifically, Petitioner argues that “the combined GUI would [] allow grouping content delivery servers by IP address block or geographic zone,” and that “selecting a server would launch a menu for assigning the preferences shown in Fig. 3 of Beaumont.” Pet., 21, 18.

But a topological layout in which users input respective preference values for each server is not a decision tree for setting a policy hierarchy. EX2017, ¶¶116-117. The view presented in the combined GUI does not even represent rules. In the combined system, the “rules” for selecting the best content delivery server are represented in the name server software, as taught by Beaumont. EX1007, ¶¶31-32. The combined GUI thus merely represents a topology of servers, as taught by Hasan. EX1006, 8:19-29. The user is unable to affect these rules using the GUI. Doing so would require an entire level of complexity and structure that the

combination of Beaumont and Hasan do not contemplate. EX2017, ¶¶113, 117-118, 130.

2. Beaumont-Hasan’s GUI does not support the setting of policies using a decision tree that includes branch nodes specifying decision criteria

Petitioner’s proposed combination of Beaumont and Hasan does not teach a decision tree that includes branch nodes that specify decision criteria to be applied in response to DNS requests.

a. Neither Beaumont nor Hasan teaches or suggests a decision tree that includes branch nodes specifying decision criteria

The cited references do not teach or suggest a GUI that supports the setting of policies using a decision tree *that includes branch nodes specifying decision criteria to be applied*. EX2017, ¶¶122-127.

Petitioner cites Beaumont’s “server health (e.g., working or down)” and “destination domains (e.g., 182.192)” as the claimed decision criteria. Pet., 19. Petitioner argues that Beaumont suggests a “decision criterion” in its depiction of rows R2, R3, and R4 in the configurable preference table (Beaumont’s Figure 2) because they “are mapped to the same destination domain, 182.192.” Pet., 19-20. But the information in the preference table is not decision criteria. EX2017, ¶¶122. The elements of rows R2, R3, and R4 of Beaumont’s preference table are—at

most—listings of the locations, capabilities, or statuses of a server. *See* EX1007, FIG. 2.

Indeed, the '517 patent describes the decision criteria as comprising the underlying rule applied via the policy. EX2017, ¶123. The decision criteria represents the actual rule, not just the parameters upon which the rule is applied. For instance, example decision criteria include “*Split on World zone*” and “*Split on country*.” EX1001, 9:17-25; *see also* EX2018, 59:16-60:10, 65:14-67-22. Here, the decision criteria is to branch or split when navigating the decision tree based on the location parameters of the requesting user. The decision criteria include the specific action to take (e.g., split) along with the parameters for which that action should be applied (e.g., world zone, country). EX2017, ¶123.

In Beaumont, the underlying decision criteria for selecting groups of servers is distinct from the information in the preference table. EX2017, ¶124. That an entry of the preference table includes a destination domain parameter does not itself provide any criteria for selecting a server. Beaumont’s underlying logic, separate and distinct from the preference table, implements the criteria used to select the servers. EX2017, ¶124. For instance, Beaumont discloses “rule[s] for selecting the best content delivery server.” EX1007, ¶31. The rules are not stored in the preference table. The rules are implemented in software. They instruct the

system to retrieve information stored in the preference tables and select servers based on this information. EX1007, ¶31.

Accordingly, the preference table listings do not provide decision criteria for selecting and grouping different content delivery servers. EX2017, ¶125.

Beaumont's rules for selecting content servers are not configurable by a user via a GUI, they are stored in the underlying software implemented by Beaumont's system. EX2017, ¶125.

What's more, even if Beaumont's destination domains (and server health) were decision criteria as claimed, Beaumont does not allow a user to input such decision criteria to be applied for DNS requests. It allows only manual entry of the preference values.² See EX1007, ¶¶31, 34. Beaumont's destination domains and server health are not input by a user. Thus, Beaumont does not disclose a GUI, or any interface, supporting the specification of such decision criteria. EX2017, ¶126.

Hasan does not teach the claimed decision criteria either, nor does Petitioner allege that it does. Hasan does not teach DNS policies, and indeed does not teach decision criteria for any policy. Thus, neither reference teaches a GUI that supports the setting of policies using a decision tree that includes branch nodes specifying

² Petitioner does not allege that the preference values represent the claimed decision criteria. See, e.g., Pet., 19-21.

decision criteria to be applied. Their collective silence fares no better. EX2017, ¶127.

b. The combination of Beaumont and Hasan does not teach or suggest a decision tree that includes branch nodes specifying decision criteria

Despite neither reference teaching a decision tree with branch nodes specifying decision criteria, Petitioner argues that the combined Beaumont-Hasan GUI teaches this element because, in the combined GUI, “content delivery servers would be grouped under the same destination domain and the destination domain would thus act as a decision criterion.” Pet., 20. Petitioner is wrong. EX2017, ¶¶128-134.

As explained, Petitioner’s combination amounts to using Hasan’s GUI to generate a topology of content delivery servers, and to input data stored in Beaumont’s preference table (i.e., preference values). *See* Pet., 13-14, 20-21. Petitioner does not argue that the combined GUI would enable users to specify the underlying decision criteria to be applied for DNS policies (e.g., select servers based on destination domain of the requestor). It only argues the ability to “group[] content delivery servers” in certain manners. Pet., 21.

To the extent Petitioner argues that the act of grouping the content delivery servers using the GUI specifies the decision criteria, Petitioner is wrong. EX2017, ¶129. Grouping servers in a hierarchy does not support the specification of the

underlying decision criteria to be applied. EX2017, ¶¶129-130. Neither reference teaches specification of such criteria, and neither does their combination. In the combined system, as in Beaumont, the underlying decision criteria is fixed in software, and would not be affected using the GUI. *See* EX1007, ¶32.

That is, arranging the servers in different manners using the GUI would have no effect on the underlying decision criteria to be applied in resolving DNS requests. EX2017, ¶130. Beaumont does not enable the user to specify such decision criteria using an interface. Neither does Hasan. Indeed, Hasan discloses many different views showing different hierarchies of the data center objects (e.g., different objects represented as roots and branches in the hierarchy). *See, e.g.*, EX1006, 9:44-10:24, FIGs. 6(a)-(e). The user can also arrange the data-center object hierarchies in various different ways. EX1006, 10:39-40, *see also* 7:48-51. Changing these hierarchies (i.e., grouping the data center objects in different manners) does not change any underlying decision criteria for policies (or rules) in Hasan because the views themselves do not represent rules or decision criteria. They merely organize data-center objects in different manners. Similarly, in the combined system, changing content delivery server hierarchies using the combined GUI would not change the underlying decision criteria or rules for selecting content delivery servers, because such decision criteria and rules are specified in software, not using the GUI. EX2017, ¶¶130-131.

