

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

WALMART INC. and WALMART STORES TEXAS, LLC,
Petitioner

v.

RAVENWHITE SECURITY, INC.,
Patent Owner

Case IPR2025-00810
U.S. Patent No. 10,594,823

**PATENT OWNER PRELIMINARY RESPONSE
UNDER 37 C.F.R. § 42.107(a)**

Mail Stop "PATENT BOARD"
Patent Trial and Appeal Board
U.S. Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-145

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	THE '823 PATENT.....	3
	A. The '823 patent provides novel techniques for enhancing the ability of websites to identify client devices.	3
	B. Level of Ordinary Skill in the Art.....	6
	C. Claim Construction	7
	1. “browser storage area”	7
III.	OVERVIEW OF THE CITED ART	19
	A. Hinton describes session cookies for authenticating a user across domains, which are established and valid only during the user’s current session.....	19
	B. Varghese describes capturing device identity information within cookies stored outside of a user’s web browser storage.	21
IV.	GROUND 1: HINTON DOES NOT RENDER OBVIOUS ANY CLAIM OF THE '823 PATENT.....	24
	A. The Petition fails to address the language of elements [1.b.iv] and [6.a.iv] and, therefore, Petitioner does not meet its burden to establish obviousness of these elements.	25
	B. Hinton’s e-community cookie is not “caused to be stored at the client device during a second previous network session.”	28
	C. Petitioner’s challenge further fails because the Petition does not establish that storage of “a second cookie . . . during a second previous network session,” as recited in elements [1.b.iv] and [6.a.iv], occurs prior to receiving the “network resource request” of elements [1.b.i] and [6.a.i], as the claims require.....	32
V.	FOUNDATIONS 2 AND 3: VARGHESE, ALONE OR IN COMBINATION WITH HINTON, DOES NOT RENDER OBVIOUS ANY CLAIM OF THE '823 PATENT.	37
	A. The Petition fails to establish that Varghese renders obvious elements [1.b.v] and [6.a.v] (“wherein the first cookie of the first type is stored in a first client device browser storage area and the second cookie of the second type is stored in a second	

client device browser storage area different from the first client
device browser storage area”).37

B. The Petition fails to establish that Varghese renders obvious
elements [1.b.iv] and [6.a.iv] (“wherein a second cookie of a
second type different from the first type was caused to be stored
at the client device during a second previous network session”).....42

VI. CONCLUSION.....48

PATENT OWNER'S EXHIBIT LIST

Exhibit No.	Description
2001	Docket Control Order [43], <i>RavenWhite Licensing LLC v. The Home Depot, Inc. et al.</i> , Case No. 2:24-cv-00688 (E.D. Tex. Nov. 22, 2024)
2002	Complaint [1], <i>RavenWhite Licensing LLC v. Walmart Inc. et al.</i> , Case No. 2:23-cv-00418 (E.D. Tex. Sept. 15, 2023)
2003	<i>Intentionally Left Blank</i>
2004	Results charted from DocketNavigator.com, run on June 9, 2025, for success of Motions to Stay Pending IPR in the Eastern District of Texas
2005	Results charted from DocketNavigator.com, run on June 9, 2025, for success of Motions to Stay Pending IPR decided by Judge Rodney Gilstrap
2006	PREVAIL Act Fact Sheet
2007	Walmart Balance Sheet, accessed June 12, 2025 at: https://stock.walmart.com/financial-information/balance-sheet
2008	Fortune Announces 2025 Fortune 500 List, accessed June 12, 2025 at: https://www.prnewswire.com/news-releases/fortune-announces-2025-fortune-500-list-302470158.html
2009	Ventura, L., “World’s Largest Companies in 2024,” Global Finance Magazine, accessed June 12, 2025 at: https://gfmag.com/data/biggest-company-in-the-world/
2010	Morris, A., “IPBC Global 2025: Acting USPTO Director says IPR Use Needs to Change,” accessed June 16, 2025 at: https://www.iam-media.com/article/ipbc-global-2025-acting-uspto-director-says-ipr-use-needs-change
2011	Hassan Djirdeh, “The Three Browser Storage Mechanisms,” Progress Software, available at https://www.telerik.com/blogs/three-browser-storage-mechanisms
2012	Croft <i>et al.</i> , “Complete Guide to Cookies and Where They’re Stored,” All About Cookies, available at https://allaboutcookies.org/what-is-a-cookie-file

Exhibit No.	Description
2013	“In which location cookies are stored on the hard disk?” GeeksforGeeks, available at https://www.geeksforgeeks.org/javascript/in-which-location-cookies-are-stored-on-the-hard-disk/
2014	“How to delete cookies in Netscape,” BlackRock, available at https://blackrock.tal.net/vx/lang-en-GB/mobile-1/brand-3/candidate/faq/how_to/local/8
2015	“Visual Design Evolution of Netscape Navigator,” Version Museum, available at https://www.versionmuseum.com/history-of/netscape-browser
2016	“How to clean Infected Temporary Internet Files in Windows,” Bitdefender, available at https://www.bitdefender.com/consumer/support/answer/2138/
2017	“Browser history logs,” NXLog, available at https://docs.nxlog.co/integrate/browser-history.html
2018	“Adobe Flash Player 10 Administration Guide,” Adobe Systems Incorporated (2008), available at https://web.archive.org/web/20081216032621/http://www.adobe.com/devnet/flashplayer/articles/flash_player_admin_guide.html
2019	“Flash Player Help – Local Storage settings,” Adobe Systems Incorporated (2008), available at https://web.archive.org/web/20081025230757/http://www.macromedia.com/support/documentation/en/flashplayer/help/help02.html
2020	Deposition Transcript of Craig Ellis Wills, Ph.D., June 17, 2025
2021	<i>Curriculum Vitae</i> of Bernard J. Jansen, Ph.D.
2022	“What Is the 127.0.0.1 IP Address?,” Lifewire, available at https://www.lifewire.com/network-computer-special-ip-address-818385
2023	“Add pictures or attach files to emails in Outlook,” Microsoft, available at https://support.microsoft.com/en-us/office/add-pictures-or-attach-files-to-emails-in-outlook-bdfafef5-792a-42b1-9a7b-84512d7de7fc

Exhibit No.	Description
2024	“Send attachments with your Gmail message,” Google, available at https://support.google.com/mail/answer/6584
2025	“Open & download attachments in Gmail,” Google, available at https://support.google.com/mail/answer/30719
2026	“Adobe Flash Player 32.0 Administration Guide,” Adobe Systems Incorporated (2020), available at https://developer.adobe.com/flash/devnet/flashplayer/articles/flash-player-admin-guide
2027	Declaration of Dr. Bernard J. Jansen, Ph.D.
2028	Herrman, J., “What Are Flash Cookies and How Can You Stop Them?,” Popular Mechanics, accessed at https://www.popularmechanics.com/technology/security/how-to/a6134/what-are-flash-cookies-and-how-can-you-stop-them/ (Sept. 23, 2010)
2029	“What are Flash Cookies and how do they Work?,” CookieScan, accessed at https://www.cookiescan.com/what-are-flash-cookies-how-do-they-work/
2030	“What are session cookies?,” CookieYes, available at https://www.cookieyes.com/blog/session-cookies/
2031	Email from Jennifer Nall dated Jan. 13, 2025
2032	Raymond Camden, <u>Client-Side Data Storage</u> , O’Reilly Media, Inc. (2016)

I. INTRODUCTION

U.S. Patent No. 10,594,823 (“the ’823 patent”) provides novel techniques that allow websites to identify client devices while addressing security concerns typically associated with traditional cookies. EX1001, 2:66-3:4. The ’823 patent accomplishes these goals through use of “cache cookies that the server can use to identify the client device (or user),” “[e]ven if the browser is blocking cookies” or “has deleted its cookies.” *Id.*, 2:66-3:4, 6:49-53.

Petitioner presents two obviousness grounds challenging independent claims 1 and 6 of the ’823 patent: one based on Hinton (EX1005) and one based on Varghese (EX1004). Petitioner fails to establish that either Hinton or Varghese teaches the features of independent claims 1 and 6. Therefore, Petitioner fails to establish that any claim of the ’823 patent is unpatentable.

For the Hinton grounds, the Petition fails to establish that Hinton renders obvious “*wherein a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session,*” as recited in independent claims 1 and 6 (elements [1.b.iv] and [6.a.iv]), for two independent reasons. First, the Petition fails to show that Hinton teaches these claim elements because it offers no theory as to how Hinton’s eCC “*was caused to be stored at the client device during a second previous network session,*” as required by the claims. Petitioner, in fact, does not address the term “*previous*” at

all in its analysis of these elements, and thus does not meet its burden to establish obviousness. Second, even had the Petition addressed the language of elements [1.b.iv] and [6.a.iv], Petitioner could not show that Hinton's operation renders obvious this claim language. Hinton's e-community cookie is a session cookie that is generated and valid only during a *current* network session—not a “*previous*” one, as required by the claims.

Additionally, the Petition fails to establish that storage of “*a second cookie . . . during a second previous network session*” as recited in elements [1.b.iv] and [6.a.iv], occurs prior to receiving the “*network resource request*” of elements [1.b.i] and [6.a.i], as independent claims 1 and 6 require. Petitioner relies on the same request in Hinton for each of these elements, and thus Petitioner does not establish that one request occurs before the other.

For the Varghese grounds, Petitioner first fails to establish that Varghese teaches that “*the first cookie of the first type is stored in a first client device browser storage area and the second cookie of the second type is stored in a second client device browser storage area different from the first client device browser storage area*” (elements [1.b.v] and [6.a.v]). Petitioner relies on flash cookies for the claimed “*second cookie*,” but both experts agree that flash cookies are stored in a storage location dedicated to Flash Player software—not a browser.

Indeed, the evidence shows that browser applications do not even have access to these stored flash cookies.

Petitioner also fails to establish that Varghese teaches that “*a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session*” (elements [1.b.iv] and [6.a.iv]).

Petitioner’s argument relies on a hypothetical scenario in which Varghese recreates cookies after being cleared by a user. But this operation is neither described nor suggested in Varghese, and Petitioner provides no reason to modify Varghese in this manner. Petitioner, therefore, does not meet its burden with respect to this element.

