

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

NETFLIX, INC.,

Petitioner

v.

VL COLLECTIVE IP LLC,

Patent Owner

IPR2023-00630
U.S. Patent 7,440,559

PATENT OWNER'S RESPONSE

TABLE OF CONTENTS

	<u>Page</u>
I. INTRODUCTION	1
II. BACKGROUND	2
A. Overview of the '559 Patent (EX1001)	2
B. The Claims of the '559 Patent.....	7
C. Overview of Cassin (EX1004)	9
D. Overview of Huston (EX1005)	15
III. PRIORITY DATE OF THE '559 PATENT	17
IV. LEVEL OF ORDINARY SKILL	17
V. RELATED MATTERS	17
VI. CLAIM CONSTRUCTION	18
VII. ASSERTED GROUNDS.....	18
VIII. PETITIONER HAS NOT SHOWN THAT ANY CHALLENGED CLAIMS ARE UNPATENTABLE	19
A. Ground 1: Cassin Does Not Anticipate Claims 1-14	19
1. Cassin Does not Disclose a Response to the Content Status that Instructs the Terminal or Controller to Perform One or More Actions to Thereby Control the Flow of Content to the Terminal Based Upon the Terminal Status Information, as Required by Claims 1, 7, 13, and 19.....	19
2. Cassin Does Not Disclose Receiving Server Status Information that Comprises a Listing of At Least One Piece of Content Available from the Source	37

3.	Cassin Does Not Disclose Terminal Status Information Comprising a Listing of At Least One Piece of Content Stored in the Memory	39
B.	Ground 2: Cassin and Huston Do Not Render Obvious Any Claim of the '559 Patent.....	43
1.	Cassin and Huston Do Not Teach Or Suggest a Response to the Content Status that Instructs the Terminal or Controller to Perform One or More Actions to Thereby Control the Flow of Content to the Terminal Based Upon the Terminal Status Information, as Required by Claims 1, 7, 13, and 19.....	43
2.	Cassin and Huston Do Not Teach Or Suggest Server Status Information that Comprises a Listing of At Least One Piece of Content Available from the Source.....	44
3.	Cassin and Huston Do Not Teach Or Suggest Terminal Status Information Comprising a Listing of At Least One Piece of Content Stored in the Memory.....	44
4.	Cassin and Huston Do Not Teach Or Suggest a Response to the Content Status that Instructs the Terminal to Delete at Least One Piece of Content from the Memory of the Terminal Based Upon the Listing of At Least One Piece of Content Stored in the Memory of the Terminal	47
C.	Ground 3: Huston Does Not Render Obvious Any Claim of the '559 Patent.....	51
1.	Huston Does Not Disclose Each Element of Claims 1, 7, 13, 19.....	51
2.	Huston Does Not Disclose Each Element of Claims 2, 3, 5, 8, 9, 11, 14, 15, 17, 20, 21, 23.	53
IX.	CONCLUSION.....	53

TABLE OF AUTHORITIES

	Page(s)
Cases	
<i>Adidas AG v. Nike, Inc.</i> , 963 F.3d 1355, 1359 (Fed. Cir. 2020).....	50
<i>Arctic Cat Inc. v. Bombardier Recreational Prod. Inc.</i> , 876 F.3d 1350, 1360 (Fed. Cir. 2017).....	49
<i>Chamberlain Grp., Inc. v. Techtronic Indus. Co.</i> , 935 F.3d 1341, 1350 (Fed. Cir. 2019)	24
<i>In re Cyclobenzaprine Hydrochloride Extended–Release Capsule Patent Litig.</i> , 676 F.3d 1063, 1068 (Fed. Cir. 2012).....	50
<i>In re Gurley</i> , 27 F.3d 551 (Fed. Cir. 1994).....	49
<i>Kennametal, Inc. v. Ingersoll Cutting Tool Co.</i> , 780 F.3d 1376 (Fed. Cir. 2015)	24
<i>Masimo Corporation, v. Apple Inc.</i> , IPR2023-00807, Paper 10 at 20-21 (November 21, 2023).....	50
<i>Microsoft Corp. v. Biscotti, Inc.</i> , 878 F.3d 1052, 1068, 1074-75 (Fed. Cir. 2017)	24
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303, 1312–13 (Fed. Cir. 2005) (en banc)	18
<i>Sensonics Inc. v. Aerosonic Corp.</i> , 81 F.3d 1566, 1570 (Fed. Cir. 1996)	45
<i>W.L. Gore & Assoc. v. Garlock, Inc.</i> , 721 F.2d 1540, 1553 (Fed. Cir. 1983).....	45
Regulations	
37 C.F.R. § 42.24	54