Petitioner argues that, in the combined GUI, “higher-level branches” are branch nodes that “represent decision criteria,” such as destination domains. Pet., 19. But as explained above, that a topological node in the GUI represents a destination domain does not mean that that node represents *decision criteria* to be applied to DNS requests. The claims require the GUI to support the specification of decision criteria. The combined GUI does not include any such functionality. EX2017, ¶132.

Petitioner further argues this element is met because “Beaumont’s system first identifies working servers sharing the destination domain of the client (182.192) and then selects a server from within that set to handle the request based on a distance or other preferences.” Pet., 19. But Beaumont does not allow the user to specify such decision criteria via an interface. It is implemented via fixed software that is not configurable by the user. *See* EX1007, ¶32. Given that Hasan does not teach such functionality for its GUI either, the combined GUI would not allow users to specify such decision criteria. EX2017, ¶133.

Thus, grouping content delivery servers via the combined GUI as alleged does not support specification of the claimed decision criteria. As explained above, the combined GUI’s “tree-like structure” merely organizes information. It does not itself represent rules or support specification of decision criteria to be applied for DNS requests. EX2017, ¶134.

D. A POSA would not have been motivated to combine Beaumont and Hasan

A POSA would not have sought to combine Beaumont and Hasan. As explained in Section VI.A, Hasan is not analogous art to the '517 patent because it is not directed to delivering content to end users over a network. But even if Hasan was analogous to the '517 patent, a POSA still would not have combined Beaumont with Hasan. A POSA would not have combined these references at all, much less in the manner proposed by Petitioner. EX2017, ¶¶136-147.

1. A POSA would not have combined Beaumont with Hasan

A POSA who sought to modify Beaumont would not have looked to Hasan. Hasan is from a different technical field and does not address the same problems as Beaumont. EX2017, ¶¶136-138.

Beaumont is directed to a “content delivery system” comprising a domain name system (DNS) server that “receive[s] a name [e.g., domain name] from a requesting client (requester)” and selects the corresponding server that “is best to the requester.” EX1007, Abstract, ¶5. As explained above, the DNS server selects a server by accessing a “preference database” that stores “preference number[s]” that represent user preference as to which server should be selected. EX1007, ¶¶23, 31. The preference numbers are “entered manually” by the user. EX1007, ¶31.

Preference numbers

371	372	373	374	375 (optional)	376 (optional)	377 (optional)	378 (optional)	379 (optional)
310	320	321	322	330	340	350	360	number
Delivery Server Address	Destination Domain	Preference	Number of lost packets	Number of connections	Estimated distance	Number of times being selected per hour	Health	Time-To-Live (minutes)
152.192.16.15	182.192	1	100	1	100	100	Working	1
153.192.16.15	182.192	2	200	2	90	90	Down	2
154.192.16.15	182.192	9	900	9	10	10	Working	9
155.192.17.15	155.192	8	800	8	20	20	Working	8
156.192.17.15	156.192	1	100	1	10	10	Working	1
157.192.17.15	156.192	5	500	5	50	50	Working	5
158.192.17.15	156.192	3	300	3	70	70	Working	3

FIG. 3

EX1007, FIG. 3 (annotated).

A POSA would not have combined Beaumont and Hasan. EX2017, ¶¶137-138. As explained in Section VI.A, Hasan is not directed to DNS systems and does not address content delivery to users. Hasan also does not address the problem to which Beaumont relates. In particular, Beaumont addresses problems with conventional DNS systems that implemented a “round robin” server selection process, stating that, in such systems, “the best content delivery server might not be selected causing unnecessary delivery delay.” EX1007, ¶¶4-5. Beaumont addresses this problem “by having a domain name server (DNS) receive a name from a requesting client (requester) and select one of the addresses mapped to the name, that is best to the requester.” EX1007, ¶5.

To the extent a POSA wished to modify Beaumont, one would not have looked to such a disparate reference as Hasan. EX2017, ¶138. A POSA would not have combined these references at all. Accordingly, Petition Grounds 1 and 2 fail.

2. A POSA would not have combined Beaumont and Hasan as proposed by Petitioner

A POSA would not have combined Beaumont and Hasan in the manner proposed by Petitioner. EX2017, ¶¶139-147. Petitioner's combination amounts to using Hasan's GUI to generate a topology of content delivery servers and to input Beaumont's preference values for the respective servers. *See* Pet., 14, 18, 21. Specifically, Petitioner argues that “the combined GUI would thus allow grouping content delivery servers by IP address block or geographic zone,” and that “selecting a server would launch a menu for assigning the preferences shown in Fig. 3 of Beaumont.” Pet., 21, 18.

A POSA would not have combined Beaumont and Hasan in this manner. EX2017, ¶140. Beaumont does not allow users to input policies for resolving DNS requests. As explained, it teaches only one user input—of preference values. *See* EX1007, ¶¶31, 34. It does not allow users to specify groups of content delivery servers or to otherwise input any rules for resolving requests. A POSA would not have sought to implement such teachings in Beaumont at all, let alone based on the teachings of Hasan. EX2017, ¶140.

Aside from the preference values, the information used in the server selection process is gathered automatically and dynamically based on changing network conditions (e.g., network performance, round-trip-times, receipt of DNS requests). *See, e.g.*, EX1007, ¶¶26, 27, 31. This is not information that can be input by a user, because it reflects dynamic network conditions. *See, e.g.*, EX1007, ¶¶26, 27, 31; EX2017, ¶141.

What's more, as explained, Hasan does not disclose a GUI for configuring DNS services. A POSA would not have looked to Hasan to implement a GUI for configuring DNS services. Hasan does not teach creating server groups for purposes of resolving DNS requests. It merely presents a "topology," or "organization of data objects," in a data center for data center administration purposes. EX1006, 5:54-56. A POSA would not have sought to adapt this topology to depict content delivery servers because Hasan is completely unconcerned with content delivery. EX2017, ¶142.