For these reasons, the Board should deny institution of this proceeding.

II. THE ’823 PATENT

A. The ’823 patent provides novel techniques for enhancing the ability of websites to identify client devices.

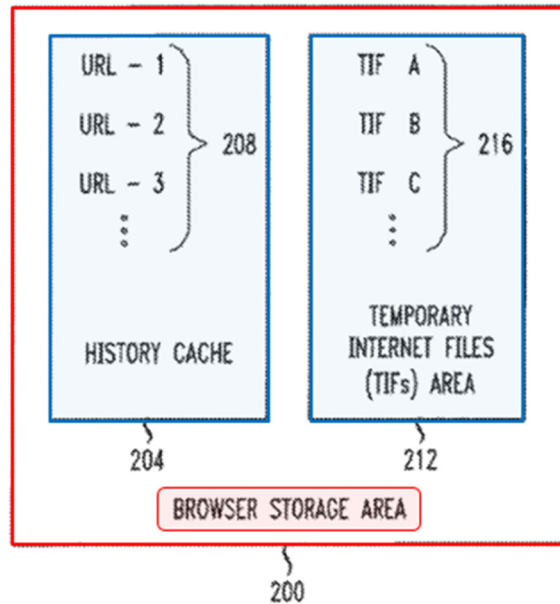
The ’823 patent generally relates to “causing a browser to store information in a browser storage area of a client device” that can later be used for “identification of a user.” EX1001, 1:31-34, 2:5-6. The ’823 patent explains that “[a] Web page . . . may request user information from the user when the user first accesses the page, such as a user’s name, password, address, interests, etc.” *Id.*, 1:39-42. “When the user accesses the same Web page at a later time, the server

may use the information previously entered by the user to customize the Web page for the user.” *Id.*, 1:42-45; EX2027, ¶28.

The ’823 patent further explains that “[t]his customization of a Web page is typically the result of cookies,” which are “message[s] transmitted to a browser by a server.” EX1001, 1:46-48. “The message can include user-specific identifiers or personal information about the user,” and “[t]he message (i.e., cookie) is then sent back to the server each time the browser requests a Web page from the server.” *Id.*, 1:48-52. However, “[d]espite the benefits associated with customizing a Web page, cookies also present drawbacks.” *Id.*, 2:11-13. “As a result, some people block or clear cookies,” which results in organizations such as “banks [to] lose another technique to identify the user.” *Id.*, 2:24-40; EX2027, ¶29.

As the ’823 patent explains, to circumvent issues associated with traditional cookies, “one or more servers instead ‘write’ and ‘read’ a cache cookie to and from a browser storage area associated with a browser requesting a Web page from the server(s).” EX1001, 2:66-3:3. The patent describes exemplary embodiments in which “[t]he browser storage area may include a history cache and/or a Temporary Internet Files (TIFs) area.” *Id.*, 3:3-4; EX2027, ¶30. These two storage areas are illustrated in Figure 2.

FIG. 2



EX1001, FIG. 2 (annotated).

The “history cache 204 [] contains Uniform Resource Locators (URLs) 208 recently visited by the browser (also called browser history).” EX1001, 6:4-8. “A server can ‘write’ any of a wide variety of cache cookies,” e.g., URLs “recently visited by the browser,” “in the history cache 204 to, for instance, facilitate the identification of a client device (or user).” *Id.*, 6:4-8, 6:22-24. Then, “when the user revisits” a URL, “the server can ‘read’ the history cache 204 of the client device to determine what Web pages the browser has recently visited.” *Id.*, 6:42-45. “The server can use the pattern of URLs stored in the browser's history cache 204 to, e.g., identify the client device (or user).” *Id.*, 6:47-49. And “[e]ven if the browser is blocking cookies (or has deleted its cookies . . .), the history cache 204 still

contains the cache cookies that the server can use to identify the client device (or user).” *Id.*, 6:49-53; EX2027, ¶¶31-32.

“The browser storage area 200 can also include a Temporary Internet Files (TIFs) area 212 for storing TIFs 216,” which “are files containing information embedded in Web pages.” EX1001, 6:54-57. When a user returns to a URL, “[t]he server can use [a] determination (i.e., of which specific image files the browser retrieves from its local TIF area 212) to identify the browser.” *Id.*, 7:25-32. The patent explains that “[a]s a result, the TIFs stored in the TIF area 212 of a browser are an embodiment of cache cookies” that “persist indefinitely.” *Id.*, 6:64-67, 7:33-34; EX2027, ¶33.

Thus, by employing “cache cookie[s],” the ’823 patent addresses issues associated with traditional cookies, while still allowing websites to identify client devices. EX1001, 2:66-3:4. Importantly, the invention utilizes native browser features (e.g., browser history and temporary Internet files) so that a user does not need to install additional applications on the client device to achieve the benefits of the invention. EX2027, ¶34.

B. Level of Ordinary Skill in the Art

A person of ordinary skill in the art (“POSITA”) at the time of the ’823 patent would have had a bachelor’s degree in computer science, computer engineering, electrical engineering, or a similar discipline, as well as two years of

academic or industry experience in computer networking, or comparable industry experience. EX2027, ¶27.

C. 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). Patent Owner asserts that no term requires express construction beyond its plain and ordinary meaning as understood by a POSITA at the time of the invention in light of the specification and the patent’s prosecution history. However, the plain and ordinary meaning of the term “*browser storage area*” is relevant to the disputes in this proceeding and, therefore, is discussed in detail below.

1. “browser storage area”

The plain and ordinary meaning of “*browser storage area*” in the context of the ’823 patent and prosecution history is a storage area managed by and native to a browser application on the client device. EX2027, ¶36. In its Institution Decision in related proceeding IPR2024-01316, the Board preliminarily found that “the claimed ‘browser storage area’ is broad in scope,” and that as long as a storage area is “accessible by the user’s browser,” the storage area would fall within the scope of the claim term. IPR2024-01316, DI, 51. This broad interpretation of “*browser storage area*,” however, is inconsistent with the ordinary meaning of the

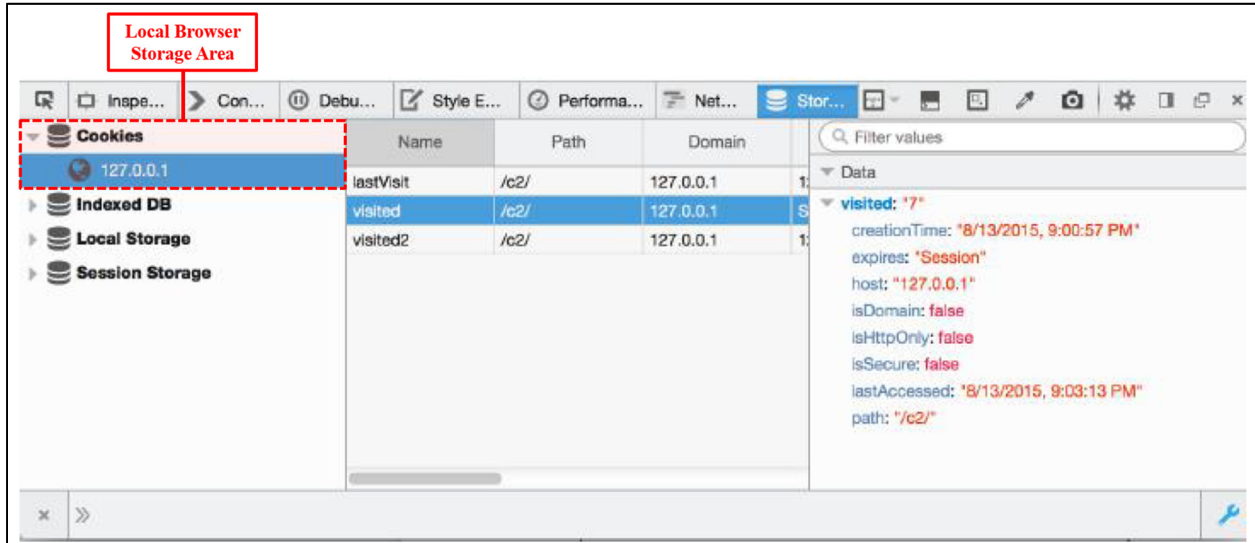
term and improperly reads the word “*browser*” out of the claim term. EX2027, ¶¶36-53.

a. As Patent Owner’s expert, Dr. Jansen, explains, browser applications have employed “client-side data storage” mechanisms for decades. EX2027, ¶36 (citing EX2032, 1¹). Rather than storing data at the server, browser applications offer “a powerful alternative—**storing data on the browser itself.**” EX2032, 1 (emphasis added). “This enables the browser to skip asking the server for information and to simply retrieve it locally from the user’s machine.” *Id.*; EX2027, ¶36.