PATENT OWNER'S EXHIBIT LIST

Exhibit	Description
2001	Declaration of Jaime F. Cardenas-Navia in Support of Motion for Admission <i>Pro Hac Vice</i>
2002	Expert Declaration of Dr. Michael Goodrich
2003	Deposition Transcript of Anthony Wechselberger

VideoLabs, Inc. (“Patent Owner”) submits this Response to the Petition of Netflix, Inc. (“Petitioner”) seeking *inter partes* review (“IPR”) of U.S. Patent No. 7,440,559 (the “’559 patent”). The Response is supported by the Expert Declaration of Dr. Michael Goodrich (EX2002). EX2002, ¶¶1-137.

I. INTRODUCTION

The Petition fails to show that any of the challenged claims are unpatentable. One of the key innovations of the ’559 patent is that the server controls the flow of content to the terminal not by taking action directly, but by instructing the terminal to take an action based on information from the terminal. For example, instead of the server sending content to the terminal, the server instructs the terminal to go and get the content. This results in a hybrid “push/pull” system, which provides the benefits of the terminal taking the action (which is characteristic of a “pull” system) while still allowing the server to retain control (which is characteristic of a “push” system).

Petitioner’s references, Cassin and Huston, take fundamentally different approaches. Cassin is directed to automatically downloading content to a client computer system and discloses a system in which the server itself sends the content to the terminal. Cassin never discloses the server sending *instructions* to the terminal, and that only makes sense—the server has already sent the content, it has

no need to send instructions for the terminal to do anything. This central disconnect means that Cassin does not disclose or suggest many elements of the '559 claims.

Huston is even more off base. Huston is directed to maintaining version control of files. Huston discloses a “differencing engine” that refreshes content on traffic servers when new content is available from an origin server. In Huston, the “differencing engine” refreshes the content when there is new content from the origin—not based on any information from the terminal. The “differencing engine” does not receive any information from the traffic servers (the alleged “terminals”) at all. Huston has little to do with the claims of the '559 patent.

II. BACKGROUND

A. Overview of the '559 Patent (EX1001)

U.S. Patent No. 7,440,559 was filed October 22, 2003 and issued October 21, 2008. It lists Ahti Muhonen, Antti-Pentti Vainio, and Ari Hännikäinen as inventors. EX2002, ¶36.

At the time of the priority date of the '559 patent, mobile networks allowed for transmitting audio and visual data over wireless networks. *See* EX1001, 1:15-2:53. However, at that time, downloading large amounts of data was difficult and time-consuming. *See id.*, 1:28-57. Additionally, video content was increasingly available for viewing over the Internet on a variety of computing devices. *See id.*, 1:15-2:53. In content delivery systems of the time, the server could not control the

flow of content because previous systems were “pull” systems. *Id.*, 2:25-39, 10:45-59. In a “pull” system, only the terminal could take action (to “pull” the content by requesting it). *Id.*, 2:25-39, 10:45-59. Thus, conventional technologies suffered from “from inefficient content flow control between the mobile terminal and the server or content provider. More particularly, current techniques for downloading content can suffer from inefficient control of content received and thereafter stored by mobile terminals, as well as inefficient control of content stored by mobile terminals.” *Id.*, 2:48-53; EX2002, ¶37.

The ’559 patent solved the deficiencies of conventional technologies by adding a separate network entity—the “content flow manager”—to provide ***instructions*** to the terminal based on terminal status and server information. EX1001, Abstract, 2:62-65, 3:10-51, Cl. 1. In the invention, the network entity sends the instruction, but it is the terminal that ultimately takes the “action” to control the flow of content. *Id.*, 2:65-3:1, 11:3-5; *see also* Figs. 1, 3. In this way, the ’559 patent achieves many of the benefits of a push system, while operating within the confines of the internet’s pull framework by using a “hybrid” pull technique. The invention also allows the server to take advantage of the fact that “user preferences, capabilities of the mobile terminal and/or previous contents stored or otherwise received by the mobile terminal can have an [e]ffect on the flow of new content to the mobile terminal.” *Id.*, 2:40-44; EX2002, ¶38.