To the extent Petitioner argues that its combination would enable users to specify actual DNS policies (i.e., rules) using Hasan's GUI, this argument also fails. EX2017, ¶¶143-147. That is, even if a POSA would have combined Beaumont and Hasan to enable manual entry of Beaumont's preference numbers via Hasan's GUI, a POSA would not have been motivated to implement Beaumont's DNS *rules* (or policies) via Hasan's GUI. EX2017, ¶143. Neither

reference teaches a GUI for inputting DNS rules or policies. Beaumont teaches only a manual entry of preference numbers, and Hasan's system does not include DNS services. *See* EX1007, ¶31; EX1006, 17:57-63.

As explained, Beaumont's underlying rule, or policy, for selecting content delivery servers is distinct from the preference number itself. Beaumont does not teach a manual entry of rules or policies—only preference numbers. Moreover, Hasan does not teach a GUI for editing DNS policies. As also explained, Hasan discloses editing generic, undefined policies for data center administration. Hasan's management system does not provide DNS services. EX1006, 17:57-63.

Thus, neither reference teaches a GUI for implementing DNS policies at all, let alone the specific policies recited in the claims. A POSA would not have sought to implement a GUI for inputting Beaumont's rules for selecting the best content delivery server. EX2017, ¶¶144-146. And even if one did, one would not have looked to Hasan for such teachings because Hasan does not disclose a GUI for inputting DNS policies.

Accordingly, a POSA would not have combined Beaumont and Hasan at all, much less to arrive at the claimed invention. Grounds 1 and 2 fail for this reason.

For the above reasons, Petitioner's Grounds 1 and 2 fail to render obvious any challenged claim.

VII. GROUNDS 3A AND 3B

Petitioner's Grounds 3A and 3B respectively rely on Ben-Shaul and the combination of Ben-Shaul and Hasan. These grounds fail to render obvious any claim in the '517 patent because: (1) the cited prior art does not teach the claimed decision tree to support the setting of at least one policy, (2) the cited prior art fails to teach or suggest providing said at least one policy to the name server network, (3) the cited prior art does not disclose a domain name server adapted to provide adaptive policy-based domain name service, and (4) a POSA would not have combined Ben-Shaul and Hasan. Moreover, as explained above, Ground 3B fails because Hasan is not analogous art to the '517 patent. As the '517 patent's independent claims are substantially similar, these arguments apply to all claims.

A. Overview of Ben-Shaul

Ben-Shaul is directed to a “content and application delivery system” that “allows content providers to directly control the delivery of content.” EX1005, Abstract; EX2017, ¶¶149-166. The system includes an “origin site... consisting of one or more origin Web servers” and “a set of subordinate edge servers which are geographically distributed and reside in the ‘edges’ of the internet.” EX1005, ¶¶35-36. The system also includes an “edge DNS server” that performs DNS services in the system. EX1005, ¶38.

1. Ben-Shaul's Edge Servers

Ben-Shaul's edge servers "provide[] a variety of unique services on behalf of" their origin site, such as "caching, mirroring, content localization, service prioritization and application delivery." EX1005, ¶214. Edge servers coordinate with their origin web servers to download content from the web servers. When a user requests content from an origin site, the edge server "intercepts" the request "to provide added targeted services relating to content." EX1005, ¶220. That is, end-user requests to the origin site are redirected to the appropriate edge server, and the edge server provides the requested content, along with "added target services," back to the user. EX2017, ¶¶150-155.

Content providers "can control the behavior of a remote edge server by defining service profiles using a 'front edge' tool." EX1005, ¶215. An edge server profile is "a collection of service and management directives that control the behavior of the edge server by enabling or disabling various services and by configuring these services with respect to service attributes, such as the target content, target time and target customers." EX1005, ¶220. The edge server directives generally describe rules for downloading content from origin web servers and for delivering content to end users. Ben-Shaul describes a number of edge server directives in "Listings" 1 through 12. *See generally* EX1005, ¶¶193-

209. Edge servers apply these directives to implement the functions specified therein.

For instance, Listing 1 describes a directive instructing an edge server to fetch a fresh copy of a web page from its origin server “as soon as [the] page expires in the edge server.” EX1005, ¶193.

Listing 1

```
if (page expired)
then if (page was modified since cached)
then fetch new copy of page.
```

EX1005, ¶193.

As another example, Listing 2 describes a directive instructing edge servers in “certain regions [to] download a given URL [from the origin server] only after five o’clock p.m. local time.” EX1005, ¶196.

Listing 2

```
<?xml version="1.0"?>
<!DOCTYPE cdml SYSTEM "CDML.dtd" >
<document url-name="news.html">
  <temporal time="17:00" action="download">
    <regional region="Western Europe"/>
    <regional region="middle East"/>
  </terrtporal>
</document>
```

EX1005, ¶196.

As yet another example, Listing 7 describes “management directives” for an edge server group (i.e., having a “group_id” of “121”). Listing 7 instructs edge

servers in the group to add specific domains to its configuration (i.e., <add rm domain action“Add” list=www.versedge.com,www.intra.versedge.com>), and to prefetch data (i.e., <cache_prefetch type=“zip”>) corresponding to these domains from the origin server at a certain time. These directives allow the edge servers to obtain website data so that they can serve this data to end-users. EX2017, ¶¶151-155.

Listing 7

```
<?xml version="1.0" standalone="yes" ?>
<!DOCTYPE mgmt (View Source for full doctype . . . )>
<mgmt version="1111">
<mgmt_set version="001" group_id="121">
<directive id="1212" eid="12121EEEE">
<add_rm_domain action“Add” list=“www.versedge.com,www.intra.versedge.com”>
<discrete Repeat“5”>
<time_element Time_Domain=“GMT” Start_Date=“01/01/2001” Start_Time“12:34”
/>
</discrete>
</add_rm_domain>
</directive>
<directive id="121212344" eid="121212FFFF">
<cache_prefetch tvpe=“zip”>
<url value=“http://www.versedge.com/cdml/top.zip” />
<discrete Repeat=“0”>
<time_element Time_Domain=“GMT” Start_Date=“01/01/2000” Start_Time=“00:00”
/>
</discrete>
</cache_prefetch>
</directive>
</mgmt_set>
</mgmt>
```