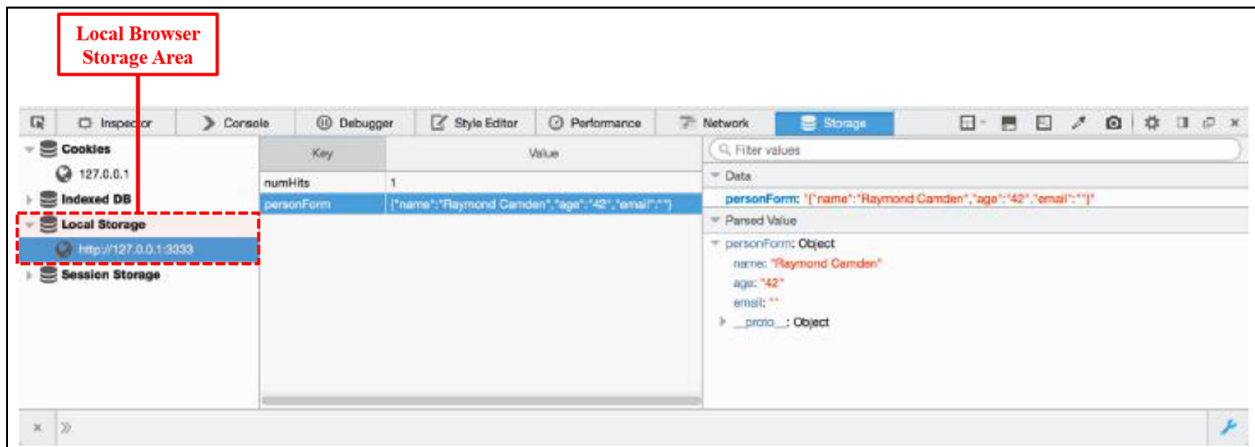
Around the time of the ’823 patent’s priority date (and still today), one primary form of client-side data storage was cookies, which were first introduced in 1994 as part of the Netscape web browser application. EX2032, 3; EX2027, ¶37. Cookies set by the browser application are stored in a storage area native to the browser on the user’s local machine. EX2027, ¶38. For example, as Camden illustrates in its book, Mozilla Firefox designates storage areas on the local machine for storing cookies, as well as other local browser data, which can be

¹ Citations to EX2032 reference original book page numbers (displayed at the bottom of each page) rather than page numbers of the PDF exhibit.

viewed from within the browser application. EX2032, 10-11; EX2027, ¶38; *see also* EX2011 (discussing “different storage mechanisms of the web browser”).



EX2032, 11 (FIG. 2-1, annotated).

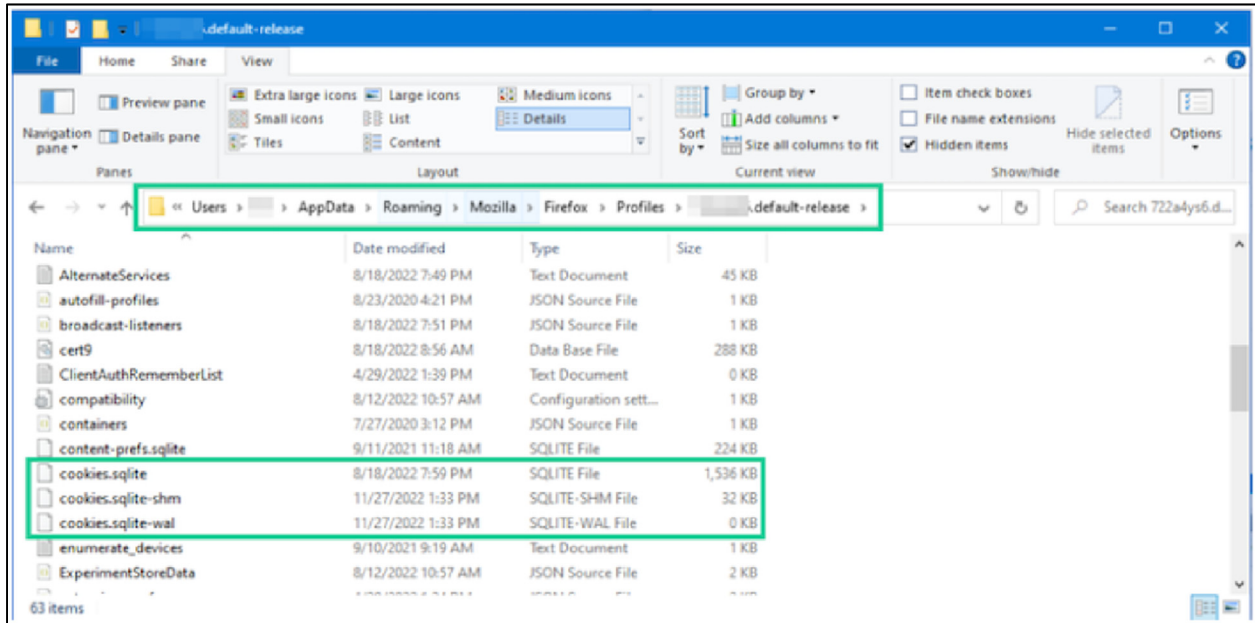


EX2032, 23 (FIG. 3-6, annotated).

As Dr. Jansen explains, the storage location illustrated above corresponds to a file system folder on the user’s local machine specific to the user’s browser application. EX2027, ¶39. For example, on the Windows operating system,

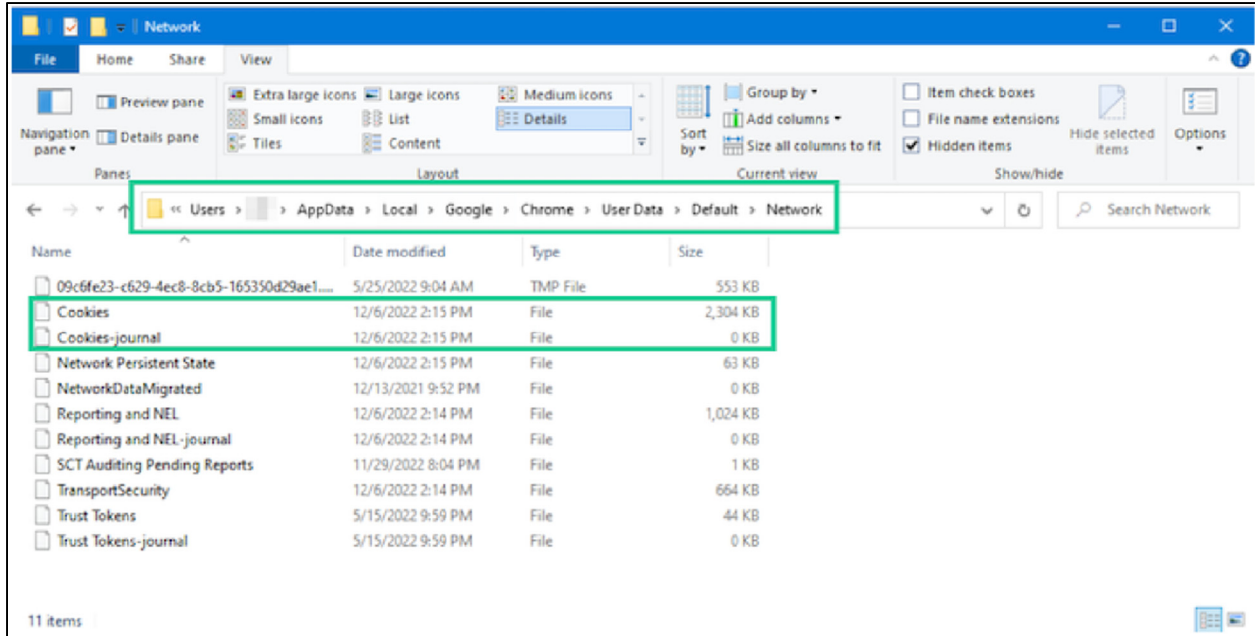
Mozilla Firefox typically stores this cookie data at

“C:\Users\Your_User_Name\AppData\Roaming\Mozilla\Firefox\Profiles”—a folder native to and dedicated for use by the Firefox browser application—as illustrated below. EX2012, 7; EX2027, ¶39.



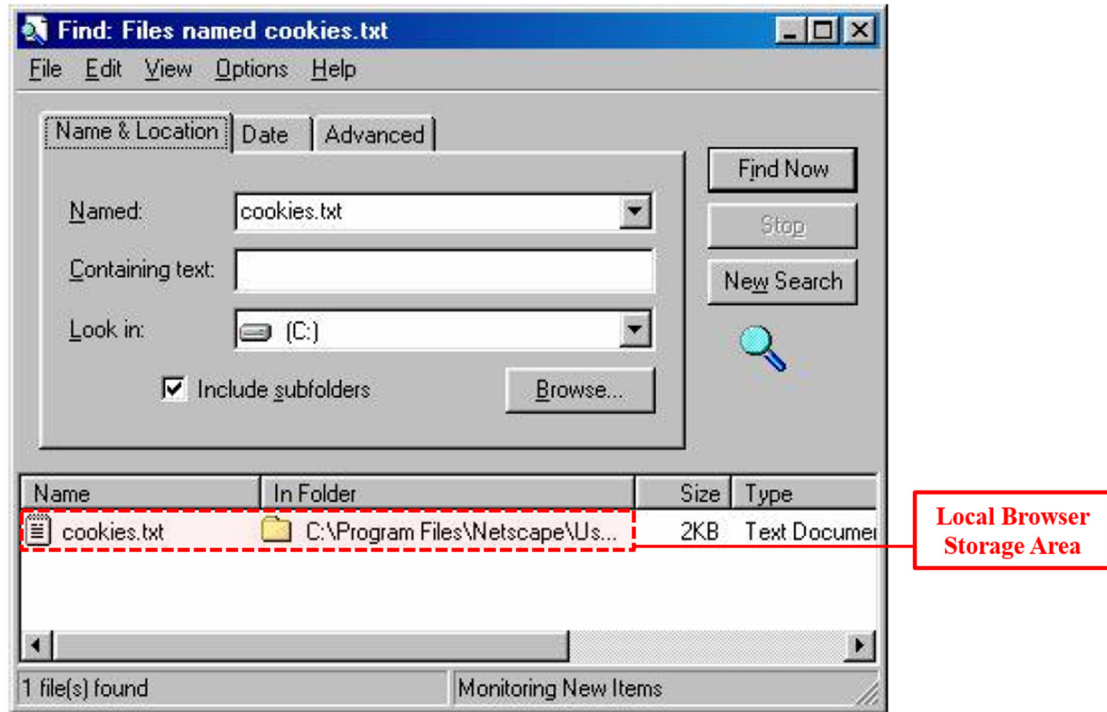
EX2012, 8.

Similarly, on Windows, Google Chrome typically stores cookie data at “C:\Users\Your_User_Name\AppData\Local\Google\Chrome\User Data\Default\Network”—again, a folder native to and dedicated for use by the Chrome browser application—as illustrated below. EX2012, 6; EX2013, 2; EX2027, ¶40.



EX2012, 7.

This convention has remained the same since cookies were introduced as part of the Netscape Navigator browser application. For example, in Netscape Navigator version 4, introduced in 1997 (*see* EX2015, 25-26), browser cookie data was stored in a file named “cookies.txt” located within a folder native to and dedicated for use by Netscape Navigator, as illustrated below. EX2014, 3; EX2027, ¶41.