In detail, the '559 patent provides “an improved system and associated terminal, method and computer program product for controlling the flow of content, particularly to mobile terminals that are operable with mobile telecommunication and digital broadcast networks.” EX1001, 2:58-62. The '559 patent illustrates an exemplary terminal and system in Fig. 1 (reproduced below).

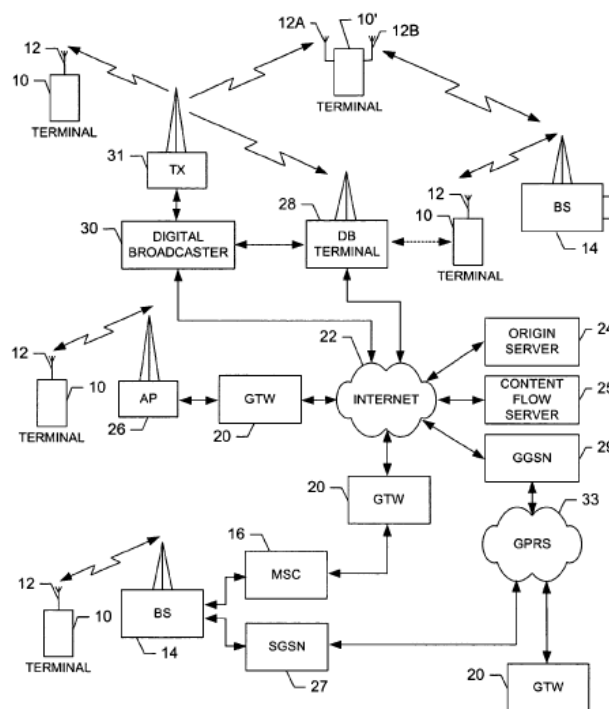
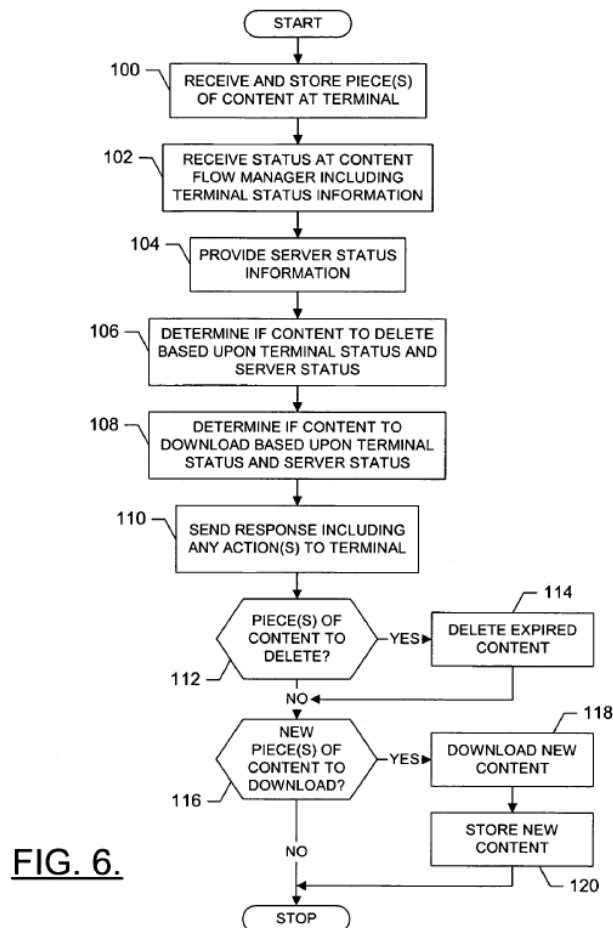


FIG. 1.