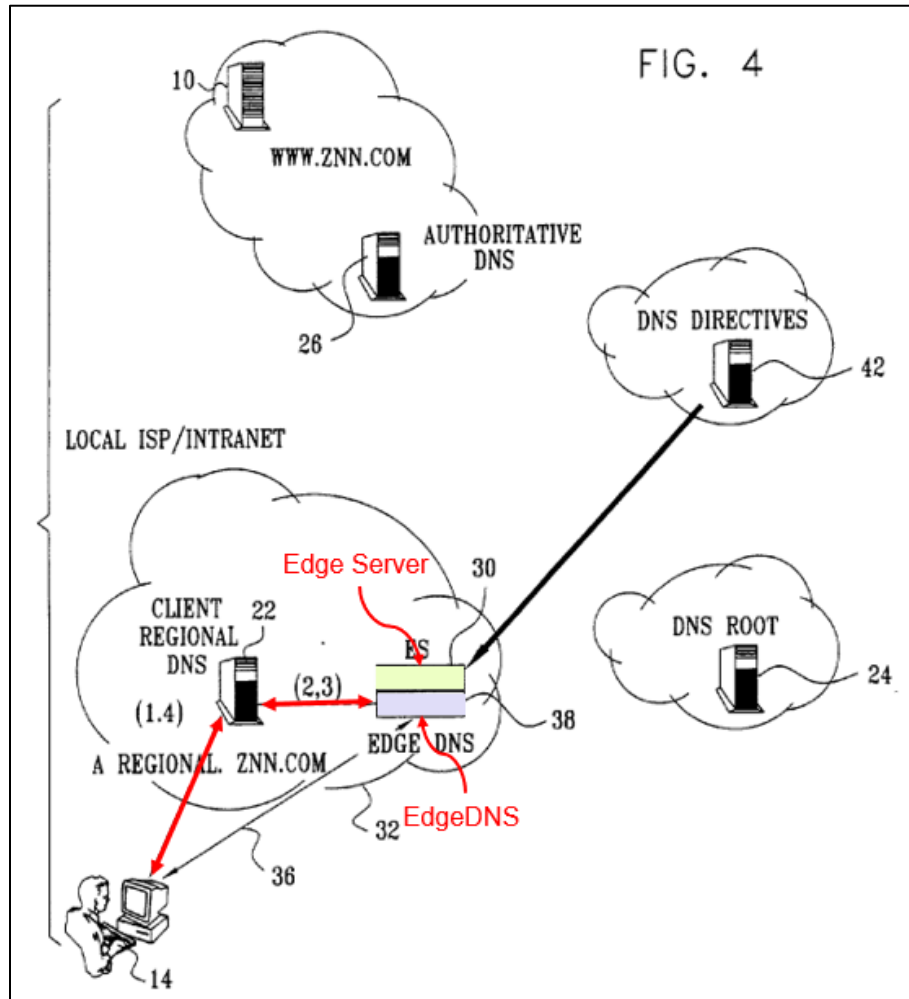
EX1005, ¶203.

2. Ben-Shaul's DNS

As explained above, Ben-Shaul also discloses an edge DNS (or EdgeDNS) server that provides DNS services within Ben-Shaul's system. EX1005, ¶41; *see also* EX2017, ¶¶156-163. "The EdgeDNS is responsible to return the IP address of either the edge server or another server for resources that whose URL resides in these domains." EX1005, ¶41. The EdgeDNS is a "conventional DNS name server" configured to route user requests to the appropriate server. EX1005, ¶359. In some embodiments, the EdgeDNS is adapted to implement "dynamic cache replacement" techniques that enable it to route user requests to the edge server, instead of the origin server, when the edge server hosts the requested content. EX1005, ¶¶360-361. This cache replacement technique enables the EdgeDNS to replace its resource record entry so that certain domains map to an edge server rather than the origin server. *E.g.*, EX1005, ¶¶361, 423.

In a representative embodiment discussed with respect to Figure 4, the EdgeDNS caches resource records mapping certain domains to the IP address of the EdgeDNS's corresponding edge server. EX1005, ¶176. In particular, a client regional DNS server 22 writes this mapping to the EdgeDNS. EX1005, ¶176. When a user requests content from one of these domains, the request is first routed to the client regional DNS server 22, which passes the request to the EdgeDNS. *Id.* The EdgeDNS identifies from its cache the IP address (e.g., of the edge server) that

is mapped to the domain, and returns the IP address to the requestor via the client regional DNS server 22. EX2017, ¶157.



EX1005, FIG. 4 (annotated).

The EdgeDNS thus stores resource records that map IP addresses to domains. EX1005, ¶¶176, 373-374. It accesses these resource records to resolve DNS requests. *E.g.*, EX1005, ¶176. FIG. 15 depicts a representative method for responding to DNS requests by the EdgeDNS. When the EdgeDNS receives a DNS query for a domain, it first determines whether it is “authoritative” for that domain (i.e., whether it holds the actual resource records mapping the IP address for that

domain). EX1005, ¶398. If so, the EdgeDNS retrieves and returns the IP address for that domain. If not, the EdgeDNS checks its cache for the requested resource record, and either returns the IP address in the event of a cache hit, or obtains the data from another DNS server in the event of a cache miss. EX1005, ¶¶398-399.

If the EdgeDNS's corresponding edge server is responsible for the requested domain, the EdgeDNS will have a resource record that maps the domain to the edge server IP address, and the EdgeDNS will return that IP address. If the edge server is not responsible, the EdgeDNS will return another server's IP address, such as the origin server. EX2017, ¶¶158-159.

Ben-Shaul's Listing 14 provides pseudo-code of a similar EdgeDNS technique for responding to DNS requests. EX1005, ¶362. As shown, upon receiving a DNS request, the EdgeDNS retrieves the relevant cache entries for the requested domain and returns either the edge server IP address (if the entry is a cache replacement (CLCR) entry) or the origin server IP address (if the entry is not a CLCR entry). EX1005, ¶362.

Listing 14

```
get cache entries relevant to the query type and domain
for each cached entry do
  if in bypass mode
    if entry type is CLCR
      ignore it
    else
      add entry
  if not in bypass mode
    if entry type is CLCR
      add entry and ignore all other types
    else if no CLCR entries found
      add entry
finally, if no entries found
  issue a normal DNS query to an external server
  after response arrives, send it back to the requester
  (while modifying the TTL value to max-ttl allowed)
```

EX1005, ¶362.