EX2014, 3 (annotated).

The same understanding holds true for other browser data such as Temporary Internet Files and browser history. EX2027, ¶42. As the '823 patent explains, “Temporary Internet Files (TIFs) . . . are files containing information embedded in Web pages,” such as “graphics (e.g., one or more icons) (e.g., .JPG file) that are downloaded by a browser when the browser requests the Web page.” EX1001, 6:54-60; EX2027, ¶42. Those “files (e.g., images, text, style sheets, etc.)” are then stored in a “browser storage area (e.g., browser cache)” of the browser application. EX1001, 4:60-63; EX2027, ¶42.

As Dr. Jansen explains, the “browser storage area (e.g., browser cache),” EX1001, 4:60-63, storing those files corresponds to a folder on the user’s local

machine managed by and native to the browser application. EX2027, ¶43. For example, on Windows, Mozilla Firefox stores browser data such as TIFs in a browser-dedicated folder found at

“C:\Users\[username]\AppData\Local\Mozilla\Firefox\Profiles\xxxxxx.default\cache,” while Google Chrome stores this data in a browser-dedicated folder found at “C:\Users\[username]\AppData\Local\Google\Chrome\UserData\Default\Cache.” EX2016, 1; EX2027, ¶43.

Similarly, a user’s browsing history “is located in the user’s profile folder” of the browser application “and the path depends on the browser and operating system.” EX2017, 1; EX2027, ¶44. For example, on Windows, Mozilla Firefox stores browsing history in a browser-dedicated folder located at “C:\Users\\AppData\Roaming\Mozilla\Firefox\Profiles\\AppData\Local\Google\Chrome\UserData\Default.” EX2017, 1-2. Thus, client-side browser storage—e.g., storage areas used to store cookies, TIFs, browsing history, and other local browser data—have long been understood to refer to storage areas managed by and native to a browser application. EX2027, ¶44.

The testimony of Petitioner’s expert, Dr. Wills, is consistent with this understanding. Dr. Wills first confirmed that a “*browser storage area*” does not

encompass any storage area that is “accessible” to a browser. EX2020, 75:22-78:7, 105:18-106:21. For example, when considering folders accessible to a “browser based email application,” such as the “My Documents” folder on a client device, Dr. Wills responded, “that doesn’t seem like that [] is a browser storage area.” *Id.*, 76:8-78:7, 105:18-106:21. Indeed, Dr. Wills explained that not only does the “*browser storage area*” need to be accessible to the browser application, “the browser storage area must be accessible to the web server with which the browser is communicating,” and “[b]rowser software or part of the installation would define where that data is stored on the client device.” *Id.*, 101:20-102:4, 103:18-104:4 (emphasis added); *see also id.*, 42:5-17 (browsers “store different aspects of a browser cache in **standard places as it is configured relative to its installation**”) (emphasis added). Thus, consistent with the previously discussed evidence, Dr. Wills’ testimony supports that a “*browser storage area*” would have been understood as a storage area managed by and native to a browser application on the client device.

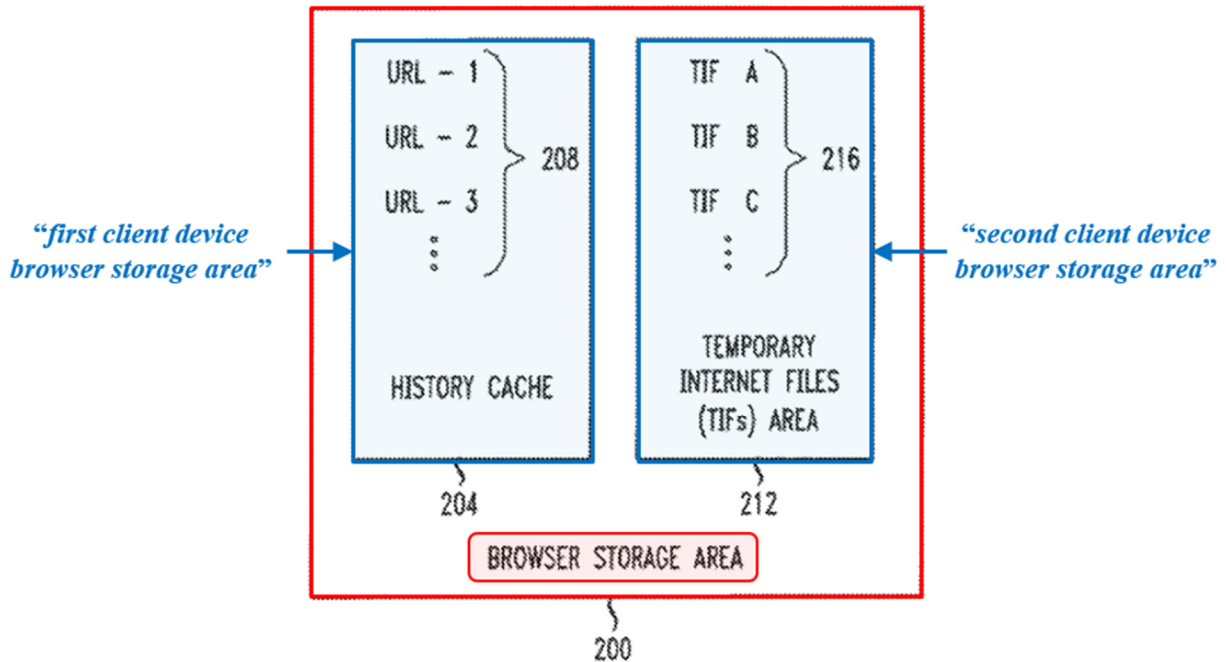
b. This understanding is consistent with the ’823 patent, which provides “a brief background of the **browser storage area 170 and how a browser typically uses its browser storage area 170.**” EX1001, 4:56-58 (emphasis added). At the outset, the ’823 patent expressly states that the “browser storage area” is part of (i.e., native to) the “browser” application—“how a **browser typically uses its**

browser storage area.” *Id.* (emphasis added); EX2027, ¶45. That understanding is also consistent with the patent’s subsequent description.

As noted above, the ’823 patent explains that “[w]hen a browser displays a Web page for the first time, the browser typically downloads one or more files (e.g., images, text, style sheets, etc.) to **its browser storage area (e.g., browser cache).**” *Id.*, 4:60-63 (emphasis added). “The next time the browser visits the same Web page, **the browser determines what is stored in its browser storage area** and displays the local copy of the files rather than downloading the same files again.” *Id.*, 4:66-5:3 (emphasis added); EX2027, ¶45.

Example “*browser storage area[s]*” are described with respect to Figure 2 of the ’823 patent. EX2027, ¶46. Specifically, Figure 2 illustrates “a browser storage area 200” including “**a history cache 204** that contains Uniform Resource Locators (URLs) 208 recently visited by the browser (also called browser history)” and “**a Temporary Internet Files (TIFs) area 212** for storing TIFs 216.” EX1001, 6:4-8, 6:54-56 (emphasis added); *see also id.*, 3:3-4 (“The browser storage area may include a history cache and/or a Temporary Internet Files (TIFs) area.”); EX2027, ¶46.

FIG. 2



EX1001, FIG. 2 (annotated).

“[T]he URLs written to the history cache 204 by the server” and “the TIFs stored in the TIF area 212 of a browser” are “embodiment[s] of cache cookies” stored in browser storage areas. EX1001, 6:39-41, 7:33-34; EX2027, ¶47. And the ’823 patent further explains that these browser storage areas are specifically associated with the user device’s browser: “[O]ne or more servers [] ‘write’ and ‘read’ a cache cookie to and from a **browser storage area associated with a browser** requesting a Web page from the server(s).” EX1001, 2:66-3:3 (emphasis added); EX2027, ¶47.

Based on this description, a POSITA would have understood that when the ’823 patent refers to a “browser storage area (e.g., browser cache),” EX1001, 4:60-

63, that term is referring to storage areas that are managed by and native to the browser application (that is, storage areas created and dedicated for use by the browser), such as those discussed above. EX2027, ¶48. Indeed, the '823 patent describes exactly the type of data stored in these storage areas, such as “Temporary Internet Files (TIFs)” “(e.g., images, text, style sheets, etc.)” and user browsing history that act as “cache cookies” for the browser. EX1001, 4:60-63, 6:54-60, 7:33-34 (“As a result, the TIFs stored in the TIF area 212 of a browser are an embodiment of cache cookies.”). EX2027, ¶48. To interpret the term “*browser storage area*” to include file storage locations that are not associated with a browser application would run counter to the plain and ordinary meaning of the term and effectively read out the word “*browser*” from the claim term. EX2027, ¶¶49-50; *Becton, Dickinson & Co. v. Tyco Healthcare Grp., LP*, 616 F.3d 1249, 1257 (Fed. Cir. 2010) (“Claims must be ‘interpreted with an eye toward giving effect to all terms in the claim.’”) (citing *Bicon, Inc. v. Straumann Co.*, 441 F.3d 945, 950 (Fed. Cir. 2006)).

c. The prosecution history of the '823 patent further illustrates the distinction between a “browser storage area” and other storage areas. During prosecution, the examiner relied on the publication *Local Shared Objects—“Flash Cookies,”* EPIC, July 21, 2005 (EX1014 in this proceeding) “for its disclosure of flash specific cookies stored in an area separate from traditional cookies.” EX1003,

203. Quoting the Flash publication, the examiner observed that Flash cookies are (1) “set through a mechanism in Macromedia’s Flash MX player” and (2) “stored in a special directory depending on the operating system on the client machine,” which for Windows is “C:\Documents and Settings\[username]\Application Data\Macromedia\FIash Player,” for Macintosh OSX is “/Users/[username]/Library/Preferences/Macromedia/FIash Player,” and for GNU-Linux is “~/FIashPlayer.” *Id.*, 217; *see also id.*, 205-06, 208, 212-13. Those directories are not native to or managed by the browser—in fact, those directories do not exist until the Flash Player software is installed on the client device. EX2027, ¶¶51-52.