Terminal 10 may be a mobile station such as portable digital assistants (PDAs), pagers, laptop computers and other types of voice and text communications systems, etc. *See id.*, 9:3-15. Terminal 10 may be communicatively coupled to digital broadcast (or broadband) receiver 28. *See id.*, 10:45-53. Digital broadcast

receiver 28 may include a content flow manager, which controls the flow of content from origin server 24 and/or digital broadcaster 30 to terminal 10. *See id.*, 10:5-9. Terminal 10 is configured to receive the content from digital broadcast receiver 28 and store the content in memory (e.g., volatile memory 94 or non-volatile memory 96 as shown in Fig. 5). *See id.*, 10:31-35, 11:54-65; EX2002, ¶¶39-40.

Fig. 6 of the '559 patent (reproduced below) illustrates the process of controlling the flow of content.



EX1001, Fig. 6.

As shown in Fig. 6, a terminal receives and stores content. *See id.*, 11:54-61. The terminal may receive audio/video content from a content flow manager. *See id.*, 8:10-24, 7:1-11. The terminal can store the content in the terminal's non-volatile memory. *See id.*, 11:61-64. After the terminal stores the content in the non-volatile memory, the terminal may transmit a content status to the content flow manager. *See id.*, 12:6-10. The content status may request one or more pieces of content from the content flow manager. *See id.* Additionally, the content status may include terminal status information. *See id.* The terminal status information may include a listing of the pieces of content stored in the terminal's memory (e.g., the non-volatile memory). *See id.*, 12:22-30. In Fig. 6 step 104, the content flow manager may receive a server status which indicates which content may be delivered to the terminal from the origin server that stores the content. *See id.*, 12:31-47; EX2002, ¶¶41-42.

The content flow manager controls the flow of content to the terminal based on the terminal and server status. *See id.*, 13:38-50. Based on the terminal status, which includes a listing of content stored in the terminal's memory, the content flow manager may determine whether content needs to be deleted from the terminal's memory. *See id.* Additionally, based on the server status, which indicates which content may be delivered to the terminal, the content flow manager may determine which pieces of content are available to be delivered to the terminal. *See id.*, 13:51-

64. Based on the receiving terminal status, the content flow manager may transmit a response to the terminal. *See id.*, 13:38-61. The response may include instructions generated based on the terminal and server status. *See id.* That is, if the content flow manager determines that there is content in the terminal's memory that should be deleted, the response includes instructions to delete content from the terminal's memory. *See id.* Alternatively or in addition to, the terminal may include instructions to download additional content. *See id.*; EX2002, ¶43.

B. The Claims of the '559 Patent

The '559 patent claims novel apparatuses, methods, and non-transitory computer-readable media that simplify how to control the flow of content allowing for an efficient download of large amounts of content. EX2002, ¶44. Exemplary apparatus claim 1 and method claim 13 are shown below:

Claim 1

1[pre] An apparatus comprising:

1[a][1] a processor configured to receive, from a terminal located remote from the apparatus, a content status including terminal status information, and

1[a][2] configured to receive server status information regarding a source of content, wherein the server status information comprises a listing of at least one piece of content available from the source,

1[a][3] wherein the processor is configured to send, to the terminal, a response to the content status that instructs the terminal to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information and the server status information, and **1[b]** wherein the at least one piece of content available from the source, and the content for which the processor is configured to control the flow, comprise multimedia content.

Claim 13

13[pre] A method for controlling a flow of content, the method comprising:

13[a] receiving, at a network entity from a terminal located remote therefrom, a content status including terminal status information comprising a listing of at least one piece of content stored in a memory of the terminal; and

13[b] sending, from the network entity to the terminal, a response to the content status that instructs the terminal to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information,