In Listing 14, the EdgeDNS distinguishes between “normal” DNS entries that map to the origin server and “canonical local cache replacement (CLCR)” DNS entries that map to the edge server. EX1005, ¶361. Upon receiving a DNS request, the EdgeDNS determines if the corresponding DNS entry in its cache is a CLCR record, and responds accordingly with the appropriate IP address. EX1005, ¶¶361-362.

Thus, the EdgeDNS provides conventional DNS services supplemented using the dynamic cache replacement technique. EX2017, ¶¶160-163. The EdgeDNS stores resource records mapping IP addresses to servers (e.g., edge servers or origin servers). In response to receiving a DNS request, the EdgeDNS

merely accesses these resource records to determine the IP address mapped to the requested domain. As with conventional DNS servers, if the EdgeDNS does not have a resource record for the requested domain, it queries another DNS server for this information.

3. Coordination of Ben-Shaul's Edge Server and Edge DNS

Ben-Shaul's EdgeDNS and edge servers coordinate to determine the appropriate servers to which to direct traffic in response to DNS requests. EX2017, ¶¶164-166. For instance, an edge server can instruct its corresponding EdgeDNS to modify its stored DNS records so that certain domains are mapped to the edge server IP address instead of the origin server address. *E.g.*, EX1005, ¶¶361-362, 387-388, 414, 423. When a subsequent end-user DNS request for those domains is received, the EdgeDNS returns the edge server IP address, as specified in the modified resource records, instead of the origin server address. EX1005, ¶362.

B. The cited prior art fails to teach or suggest the claimed decision tree

All independent claims of the '517 patent require a GUI that supports the setting of said at least one policy using a decision tree representing rules. The decision tree includes one or more resource nodes that specify one or more answers to be provided in response to a DNS request and one or more branch nodes that specify one or more decision criteria to be applied in response to the DNS request.

Petitioner's cited prior art does not teach a decision tree at all, let alone one with the required branch or resource nodes. EX2017, ¶¶167-204.

1. The cited prior art fails to teach or suggest a decision tree that includes resource nodes specifying answers to be provided in response to a DNS request

The cited prior art does not teach a decision tree that includes *resource nodes that specify answers to be provided in response to a DNS request*. Petitioner argues that three different GUIs teach or suggest the claimed decision tree: (1) Ben-Shaul's front edge tool, (2) the "Ben-Shaul single GUI," and (3) the combined Ben-Shaul-Hasan GUI. *See* Pet., 48, 50-52. But none of these GUIs teach a decision tree with resource nodes specifying answers to be provided in response to a DNS request. EX2017, ¶¶169-187.

Regarding the Ben-Shaul GUIs, Petitioner argues that the "hierarchical map" provided by these GUIs corresponds to the claimed decision tree representing rules. *See* Pet., 48, 50 (arguing that the Ben-Shaul single GUI teaches a decision tree "for the same reasons the front edge tool does"). Petitioner argues that the "leaves" of the hierarchical map specify answers "because an IP address for a selected server is provided as an answer to a DNS request." Pet., 53. But these leaves do not specify answers to DNS requests.

Ben-Shaul's GUIs, including the front edge tool and the Ben-Shaul single GUI, do not enable users to specify policies for responding to DNS requests.

EX2017, ¶¶171-172; Section VII.A *supra*. They are used to write edge server policies—they are not used to write EdgeDNS policies. *See* EX1005, ¶¶170, 211 (front edge tool is for “monitoring and managing the edge servers”), 221 (front edge tool defines “edge server profiles”), 234, 350 (“CDML instruction files for the edge server”); *see also* EX2018, 218:12-222:20. They do not specify answers to be provided in response to a DNS request as claimed. EX2017, ¶¶172-175.

In fact, as Dr. Lin acknowledged, Ben-Shaul’s EdgeDNS does not resolve DNS requests using policies. EX2018, 213:1-215:4. Instead, it resolves requests using conventional DNS adapted to implement a cache replacement technique. EX1005, ¶¶360-365. In particular, as explained above, Ben-Shaul discloses that the EdgeDNS responds to DNS requests by retrieving the resource records corresponding to the requested domain, and returning the IP address specified by the resource records. *See* EX1005, ¶¶176, 177, 362, 397-399, FIG. 15, Listing 14; *see also* Section VII.A *supra*. Nowhere does Ben-Shaul disclose that any of these techniques for resolving DNS queries are specified in policy set using a user interface. EX2017, ¶176. Petitioner does not point to any such policy, nor can it. Indeed, Dr. Lin could not identify a single DNS policy created using a user interface in Ben-Shaul. EX2018, 207:13-215:4, 221:11-222-20, 241:1-246-7.

Petitioner argues that the hierarchical map’s “edge server ‘leaves’ specify one or more answers to be provided in response to a [DNS] request’ because an IP

address for a selected server is provided as an answer to a DNS request.” Pet., 53.

Petitioner further argues that, in the example hierarchical map shown in Figure 10, “the IP address for the server ‘Haifa’ (‘one or more answers’) may be returned based on its location in Israel (‘one or more decision criteria’).” This is wrong. EX2017, ¶¶177-178.

Ben-Shaul does not consider edge server location or groups in response to receiving a DNS request. EX2017, ¶178. As explained, upon receiving a DNS request for a domain from an end-user, the EdgeDNS simply checks its stored resource records to identify the server (e.g., edge server or origin server) mapped to that domain. *See* Section VII.A *supra*. If such a record exists, the EdgeDNS returns the IP address specified in that record. If not, the EdgeDNS queries another DNS server for the IP address. The EdgeDNS does not apply any policy that considers edge server grouping or location in responding to DNS requests. It does not apply any policy governing the selection of a server based on the location of that server or its inclusion in a particular group.

Petitioner cites Ben-Shaul paragraph 41 to support its assertion that location is considered in resolving DNS requests. *See* Pet., 53-54. This paragraph merely states: “[t]he EdgeDNS is responsible to return the IP address of either the edge server or another server for resources that whose URL resides in these domains.” EX1005, ¶41. It says nothing about edge server location or the hierarchical map,

such as the one shown in Figure 10. EX2017, ¶179. Petitioner also cites Ben-Shaul paragraph 44 to support its assertion that edge servers are selected based on their “association with the group” specified in the hierarchical map. Pet., 53. But this paragraph describes *edge server* functionality for providing “differentiate[d]” content to users in different regions or groups. It does not describe techniques for resolving DNS requests. EX2018, 229:2-230:2, 233:11-16. For instance, this paragraph explains that edge servers may give different priorities to different resources, or that edge servers may make certain resources visible to end-users or organizations in different regions. *See* EX1005, ¶44, *see also* ¶197 (discussing “content-based differentiation” performed by the edge server). As Dr. Shenoy explains, these are not EdgeDNS functions, and they do not describe techniques for resolving DNS requests. EX2017, ¶¶180-181.