The applicant subsequently amended claims 8 and 12 to make explicit that the claimed first and second storage areas are “*browser* storage areas.” EX1003, 346, 348. These amendments directly addressed the examiner’s reliance on the Flash publication “for its disclosure of flash specific cookies stored in an area separate from traditional cookies,” supporting the difference between a “browser storage area”—i.e., a storage area managed by and native to a browser application on the client device—and other storage areas on the client device. EX2027, ¶¶52-53.

Accordingly, a “*browser storage area*” would have been understood as a storage area managed by and native to a browser application on the client device.

EX2027, ¶53.

III. OVERVIEW OF THE CITED ART

A. **Hinton describes session cookies for authenticating a user across domains, which are established and valid only during the user’s current session.**

Hinton (EX1005) is directed to “cross-domain log on technologies and technologies which create and manage virtual communities of online users.”

EX1005, ¶7. “Each Internet user is served by a ‘home domain’, which is a domain in which a user is ‘registered’.” *Id.*, ¶9. “[T]he home domain itself may have ‘long term’ relationships with other domains,” such as “e-community domains, where one domain (e.g. the home domain) is responsible for user registration issues.” *Id.*, ¶10; EX2027, ¶55.

Hinton further explains that “[o]ften, a user will access resources in different (‘participating’) domains on behalf of their home domain,” and “the user will have to resubmit to a log in or authentication process as he or she moves from the home domain to another domain.” EX1005, ¶11. Hinton sought to address this issue through “a cross-domain single-sign-on system and method which allows an Internet user to establish a long-term relationship with participating domains, and which gives the user the ability to go directly to participating domains, via

bookmarks or direct URL's for example, without having to go through a home domain first." *Id.*, ¶15; EX2027, ¶56.

Hinton's process requires users to enroll into an "e-community." EX1005, ¶70. "As a result of enrollment into the e-community, a user will have a 'domain identity cookie' ('DIDC') established by each of the participating e-community domains," which provides for "single-sign-on functionality." *Id.* Following establishment of the DIDC, "[i]f a user requests a protected resource in another domain, then authentication information must be transferred across the e-community." *Id.*, ¶129. The DIDC allows the user's home domain to "vouch[] for a user's identity," and "[a]s a result of authentication, the SSO plug-in generates an 'e-Community Cookie' (an eCC or e-community cookie)." *Id.*, ¶¶130, 136; EX2027, ¶57.

"[A] user has one e-community cookie set for each domain at which it has a current, authenticated (or vouched-for) session." EX1005, ¶132. And, importantly, "[a]n eCC is valid for only for [sic] the duration of a browser session, and it is expired when a user invokes logout functionality." *Id.*, ¶249. In other words, Hinton's eCC is established and valid only within a user's current network session—it is never reused in a subsequent session. *Id.*; EX2027, ¶58.

B. Varghese describes capturing device identity information within cookies stored outside of a user’s web browser storage.

Varghese (EX1004) is directed to “providing protection against identity theft over a computer network.” EX1004, 1:16-18. Varghese’s purported “invention includes secure cookies, flash objects and other technologies to recognize and to fingerprint [] from which device a user access an application, whether it is a computer, laptop, mobile device or any other.” *Id.*, 5:64-67. “These user devices thus become additional authentication factors without requiring any change in user behavior” and “[i]nformation concerning these user devices is fingerprinted and stored into a device token or device id for one-time use.” *Id.*, 5:67-6:4; EX2027, ¶59.

Figure 4A of Varghese illustrates “exemplary embodiments of the device fingerprinting process 400 of the system and method of the present invention.” EX1004, 24:33-35; EX2027, ¶60.

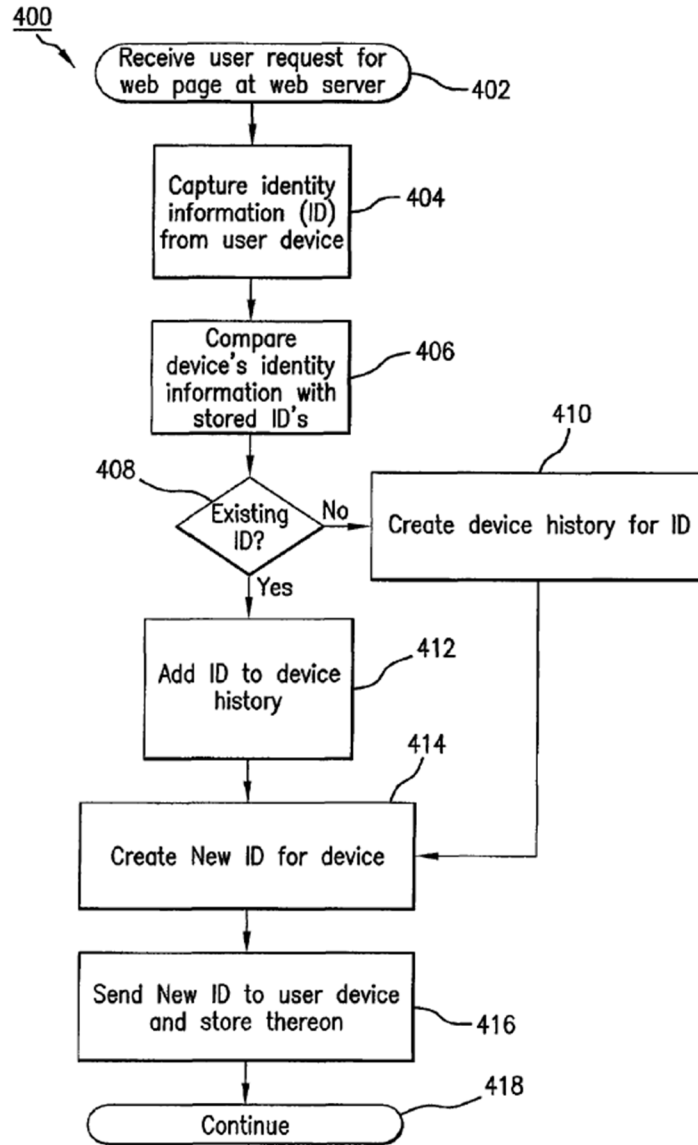


FIG.4A

EX1004, FIG. 4A.

Varghese describes, “[i]n Step 402, a request is received at a service provider server from a user device.” EX1004, 24:40-42. “The fingerprinting process is invoked and information describing the request is transferred.” *Id.*, 24:42-44. Then, “[i]n Step 404, device identity information for the user device is

captured . . . by a client program already resident on the user device,” such as “a web browser.” *Id.*, 24:50-53; EX2027, ¶61.

Varghese explains that captured device information can be stored in the form of “cookies.” EX1004, 25:7-42. “One such technique is known as ‘secure cookies,’” which is “a data packet sent by a web server to a web browser for saving to a file on the host machine,” and that “has been secured against modification or tampering.” *Id.*, 25:22-32. “Another such technique is known as ‘flash cookies’.” *Id.*, 25:33. In this case, “Flash” “software can create local shared objects, known as ‘flash cookies’, for maintaining locally persistent data on a user’s device akin to the standard ‘cookies’ stored by web browsers.” *Id.*, 25:34-43; EX2027, ¶62.

Of note, because flash cookies are stored outside of browser storage in a “special directory” associated with “Flash” software, EX1014, 2, Varghese explains that they “have the advantage [of] not being as easily removed from the user’s device as are standard cookies,” EX1004, 25:40-43; EX2027, ¶63. Nonetheless, a significant disadvantage is that Varghese’s system requires that the end-user install and maintain additional software (Flash software) on the host machine, and that the supplier of the software continue supporting and not discontinue that additional software. *See* EX1004, 25:34-37 (requiring installation of “software applications and/or plug-ins from Macromedia” for use of “flash

cookies”); EX1014, 2 (installation of Macromedia “Flash Player” software required for use of flash cookies); EX2029, 3-4 (same); EX2027, ¶63.

IV. GROUND 1: HINTON DOES NOT RENDER OBVIOUS ANY CLAIM OF THE '823 PATENT.

The Petition fails to establish that Hinton renders obvious “*wherein a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session,*” as recited in independent claims 1 and 6 (elements [1.b.iv] and [6.a.iv]), for two independent reasons. First, the Petition fails to show that Hinton teaches these claim elements because it offers no theory as to how Hinton’s eCC “*was caused to be stored at the client device during a second **previous** network session,*” as required by the claims. Second, even had the Petition addressed the language of elements [1.b.iv] and [6.a.iv], Petitioner could not show that Hinton’s operation renders obvious this claim language.

The Petition also fails to establish that Hinton teaches storage of “*a second cookie . . . during a second previous network session,*” as recited in elements [1.b.iv] and [6.a.iv], occurs prior to receiving the “network resource request” of elements [1.b.i] and [6.a.i], as independent claims 1 and 6 require.

The Petition, therefore, does not establish that Hinton renders obvious any claim of the '823 patent.

A. The Petition fails to address the language of elements [1.b.iv] and [6.a.iv] and, therefore, Petitioner does not meet its burden to establish obviousness of these elements.

Each of independent claims 1 and 6 recites, “*a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session.*” EX1001, 15:47-50, 16:41-44 (emphasis added). The Petition alleges that Hinton’s “domain identity cookie (DIDC)” is the recited “*first cookie*” of the independent claims, and that Hinton’s “e-Community Cookie (eCC)” is the recited “*second cookie.*” Pet., 9, 12, 18. The Petition fails to show that Hinton teaches these claim elements because it offers no theory as to how Hinton’s eCC “*was caused to be stored at the client device during a second previous network session,*” as required by the claims. EX1001, 15:47-50, 16:41-44 (emphasis added). The Petition, therefore, does not meet its burden to establish obviousness of this claim limitation.