Petitioner thus does not provide any evidence supporting its assertion that Ben-Shaul’s DNS server answers DNS requests based on edge server groups or locations specified by the hierarchical map. This is because Ben-Shaul discloses no such functionality. Indeed, Dr. Lin could not identify a single disclosure of such functionality in Ben-Shaul. EX2018, 246:8-253:3. In particular, Dr. Lin argued in deposition that “a person of ordinary skill in the art... would understand that ... Beaumont teaches how an EdgeDNS *might be configured based on such policies*,”

but he could not identify a single policy that considers edge server grouping or location when resolving DNS requests. EX2018, 246:8-253:3 (emphasis added).

Of course, Petitioner is correct in a vacuum that “an IP address for a selected server is provided as an answer to a DNS request.” *See* Pet., 53. But the hierarchical map leaves do not specify or represent these answers. EX2017, ¶182. They do not govern which IP addresses are returned. They merely specify edge servers that are members of a group or in a particular location. The hierarchical map merely provides a “hierarchical view” of the edge servers that “allows the user to select and manipulate the hierarchies in an easy and intuitive manner.” EX1005, ¶313. It is an “interface used for browsing the edge server map.” EX1005, ¶326. It does not specify answers to DNS requests.

Thus, neither of Ben-Shaul’s GUIs (front edge tool or single GUI) teaches or suggests this element.

Hasan does not cure Ben-Shaul’s deficiencies for similar reasons provided above for the Beaumont-Hasan combination. EX2017, ¶¶184-187. Hasan’s GUI merely provides “a hierarchical layout of [a] data center, its facilities, subscriber[s], devices and/or log servers.” EX1006, 8:19-22. It is not a decision tree, it is merely “an organization of data objects” that is used to enable an administrator to “perceive[] the virtualization topology” of the data center. EX1006, 6:54-56, 8:19-22. Thus, for the reasons provided in Section VI.C

regarding Hasan, Hasan does not teach a decision tree specifying answers to be provided in response to DNS requests. EX2017, ¶185.

And as Hasan is unconcerned with DNS, it doesn't cure Ben-Shaul's deficiencies regarding the DNS aspects of this element either. EX2017, ¶186. That is, Hasan merely provides a GUI that organizes data center objects in a hierarchy. It does not disclose any DNS policies for responding DNS requests. The combination of Ben-Shaul and Hasan thus does not teach or suggest a decision tree that includes resource nodes specifying answers to be provided in response to a DNS request.

For instance, Petitioner argues, that in the combined GUI, "the policy setter could select an object, e.g., 'Jerusalem' and edit the policy applicable to that server (or group of servers)." Pet., 51. But this amounts to the same functionality of the Ben-Shaul GUIs. The combined GUI still does not enable specification of DNS policies, and thus does not specify answers for responding to DNS requests. Neither reference teaches such functionality and their combination thus does not either. EX2017, ¶187.

Grounds 3A and 3B fail for this reason.

2. The cited prior art fails to teach or suggest a decision tree that includes branch nodes specifying decision criteria to be applied

The cited prior art does not teach a decision tree that includes *branch nodes specifying decision criteria to be applied*. Neither Ben-Shaul nor Hasan, alone or in combination, teaches this element. As explained, Petitioner points to the Ben-Shaul GUIs’ “hierarchical map” and also to the Ben-Shaul-Hasan GUI as teaching the claimed decision tree. *See* Pet., 48, 50, 52. But none of these GUIs teach a decision tree with branch nodes specifying decision criteria to be applied. EX2017, ¶¶189-197.

Regarding Ben-Shaul, Petitioner argues that the “internal nodes” of Ben-Shaul’s hierarchical map are branch nodes that specify decision criteria to be applied because ‘they represent logical aggregations of nodes’ based on a particular decision criteria,” and because “policy application hinges on which of the groups is implicated.” Pet., 53. Petitioner also alleges that “[a] DNS request is thus directed to an edge server based on that server’s association with the group.” Pet., 53. Petitioner is wrong for similar reasons as explained in Section VII.B.1 *supra*. EX2017, ¶190.

As explained, Ben-Shaul’s GUIs, including the front edge tool and the Ben-Shaul single GUI, do not enable users to specify policies for providing answers in response to DNS requests. EX2017, ¶¶171-176. What’s more, the EdgeDNS does

not consider edge server groupings or location when responding to DNS requests.

The EdgeDNS does not apply any policy that considers edge server grouping or location in responding to DNS requests. EX2017, ¶¶178-183. The hierarchical map does not specify decision criteria to be applied in response to DNS requests.

EX2017, ¶191.

Moreover, Ben-Shaul's hierarchical map does not include branch nodes specifying decision criteria to be applied in *any* policy. EX2017, ¶192. That users can write edge server directives that apply to groups of edge servers does not mean that the hierarchical maps of servers themselves represent rules or decision criteria to be applied. Ben-Shaul states that “[a]dministrators of the system 80 can construct hierarchies of edge servers, *which are later used by profile editors in order to apply a particular service directive to a collection of edge servers.*” See EX1005, ¶309 (emphasis added). The directives are distinct from the groups themselves, and are applied to groups independently from, and subsequent to, the formation of the groups using the hierarchical map. EX2017, ¶192. The user can choose to apply directives to server groups, but the groups themselves do not specify any decision criteria. See EX1005, ¶309.

Indeed, the hierarchical map merely provides a “hierarchical view” of the edge servers that “allows the user to select and manipulate the hierarchies in an easy and intuitive manner.” EX1005, ¶313. It is an “interface used for browsing the

edge server map.” EX1005, ¶326. These nodes in this map do not specify any decision criteria to be applied in a policy. EX2017, ¶193.

Thus neither of Ben-Shaul’s GUIs supports the setting of policies using a decision tree that includes branch nodes specifying decision criteria to be applied.

Hasan does not cure Ben-Shaul’s deficiencies. EX2017, ¶¶195-197. As explained above, Hasan’s GUI merely provides “a hierarchical layout of [a] data center, its facilities, subscriber[s], devices and/or log servers.” EX1006, 8:19-22. Hasan refers to this hierarchical layout as a “view.” EX1006, 8:19-22. This view is not a decision tree, it is merely “an organization of data objects” that is used to enable an administrator to “perceive[] the virtualization topology” of the data center.” EX1006, 5:54-56, 8:19-22. Thus, for the reasons provided in Section VI.C.1 regarding Hasan, Hasan does not teach a decision tree as claimed.

And as Hasan is unconcerned with DNS, it doesn’t cure Ben-Shaul’s deficiencies regarding the DNS aspects of this element either. EX2017, ¶¶196-197. That is, Hasan merely provides a GUI that organizes data center objects in a hierarchy. It does not disclose any DNS policies for responding DNS requests. The combination of Ben-Shaul and Hasan thus does not teach or suggest GUI that supports the setting of policies using a decision tree that includes branch nodes specifying decision criteria to be applied.

For instance, Petitioner argues, that in the combined GUI, “the policy setter could select an object, e.g., ‘Jerusalem’ and edit the policy applicable to that server (or group of servers).” Pet., 51. But this amounts to the same functionality of the Ben-Shaul GUIs. The combined GUI still does not enable specification of DNS policies, and thus does not specify answers for responding to DNS requests. Neither reference teaches such functionality and their combination thus does not either. EX2017, ¶197.

Grounds 3A and 3B fail for this reason.

3. The cited prior art fails to teach or suggest a decision tree at all

Not only does the cited prior art fail to teach the decision tree having specific branch and resource nodes, it also fails to teach a decision tree at all. EX2017, ¶¶199-204. Neither Ben-Shaul nor Hasan, alone or in combination, teaches or suggests a decision tree representing rules. As explained, Petitioner points to the Ben-Shaul GUIs’ “hierarchical map” and the Ben-Shaul-Hasan GUI as teaching the claimed decision tree. *See* Pet., 48, 50, 52. But none of these GUIs teach a decision tree that “represent[s] rules.”

Petitioner argues that the hierarchical map provided by the Ben-Shaul GUIs, including the front edge tool and the Ben-Shaul single GUI, is a decision tree representing rules “because directives apply based on inclusion of edge servers within the groups.” Pet., 48. Petitioner is wrong. EX2017, ¶¶199-200. Ben-Shaul’s

hierarchical map merely provides a “hierarchical view” of the edge servers that “allows the user to select and manipulate the hierarchies in an easy and intuitive manner.” EX1005, ¶313. It is an “interface used for browsing the edge server map.” EX1005, ¶326. It does not represent rules.

That users can apply directives to edge server groups does not mean that the hierarchical map itself is a decision tree that represents rules. As explained in Section VI.C.1, the ’517 patent describes the claimed decision tree representing rules as a “policy hierarchy.” EX1001, 8:56-59. Conversely, Petitioner admits that the hierarchical map merely represents “hierarchies of edge servers.” Pet., 52. That this hierarchy of edge servers can be used “for effecting directives” does not make it a decision tree representing rules as claimed. *See* Pet., 52. Rather, it merely represents a topology of edge servers to which policies can be applied. EX2017, ¶201.

Ben-Shaul teaches that the edge server groups “simplify the application of policies applicable to a set of related edge servers.” EX1005, ¶309. But as explained above, the edge server groups in the hierarchical map do not represent any rules. EX2017, ¶202. Ben-Shaul states that “[a]dministrators of the system 80 can construct hierarchies of edge servers, *which are later used by profile editors in order to apply a particular service directive to a collection of edge servers.*” *See* EX1005, ¶309 (emphasis added). The directives are distinct from the groups

themselves, and are applied to groups independently from, and subsequent to, the formation of the groups using the hierarchical map. The user can choose to apply directives to server groups, but the groups themselves do not represent any rules. *See* EX1005, ¶309.

Hasan does not cure Ben-Shaul's deficiencies. EX2017, ¶¶203-204. It does not teach the claimed decision tree for the reasons described above in Section VI.C.1. Moreover, Petitioner's proposed combined Ben-Shaul-Hasan GUI is redundant to Ben-Shaul's disclosed GUIs given that Ben-Shaul already discloses a hierarchy of servers.

According to Petitioner, the Ben-Shaul-Hasan combined GUI would merely enable a user to "select an object" such as a server or group of servers, "and edit the policy applicable to that server (or group of servers)." Pet., 51. But, as with the Hasan GUI, this merely amounts to a topological layout that enables users to input respective policies for servers. The Ben-Shaul-Hasan GUI does not represent a policy hierarchy, it is at best a hierarchy of content delivery servers. EX2017, ¶204. It merely arranges content delivery servers in a tree structure. The claimed GUI must do more than allow users to set policies for content delivery servers, it must support the specification of policy using a decision tree that itself represents rules. The Beaumont-Hasan GUI does not teach or suggest such a decision tree. EX2017, ¶204.

C. The cited prior art fails to teach or suggest providing said at least one policy to the name server network

The cited prior art fails to teach or suggest a policy that is obtained using a GUI and provided to a name server network as claimed. EX2017, ¶¶206-207. The independent claims require “using said GUI to obtain at least one policy for direction of network traffic” and “providing [the same] at least one policy to the name server network.” Ben-Shaul-Hasan does not teach these elements.

Petitioner identifies only Ben-Shaul’s “DNS directives” as the policies that are provided to the name server network. *See* Pet., 52 (citing Ben-Shaul, ¶177, which discloses “DNS directives”). But these DNS directives are not obtained using a GUI. EX2017, ¶207. Indeed, Ben-Shaul only mentions DNS directives twice, stating only that they are used for “configuration of the EdgeDNS 40 and its coordination with the client regional DNS server 22” and that they “originate from one or more policy control servers 42.” EX1005, ¶177, *see also* ¶176. Petitioner does not identify any evidence that these DNS directives are obtained or set using a GUI, as claimed. There is none.

Thus, Ben-Shaul-Hasan does not teach these elements.