In its analysis of this limitation, Petitioner explains that “the eCC cookie is a session cookie type that is a different type of cookie from the DIDC cookie of the persistent cookie type.” Pet., 19. Petitioner continues, “During a session subsequent to the session that created the DIDC, when a user requests access to an affiliated domain, the affiliated domain generates an eCC for that domain and causes the eCC cookie to be stored at the user’s device by sending the eCC cookie to the user.” *Id.* Petitioner then concludes that “Hinton’s system discloses saving

the eCC cookie in a **different** session than the DIDC cookie.” *Id.*, 20 (emphasis added). Noticeably absent from Petitioner’s discussion, however, is *any* discussion of how generation of the eCC occurs in a “**previous network session**,” as the independent claims require. *Id.*, 18-20; EX2027, ¶66. Indeed, the term “*previous*” is not mentioned a single time in Petitioner’s analysis. *Id.*

Petitioner’s failure to address this language of the claims is alone dispositive as to the Hinton grounds. *Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd.*, 821 F.3d 1359, 1369 (Fed. Cir. 2016) (cleaned up) (“It is of the utmost importance that petitioners . . . adhere to the requirement that the initial petition identify ‘with particularity’ the ‘evidence that supports the grounds for the challenge to each claim.’”) (quoting 35 U.S.C. § 312(a)(3)); Petitioner “must ‘specify where each element of the claim is found in the [relied upon] prior art patents.’” *In-Depth Geophysical, Inc. v. ConocoPhillips Co.*, IPR2019-00850, Paper 56, 27 (Sept. 3, 2020) (quoting 37 C.F.R. § 42.104(b)(4)). Petitioner raises no theory identifying how generation of Hinton’s eCC occurs in a “**previous network session**,” and the Board cannot “deviate from the grounds in the petition and raise its own obviousness theory.” *Sirona Dental Sys. GmbH v. Institut Straumann AG*, 892 F.3d 1349, 1356 (Fed. Cir. 2018); *SAS Inst., Inc. v. Iancu*, 584 U.S. 357, 365-367 (2018) (the Petition “define[s] the scope of” an IPR proceeding “all the way from

institution through to conclusion,” and “the Board does not “enjoy[] a license to depart from the petition.”).

Petitioner’s expert does not fill the gaps of the Petition, merely repeating the Petition verbatim with no further reasoning or evidence. *Compare* Pet., 19-20, with EX1003, ¶93; *Xerox Corp. v. Bytemark, Inc.*, IPR2022-00624, Paper 9 at 15 (P.T.A.B. Aug. 24, 2022) (holding that testimony that “merely repeats, *verbatim*, the conclusory assertion for which it is offered to support . . . is entitled to little weight.”); *TQ Delta, LLC v. Cisco Systems, Inc.*, 942 F.3d 1352, 1359 (Fed. Cir. 2019) (“This court’s opinions have repeatedly recognized that conclusory expert testimony is inadequate to support an obviousness determination . . .”).

In its Institution Decision in IPR2024-01316, the Board overlooked Petitioner’s failure address the requirement that Hinton’s eCC be “*stored at the client device during a second **previous** network session.*” (emphasis added). Respectfully, the Board misapprehended Patent Owner’s argument, stating: “The parties dispute turns on whether Hinton teaches generating the DIDC in a first network session that is **different** from a second network session that results in generating the eCC.” IPR2024-01316, DI, 32 (emphasis added). That Hinton’s DIDC and eCC are generated in different sessions, however, is insufficient to meet the claim language. Petitioner must also establish that Hinton’s eCC is generated in a “*second **previous** network session*”—not simply a different network session

subsequent to the “*first network session.*” Petitioner has not made (or attempted to make) that showing.

B. Hinton’s e-community cookie is not “caused to be stored at the client device during a second previous network session.”

In any event, Hinton’s e-community cookie is not “*caused to be stored at the client device during a second previous network session.*” EX1001, 15:47-50, 16:41-44 (emphasis added). That is because Hinton’s e-community cookie is a session cookie that is generated and valid only during a *current* network session—not a “*previous*” one. EX1005, ¶¶137, 249; EX2027, ¶67.

Hinton’s system “allows an Internet user to transfer directly to a domain that is participating in the e-community, by means such as a Bookmark or a directly-typed URL, without the necessity of returning to a home domain prior to transferring to the participating domain.” EX1005, ¶23. “As a result of enrollment into the e-community, a user will have a ‘domain identity cookie’ (‘DIDC’) established by each of the participating e-community domains,” which provides for “single-sign-on functionality.” *Id.*, ¶70; EX2027, ¶68.

Following establishment of the DIDC, “[i]f a user requests a protected resource in another domain, then authentication information must be transferred across the e-community.” EX1005, ¶129. In particular, the Petition alleges that “[d]uring a session subsequent to the session that created the DIDC, when a user requests access to an affiliated domain, the affiliated domain generates an eCC for

that domain and causes the eCC cookie to be stored at the user's device by sending the eCC cookie to the user." Pet., 19-20 (citing EX1005, ¶137). But Petitioner's analysis improperly ignores the term "*previous*" in the claims. *Tyco, LP*, 616 F.3d at 1257 ("Claims must be 'interpreted with an eye toward giving effect to all terms in the claim.'"); EX2027, ¶69.

As Petitioner concedes, "the eCC cookie is a **session cookie**," Pet., 19, which is valid only during the current network session, while the session is active. EX2027, ¶70 (citing EX2030, 1). As Hinton explains, "a user has one e-community cookie set for each domain at which it has a **current, authenticated (or vouched-for) session**." EX1005, ¶132 (emphasis added). The "eCC is valid for only for [sic] the duration of a browser session, and **it is expired** when a user invokes logout functionality." EX1005, ¶249 (emphasis added). Because the eCC is valid only for the user's *current* network session and expires when the session is terminated, the eCC is not "*caused to be stored at the client device during a second previous network session*," as required by the independent claims. EX1001, 15:47-50, 16:41-44 (emphasis added); EX2027, ¶70.

Additionally, in related proceeding IPR2024-01316, the petitioner, Home Depot, submitted an email to the Board alleging that its analysis for element [1.e] shows that Hinton's eCC is established during "*a second previous network session*" because, allegedly, "authentication occurs in subsequent sessions using

the second cookie which was saved during the second previous network session.”

See EX2031. That theory was not presented in the Petition here. And even if it had, Petitioner would be wrong.²

In its Petition, Petitioner alleges, “[w]hen the user requests access to one of the servers in the domain that created the eCC, that server checks whether the user has an eCC to access the DNS domain,” and “[i]f present, this would indicate that the user has a session with a different front-end within the associated domain (106).” Pet., 27 (citing EX1005, ¶154). But Hinton does not create a *new* session when it checks for the eCC, nor does Petitioner allege otherwise. *Id.*; EX2027, ¶71. Rather, Hinton explains that generation of the eCC for a domain establishes “**a single session**” for the domain, “such that the user can engage in e-community actions.” EX1005, ¶126 (emphasis added); EX2027, ¶71. In other words, a user is provided access to any server within the domain when it has a “current, authenticated (or vouched-for) session” for that domain. EX1005, ¶132 (“a user has one e-community cookie set **for each domain at which it has a current, authenticated (or vouched-for) session**”) (emphasis added); EX2027, ¶71.

² Because this new theory was not presented in the Petition, the Board should not consider it. *SAS*, 584 U.S. at 365-367; *Sirona Dental*, 892 F.3d at 1356; 35 U.S.C. § 312(a)(3).

Hinton provides an example of this operation, explaining that the user's "current, authenticated session" provides access to any server within a domain:

As an example, consider a hypothetical site "www.acme.com" that is partitioned so that there is a distinct security policy server set of replicas protecting each of the engineering, accounting, and human resource departments. In this situation, if a user authenticates (or is vouched for) first to engineering, **they will have a domain-wide e-community cookie set** by the engineering security policy server. **When this user then goes to the accounting server, this e-community cookie indicates that the user has a current, authenticated session, and that the accounting server need not re-authenticate the user.**

EX1005, ¶134 (emphasis added).

Put simply, because the user already "has a current, authenticated session" with the domain (as indicated by the eCC) when it attempts to access the domain's accounting server, the user is given access without the need to re-authenticate or establish a new session. EX1005, ¶134; EX2027, ¶¶72-73. Petitioner's expert, Dr. Wills, confirms this operation. Dr. Wills explained that "by setting an E-community cookie for a domain," the user "has a current authenticated session" "[c]orresponding to the domain." EX2020, 110:3-11. In the "acme" example discussed above, "[t]he user has a domain-wide E-community cookie set," and "because the user already has an active session with the domain, **the user can use that session to access the accounting server.**" *Id.*, 112:13-113:9. In other words,

Dr. Wills' testimony confirms that a single session is used across a domain, e.g., the entire acme.com website, allowing access to resources (e.g., servers) within the domain as part of the same session. EX2027, ¶74. Petitioner's untimely analysis of element [1.e] from its email thus still does not establish that the eCC was "*caused to be stored at the client device during a second **previous** network session,*" because subsequent access requests occur as part of the *same* session in which the eCC was established. EX2027, ¶74.

For these reasons, the Petition fails to establish that Hinton renders obvious elements [1.b.iv] and [6.a.iv].

C. Petitioner's challenge further fails because the Petition does not establish that storage of "a second cookie . . . during a second previous network session," as recited in elements [1.b.iv] and [6.a.iv], occurs prior to receiving the "network resource request" of elements [1.b.i] and [6.a.i], as the claims require.

Independent claims 1 and 6 not only require that "*a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session,*" as discussed above. They also require that the recited storage of the "*second cookie . . . during a second previous network session,*" as recited in elements [1.b.iv] and [6.a.iv], have occurred before "*receiv[ing] a network resource request from a client device,*" as recited in elements [1.b.i] and [6.a.i]. EX1001, 15:35-54, 16:30-49; EX2027, ¶75.

Specifically, the claims recite the active step of “*receiv[ing] a network resource request from a client device, wherein the network resource request corresponds to a first cookie of a first type that **was caused to be stored** to the client device during a first previous network session.*” EX1001, 15:35-39, 16:30-34 (emphasis added). This element uses past tense to indicate that the “*first cookie*” was stored during the “*first ... network session.*” It also recites that the “*first ... network session*” is “*previous,*” indicating that the session occurred before receipt of the “*network resource request.*” Thus, the claims require that “*a first cookie of a first type [] was caused to be stored to the client device during a first previous network session*” before “*receiv[ing] a network resource request from a client device.*” *Id.*; EX2027, ¶76.

Within the same indented element, the claims further recite—again using past tense—“*wherein a second cookie of a second type different from the first type **was caused to be stored** at the client device during a second previous network session.*” EX1001, 15:47-50, 16:41-44 (emphasis added). Thus, following the same logic, the claims also require that “*a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session*” before “*receiv[ing] a network resource request from a client device.*” *Id.*; EX2027, ¶77; EX2020, 73:5-74:18 (Petitioner’s expert confirming this interpretation of the claims: “**Q.** [D]oes that mean that the second cookie must

have also been stored during a previous network session that occurred before the claimed network resource request is received? **A.** It does, that is my understanding, yes.”). Petitioner fails to make this showing.

For the recited “*network resource request*” of element [1.b.i], Petitioner contends, “Hinton’s **affiliated domain receives a request from user 100 to access contents of its domain**, which a POSITA would have understood to be a network resource request.” Pet., 12 (citing EX1005, ¶137) (emphasis added). Petitioner then points to this *same* request, citing the *same* paragraph of Hinton, to meet element [1.b.iv]: “[W]hen a user **requests access to an affiliated domain**, the affiliated domain generates an eCC for that domain and causes the eCC cookie to be stored at the user’s device by sending the eCC cookie to the user.” *Id.*, 19 (citing EX1005, ¶137) (emphasis added); EX2027, ¶78.

In other words, Petitioner asserts that the same request (i.e., a request to access the affiliated domain before an eCC has been set for that domain) satisfies both claim elements [1.b.i] and [1.b.iv]. EX2027, ¶79. Because Petitioner relies on the same request for each of these elements, however, Petitioner does not establish that Hinton teaches “*a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session*” before “*receiv[ing] a network resource request from a client device*,” as required by the independent claims. EX2027, ¶79.

Petitioner's expert, Dr. Wills, attempted to amend his theory during deposition in related proceeding IPR2024-01316. Following cross-examination, counsel for the petitioner, Home Depot, in IPR2024-01316 coached Dr. Wills regarding the testimony Dr. Wills would provide on redirect, breaking for approximately an hour between cross-examination and redirect. EX2020, 128:12-131:8. Upon returning, Home Depot's counsel proceeded to ask a series of questions extending beyond the scope of cross-examination. *Id.*, 122:13-15, 123:15-124:1, 124:15-125:1, 127:13-15. Although Dr. Wills would not answer whether he formed his answers "without the assistance of counsel during [the previous] hour," Dr. Wills answered these questions without consulting any documents, and without reviewing any documents during the break. *Id.*, 130:12-131:8, 131:14-22. The extensive break between cross-examination and redirect during which Home Depot's counsel coached Dr. Wills and Dr. Wills' subsequent recitation of canned answers without consulting any documents demonstrate that Dr. Wills was merely a conduit for Home Depot's attorney argument. The Board should not give any weight to his redirect testimony.

Relevant to claim elements [1.b.i] and [1.b.iv], Dr. Wills was asked, "In Hinton what is the disclosure you rely on for" element [1B]. EX2020, 125:22-126:4. Dr. Wills responded: "So, Hinton discloses **multiple affiliated domains**. So, another affiliated domain, a -- **the client's browser generates a network**

resource request to *another* affiliated domain.” *Id.*, 126:5-14 (emphasis added).

In other words, Dr. Wills appears to suggest that the request relied on for element [1.b.i] relates to a *different* affiliated domain than the affiliated domain referenced for element [1.b.iv]. EX2027, ¶80.

At the outset, nowhere does the Petition or Dr. Wills’ declaration present this theory. *See* Pet., 12-14, 18-20; EX1002, ¶¶78-85, 89-93. In fact, Petitioner and Dr. Wills map the claimed “*system*” of claim 1 to “an affiliated domain server”—not multiple affiliated domains. Pet., 10; EX1002, ¶78. Petitioner refers to that particular “affiliated domain” throughout its analysis, never arguing that “the affiliated domain” referenced for element [1.b.iv] is a *different* affiliated domain than the one referenced for every other element. Pet., 18-20 (discussion of element [1.b.iv]); EX1002, ¶¶89-93. The Board, therefore, should not consider this new theory. *SAS*, 584 U.S. at 365-367; *Sirona Dental*, 892 F.3d at 1356; 35 U.S.C. § 312(a)(3).

Moreover, this new theory does not remedy Petitioner’s obviousness challenge. Assuming a user proceeded to a second affiliated domain, Hinton would simply repeat its process for that domain, issuing another e-community cookie corresponding to the domain. EX2027, ¶80. But the user would continue to have an active “current, authenticated session” with the first affiliated domain, unrelated to the second affiliated domain. EX1005, ¶134; EX2027, ¶80. As both sessions would

be current, neither would have occurred prior to the other. EX2027, ¶81. This understanding is consistent with the description in the '823 patent, for example describing “**end[ing] the first network session**” before “a second network session” can be initiated. EX1001, 5:10-32 (emphasis added); EX2027, ¶81. If both sessions were simultaneously active, neither would have occurred before the other. EX2027, ¶81. Accordingly, Dr. Wills’ new theory still fails to meet the claim language.

V. GROUNDS 2 AND 3: VARGHESE, ALONE OR IN COMBINATION WITH HINTON, DOES NOT RENDER OBVIOUS ANY CLAIM OF THE '823 PATENT.

The Petition fails to establish that Varghese renders obvious at least the following elements of independent claims 1 and 6: (1) “*wherein the first cookie of the first type is stored in a first client device browser storage area and the second cookie of the second type is stored in a second client device browser storage area different from the first client device browser storage area*” (elements [1.b.v] and [6.a.v]); and (2) “*wherein a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session*” (elements [1.b.iv] and [6.a.iv]).

The Petition, therefore, does not establish that Varghese renders obvious any claim of the '823 patent.

A. The Petition fails to establish that Varghese renders obvious elements [1.b.v] and [6.a.v] (“wherein the first cookie of the first

type is stored in a first client device browser storage area and the second cookie of the second type is stored in a second client device browser storage area different from the first client device browser storage area”).

Each independent claim recites two storage areas that are specific to the client device browser: “*a first client device browser storage area*” and “*a second client device browser storage area*.” EX1001, 15:50-54, 16:44-49. The Petition fails to establish that Varghese teaches this element because Varghese’s “flash cookie”—relied on for the claimed “*second cookie*,” Pet., 48-49—is not stored in a “*browser storage area*,” as required by the independent claims. EX2027, ¶82.

Specifically, Petitioner alleges that “Varghese discloses the use of two cookies – secure cookies and flash cookies.” Pet., 36. Petitioner equates Varghese’s “secure cookie” to the recited “*first cookie*” and Varghese’s “flash cookie” to the recited “*second cookie*.” *Id.*, 48-49. But, as demonstrated by Petitioner’s own evidence, Varghese’s flash cookies are not stored in a “*browser storage area*,” as required by the independent claims. EX2027, ¶83.

A POSITA would have understood the claimed “*browser storage area*” to refer to a storage area managed by and native to a browser application on the client device. *See* Section II.C.1; EX2027, ¶84. Varghese’s flash cookies are not stored in a storage area managed by and native to any browser application on the client device. As Petitioner’s own expert explains, “[s]ecure cookies are standard cookies known to be saved by the Web browser in browser-maintained files,” but, “[i]n

contrast, Flash cookies are stored in a separate storage area.” EX1002, ¶168

(emphasis added); EX2027, ¶84. Petitioner’s expert further explains that flash cookies not stored in local browser storage at all; rather, they “are stored on the client machine as Local Shared Objects: ‘Windows C:\Documents and Settings\[username]\Application Data\Macromedia\Flash Player; Macintosh OSX /Users/[username]/Library/Preferences/Macromedia/Flash Player; GNU-Linux ~/.macromedia.’” EX1002, ¶168 (citing EX1014, 2).

In other words, Varghese’s flash cookies are stored in a “special directory” on the user’s hard disk that is associated with Macromedia Flash Player software—not with the user’s Web browser. EX1014, 2; EX2028, 1 (“**Instead of using the browser’s local storage system, though, it has its own.**”) (emphasis added); EX2027, ¶85. As Dr. Jansen explains, the directory storing flash cookies is neither managed by nor native to the client’s browser application. EX2027, ¶¶85-86; EX2018, 12 (“Shared object files are used by Flash Player to store data locally. For example, a developer may create a game application that stores information on high scores.”). In fact, Adobe’s Flash Player Administration Guide from the timeframe of the ’823 patent expressly states that “users of other applications outside Flash Player, such as **a web browser, cannot use those applications to access the data**” in Flash Player’s local shared objects folder. EX2018, 12; EX2026, 6 (providing the same access limitation in another version of the Flash

Player Administration Guide downloaded from Adobe’s website); EX2027, ¶86; *see also* EX2019, 2 (“**the information stored by Flash Player is not the same as a cookie; it is used only by the application that runs in Flash Player**, and has no relation to any other Internet privacy or security settings you may have set in your browser.”) (emphasis added). Thus, not only are flash cookies stored in a folder that is not managed by the client’s browser application, Adobe’s Flash Player documentation indicates that they are not even accessible to the browser application. EX2027, ¶86.

Indeed, Petitioner’s expert explained that to be considered a “*browser storage area*,” the “[b]rowser software or part of the installation would define where that data is stored on the client device.” EX2020, 103:18-104:4 (emphasis added). And Petitioner’s own evidence confirms that “Flash cookies are set through a mechanism in Macromedia’s Flash MX player,” stored in a “special directory” defined by the Flash software’s installation—*not* by the browser application. EX1014, 1. Thus, the storage area where Flash cookies are stored is not a “*browser storage area*” because it is neither managed by nor native to a browser application; rather it is independently managed by and native to the Flash software installed on the client device. EX1014, 1-2. EX2027, ¶87.

Moreover, Petitioner provides no reason to store Varghese’s flash cookies in a “*browser storage area*,” merely contending that “it was obvious to a POSITA to

organize different types of files in different folders.” Pet., 48-49; EX2027, ¶88.