D. Ben-Shaul does not disclose a domain name server adapted to provide adaptive policy-based domain name service

Petitioner fails to show that Ben-Shaul discloses a domain name server adapted to provide adaptive policy-based domain name service. EX2017, ¶¶210-

212. Petitioner points to Ben-Shaul’s EdgeDNS as corresponding to the claimed domain name server. Pet., 44. Petitioner alleges that the EdgeDNS provides adaptive policy-based domain name service because “requests are handled dynamically based on network conditions,” and cites only to Ben-Shaul paragraph 65 for support. Pet., 46. But paragraph 65 describes conditions for when an *edge server* may pre-fetch content from its corresponding origin server. It does not describe domain name service functionality, nor does it describe functionality of the EdgeDNS. EX2017, ¶210.

Edge server content pre-fetching is unrelated to the DNS functionality of the system. EX2017, ¶211. It is an interaction between the edge server and its origin server for determining when the edge server should download content from the origin server. It is not domain name service, nor is it performed by the EdgeDNS (alleged domain name server). The EdgeDNS is responsible for mapping domains to the appropriate server IP address, and for handling end-user DNS requests by “return[ing] the IP address of either the edge server or another server for resources that whose URL resides in these domains.” EX1005, ¶41. It is not responsible for pre-fetching of content by the edge server, and even if it was, this would not be “adaptive policy-based domain name service.”

Indeed, Ben-Shaul’s EdgeDNS does not provide *adaptive* policy-based domain name service. EX2017, ¶212. It simply returns IP addresses that are

specified in its resource records, or that are provided to it by other DNS servers.

See Section VII.A *supra*. It does not apply adaptive policies in response to receiving DNS requests, nor does it respond to DNS requests (i.e., provide domain name service) based on “changing network conditions and/or changing user directives,” as alleged by Petitioner. *See* Pet., 74. As explained above, it has fixed rules for responding to DNS requests and for configuring its internal resource records. It simply follows these rules to identify the appropriate IP address to return. It does not apply adaptive policies.

Grounds 3A and 3B fail for this additional reason.

E. A POSA would not have combined Ben-Shaul and Hasan

A POSA would not have sought to combine Beaumont and Hasan at all, much less to arrive at the challenged claims.

As explained in Section VI.A, Hasan is not analogous art to the ’517 patent because it is not directed to delivering content to end users over a network. For similar reasons, it is also not analogous to Ben-Shaul. Accordingly, a POSA would not have sought to combine Ben-Shaul and Hasan. EX2017, ¶¶215-219.

Ben-Shaul is directed to a “content and application delivery system” that “allows content providers to directly control the delivery of content.” EX1005, Abstract. The system includes an “origin site... consisting of one or more origin Web servers” and “a set of subordinate edge servers which are geographically

distributed and reside in the ‘edges’ of the internet.” EX1005, ¶¶35-36. The system also includes an “edge DNS server” that performs DNS services in the system. EX1005, ¶38.

A POSA would not have sought to combine Ben-Shaul with Hasan. EX2017, ¶¶216-217. Hasan is not directed to DNS systems and does not address content delivery to users. Hasan also does not address the problem to which Ben-Shaul relates. In particular, Ben-Shaul addresses problems with conventional “content delivery” systems, such as problems related to the “centralized control” provided by such systems, their scalability issues, the lack of control these systems give to content providers, their inability to provide dynamic content, etc. *See* EX1005, ¶¶18-24.

Hasan does not teach a GUI for editing content delivery policies. As explained, Hasan discloses editing generic, undefined policies for data center administration. Hasan’s management system does not provide content delivery or DNS policies. EX1006, 17:57-63. It leave DNS to be “implemented by other systems.” EX1006, 17:57-63. A POSA would not have looked to Hasan for such teachings because Hasan does not disclose a GUI for inputting content delivery policies. EX2017, ¶¶218.

To the extent a POSA wished to modify Ben-Shaul, one would not have looked to such a disparate reference as Hasan. A POSA would not have sought to

combine these references at all, much less to arrive at the claimed invention.

Ground 3B fails for this reason. EX2017, ¶¶219.

For the above reasons, Petitioner's Grounds 3A and 3B fail to render obvious any challenged claim.

VIII. CONCLUSION

The Board should uphold the patentability of all challenged claims of the '517 patent.

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX PLLC

/Jason D. Eisenberg/

Jason D. Eisenberg, Reg. No. 43,447

Daniel Block, Reg. No. 68,395

Kristina Caggiano Kelly, Reg. No. 58,164

David Haars, Reg. No. 73,112

Xinxiang Zhang, Reg. No. 82,382

Counsel for Patent Owner

Date: February 9, 2026

1101 K Street, NW

10th Floor

Washington, DC 20005

(202) 371-2600

CERTIFICATE OF WORD COUNT (37 C.F.R. § 42.24(d))

1. This Patent Owner's Response complies with the type-volume limitation of 14,000 words, comprising 13,789 words, excluding the parts exempted by 37 C.F.R. § 42.24(a)(1).
2. This Patent Owner's Response complies with the general format requirements of 37 C.F.R. § 42.6(a) and has been prepared using Microsoft® Word 2016 in 14-point Times New Roman font.

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX PLLC

/Jason D. Eisenberg/

Jason D. Eisenberg
Registration No. 43,447
Counsel for Patent Owner

Date: February 9, 2026

1101 K Street, NW
10th Floor
Washington, DC 20005
(202) 371-2600

CERTIFICATE OF SERVICE (37 C.F.R. § 42.6(e))

I certify that the above-captioned **PATENT OWNER'S RESPONSE** and associated **Exhibits 2017-2018** were served in their entireties on February 9, 2026, upon the following parties via electronic mail:

Erika H. Arner (Lead Counsel)	erika.arner@finnegan.com
Kara A. Specht (Back-Up Counsel)	kara.specht@finnegan.com
Cara E. Regan (Back-Up Counsel)	cara.regan@finnegan.com
Sydney R. Kestle (Back-Up Counsel)	sydney.kestle@finnegan.com
William C. Neer (Back-Up Counsel)	william.neer@finnegan.com
Daniel C. Tucker (Back-Up Counsel)	daniel.tucker@finnegan.com
Christopher B. Anderson (Back-Up Counsel)	christopher.anderson@finnegan.com
	Google-Sandpiper-IPRs@finnegan.com
FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER, LLP	

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX PLLC

/Jason D. Eisenberg/

Jason D. Eisenberg
Registration No. 43,447
Counsel for Patent Owner

Date: February 9, 2026

1101 K Street, NW
10th Floor
Washington, DC 20005
(202) 371-2600