That vague analysis improperly reads the word “*browser*” out of the term “*browser storage area*.” *Tyco*, 616 F.3d at 1257 (Fed. Cir. 2010). Indeed, Varghese explains that because flash cookies are *not* stored in a “*browser storage area*,” they “have the advantage [of] not being as easily removed from the user’s device as are standard cookies.” EX1004, 25:40-43; EX2027, ¶88. That is—consistent with Petitioner’s own expert’s testimony—flash cookies are specifically designed to persist *outside* of browser storage. *See* EX2028, 2 (“[T]he Flash plug-in is able to store data locally just like your browser does, **but in a different location on your hard drive**. ‘And since Flash is an add-on component,’ . . . ‘**built-in browser controls over standard cookies don’t apply to Flash cookies.**’”) (emphasis added); EX2029, 2 (same), 3 (“**Because these cookies are stored outside the browser** you cannot protect yourself by using a different browser”) (emphasis added); EX1002, ¶168; EX2027, ¶88.

Thus, storing Varghese’s flash cookies in a “*browser storage area*” would not only run contrary to Varghese’s express teachings, but to the purpose of flash cookies themselves. EX2027, ¶90. Varghese’s express teachings thus demonstrate that a POSITA would not have been motivated to store flash cookies in a “*browser storage area*,” as the term would have been properly understood. *See* Section II.C.1; EX2027, ¶90.

On the other hand, the '823 patent's use of browser storage areas provides a significant advantage over flash cookies. Specifically, the patented invention does not require end-users to separately install and maintain Flash software on their host machines. EX1004, 25:34-37; EX1014, 2; EX2029, 3-4; EX2027, ¶89. Nor does the '823 patent rely on an additional third party to continue supporting (and not discontinue) Flash software in order to store flash cookies. EX1004, 25:34-37; EX1014, 2; EX2029, 3-4; EX2027, ¶89.

Accordingly, because Varghese's flash cookies are not "*stored in a second client device browser storage area,*" Varghese does not render obvious elements [1.b.v] and [6.a.v].

B. The Petition fails to establish that Varghese renders obvious elements [1.b.iv] and [6.a.iv] ("wherein a second cookie of a second type different from the first type was caused to be stored at the client device during a second previous network session").

Each of independent claims 1 and 6 requires that the "*first cookie . . . was caused to be stored to the client device during a first previous network session,*" and the "*second cookie . . . was caused to be stored at the client device during a second previous network session.*" EX1001, 15:36-50, 16:31-44 (emphasis added). As noted in Section V.A, Petitioner equates Varghese's "secure cookie" to the recited "*first cookie*" and Varghese's "flash cookie" to the recited "*second cookie.*" Pet., 48-49. But the Petition fails to establish that Varghese teaches storing these

cookies during two different “*previous network session[s]*,” as required by the independent claims.

To meet this limitation, Petitioner asserts that Varghese “discloses that secure cookies can be removed from the browser’s memory when the user clears the browser’s cookies,” and “that the cookies are routinely replaced with each login (*i.e.*, new network session).” Pet., 46-47 (citing EX1004, 26:13-14). From these statements, Petitioner concludes, “[t]hus, a POSITA would have understood and found obvious that Varghese discloses a scenario in which the flash cookie was created and stored in a second previous session than the secure cookie (e.g., the user cleared the browser’s cookies).” *Id.*, 47; EX2027, ¶¶91-92.

Varghese, however, provides no support for Petitioner’s hypothetical scenario. Indeed, the passage of Varghese relied on by Petitioner supports the opposite conclusion—that *both* secure cookies and flash cookies would be replaced upon each login, regardless of whether a user cleared the browser’s secure cookies. EX2027, ¶93.

As part of Varghese’s “device fingerprinting process,” Varghese explains that “a new Device ID token is created for the device,” which is “sent to the user device and stored thereon, e.g., **as a standard cookie or as a flash cookie.**” EX1004, 24:33-35, 26:4-11 (emphasis added). Referring to these cookies, Varghese further explains that “[a] feature of the invention relates to the

replacement of the cookie on the user’s machine upon each login” so that “stolen fingerprints, tokens, or device ids cannot be fraudulently reused.” *Id.*, 26:13-29 (emphasis added). In other words, Varghese discloses replacing cookies—including *both* secure cookies and flash cookies—upon each login. EX2027, ¶94.

Indeed, Petitioner’s expert agreed that Varghese’s “replacement [] of cookies upon login [is] due to security concerns.” EX2020, 119:19-22. And when asked, “So based on those security concerns, would it then make sense to you for Varghese to replace whatever previous cookies existed,” Dr. Wills responded, “I think that that would be the most likely, that it would be replacing both of them.” *Id.*, 120:1-8. Thus, in Petitioner’s hypothetical scenario in which a user cleared the browser’s cookies, a POSITA would have understood that Varghese’s system would still replace *both* the secure cookie and flash cookie on the next login (i.e., during the *same* session). EX2027, ¶¶94-95.

In its Institution Decision in IPR2024-01316, the Board preliminarily disagreed that Varghese “teaches replacing both the secure cookies and flash cookies upon each login (i.e., during the same network session)” because, according to the Board, “Varghese does not clarify what type of cookie it is referring to in this particular context.” IPR2024-01316, DI, 49. But Varghese explains the purpose of replacing these cookies upon each login, stating: “This

provides further security so that even if a user's machine information is improperly acquired by a third party, **even including that embodied in a previous cookie, the authentication system can identify that the user is not authorized and deny access to the system.**" EX1004, 26:14-19; EX2027, ¶95. In other words, regardless of the type of cookie (e.g., secure cookie or flash cookie), Varghese's techniques are designed to prevent unauthorized users by replacing the content of each cookie upon each login. EX2027, ¶95.

Petitioner's citation to Varghese's Table 8 does not salvage its argument. Petitioner contends, with no explanation, "Varghese contemplates the cookies being created in different network sessions when only one of the secured cookie or the flash cookie is present." Pet., 47 (citing EX1004, Table 8); EX1002, ¶166 (repeating Petition verbatim). Nowhere does Varghese describe or suggest this operation. EX2027, ¶96.

Varghese's Table 8 merely depicts "exemplary security evaluation in the form of decision tables." EX1004, 19:28-30, Table 8. For example, Varghese explains how security decisions are made based on the existence of certain data, such as when a flash cookie exists but a secure cookie does not. *Id.*, 19:55-20:10, Table 8. Although Table 8 does provide an example scenario in which a flash cookie is present and a secure cookie is missing, nowhere does Varghese teach that a secure cookie or flash cookie is created upon performing a security check. *Id.*;

EX2027, ¶96. Rather, “[i]f the retrieved data tokens that were previously stored on a device by this invention, **e.g., a secure cookie, a Flash cookie, or Flash data, are not present, a further pattern check is performed,**” which “examines the particulars of the pattern of the location and device criteria and assigns an appropriate score.” EX1004, 19:62-67 (emphasis added). In other words, when a cookie is missing (e.g., if a user were to clear the browser’s cache), Varghese simply assesses other device characteristics to make a security decision. EX2027, ¶96. Varghese’s description with respect to Table 8 in no way supports that Varghese’s secure cookie and flash cookie would be created in two different “*previous network session[s]*,” as required by the independent claims. EX2027, ¶96.

To be clear, although the Board is correct that Table 8 demonstrates a scenario in which “a flash cookie is present, but a secure cookie is absent,” IPR2024-01316, DI, 50, as noted above, that is not enough to render obvious the claim language. Neither Petitioner nor its expert identifies any teaching in Varghese indicating that a secure cookie would be created in this scenario, outside of a new “login.” Pet., 46-48; EX1002, ¶¶165-66. And even then, neither Petitioner nor its expert identifies which cookies would be replaced upon login. Pet., 46-48; EX1002, ¶166. Nor does Petitioner or its expert provide any reason why a POSITA would modify Varghese to include this operation, especially when flash cookies

and secure cookies are used for the same purpose of identifying a user device. Pet., 46-48; EX1002, ¶166; EX1004, 24:23-27; EX2027, ¶94. The Board should not, and cannot, fill these gaps in Petitioner’s analysis. *In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1381 (Fed. Cir. 2016) (the Board is not “free to adopt arguments on behalf of petitioners that could have been, but were not, raised by the petitioner during [a post-grant proceeding]”); *In-Depth Geophysical*, IPR2019-00850, Paper 56, 27; *Sirona Dental*, 892 F.3d at 1356; *SAS*, 584 U.S. at 365-367.

Accordingly, the Petition fails to establish that Varghese renders obvious elements [1.b.iv] and [6.a.iv].

VI. CONCLUSION

For at least the foregoing reasons, Petitioner has failed to show a reasonable likelihood of prevailing with respect to any challenged claim. Therefore, the Board should deny institution.

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX PLLC

/Steven M. Pappas/

Steven M. Pappas
Registration No. 73,904
Attorney for Patent Owner

Date: July 18, 2025

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 Preliminary Response complies with the type-volume limitation of 14,000 words, comprising 9,513 words, excluding the parts exempted by 37 C.F.R. § 42.24(a)(1).

2. This Patent Owner Preliminary 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

/Steven M. Pappas/

Steven M. Pappas
Registration No. 73,904
Attorney for Patent Owner

Date: July 18, 2025

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 PRELIMINARY RESPONSE UNDER 37 C.F.R. § 42.107(a)** and associated Exhibits 2011-2032 were served in their entireties on July 18, 2025, upon the following parties via electronic mail:

Nathaniel St. Clair, II (Lead Counsel)
Blake T. Dietrich (Back-up Counsel)
Leisa Talbert Peschel (Back-up Counsel)
JACKSON WALKER LLP
nstclair@jw.com
bdietrich@jw.com
lpeschel@jw.com

Respectfully submitted,

STERNE, KESSLER, GOLDSTEIN & FOX PLLC

/Steven M. Pappas/

Steven M. Pappas
Registration No. 73,904
Attorney for Patent Owner

Date: July 18, 2025

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