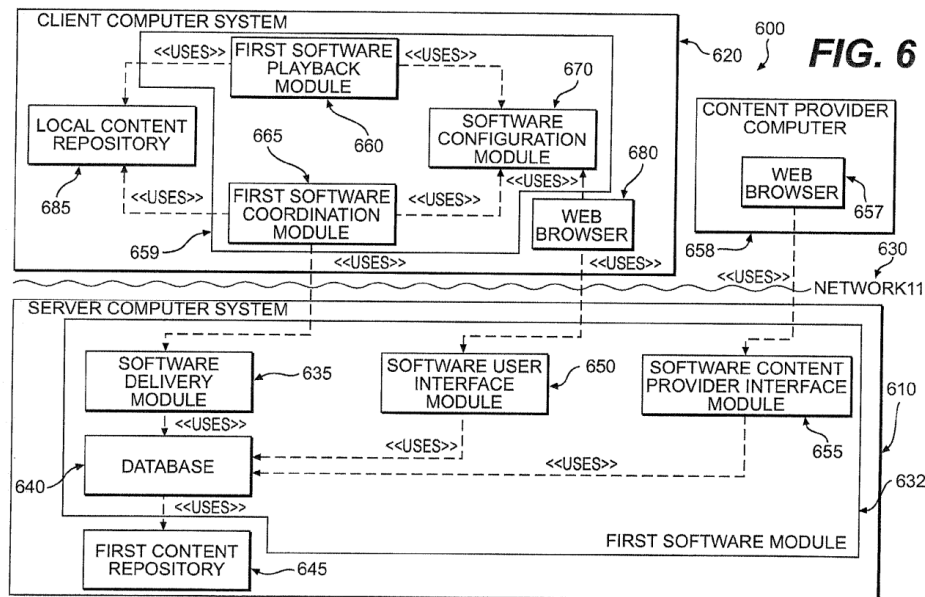
13[c] wherein the at least one piece of content stored in the memory of the terminal, and the content for which the flow is controlled, comprise multimedia content.

EX1001, cls. 1, 13.

C. Overview of Cassin (EX1004)

Cassin is U.S. Patent Application Publication No. 2003/0023427, entitled “Devices, Methods And A System For Implementing A Media Content Delivery And Playback Scheme,” which was filed July 26, 2001. EX2002, ¶45.

Cassin is directed to automatically downloading content from a server computer system to a client computer system. See EX1004, ¶¶0140-0143. Cassin’s Fig. 6 (reproduced below) illustrates an exemplary system.



EX1004, Fig. 6. As shown in Fig. 6, the first device may be a server computer system 610, communicatively coupled to a content provider computer 658, and the second device may be a client computer system 620. *See id.*, Fig. 6. The server computer system 610 may deliver content to client computer system. *See id.*, ¶¶0145-0146. First content repository 645 may store the media files (e.g., the content). *See id.*, ¶0146. Software delivery module 635 may access the media files in first content repository 645 via database 640. *See id.* Software delivery module 635 may deliver the media files in first content repository 645 to client computer system 620. *See id.*, ¶¶0140-0146; EX2002, ¶¶46-47.

Notably, unlike in the '559 patent, Cassin discloses that the server computer sends content to the client computer. EX1004, ¶¶0162-0168, Figs. 8, 9; *see also id.*, ¶¶0012 (“The device delivers the media content to the remote device at the second predetermined time.”), 0050 (“According to another aspect of the ninth embodiment, the device delivers the media content.”), 0103, 0108 (“The first device 110 delivers the media content to the remote device at the second predetermined time.”), 0130, cl. 2 (“The device defined in claim 1, wherein said device delivers the media content”), cls. 13, 104, 118. The server computer itself takes action to control the flow of content instead of sending instructions to the client for it to take the action. *Id.*, ¶¶0012, 0050, 0103, 0130, cls. 2, 13, 104, 118; EX2002, ¶48.

Also, in sharp contrast to the '559 patent, Cassin teaches away from streaming content. Cassin encourages client computer systems automatically downloading the content for playback. For example, Cassin states “[b]y storing the media content on the client computer (i.e., a remote device), which is local to the user of the client computer, *the system 600 avoids the inconsistent quality that may result from streaming media content over the network 630.*” EX1004, ¶0143 (emphasis added); EX2002, ¶49.

Cassin discloses two discrete and separate protocols. Cassin’s Fig. 8 (shown below) illustrates a *first* protocol for communications between server computer system 610 and client computer system 620. *See* EX1004, ¶0164. In the first protocol, client computer system 620 sends user information to the server computer system 610. *See id.* Server computer system 610 queries database 640 to retrieve a list of media content that the user is entitled based on the user information. *See id.* Server computer system 610 sends the list to client computer system 620. *See id.* Server computer system 610 attempts to send media content from the list to client computer system 620. *See id.* Client computer system 620 determines if the media content is already stored in the client computer 620’s memory. *See id.* If so, client computer system 620 provides an indication to server computer system 610 that it already has the media content. *See id.* If not, client computer system 620 accepts delivery of the media content. *See id.*, ¶¶0164-0165; EX2002, ¶50.

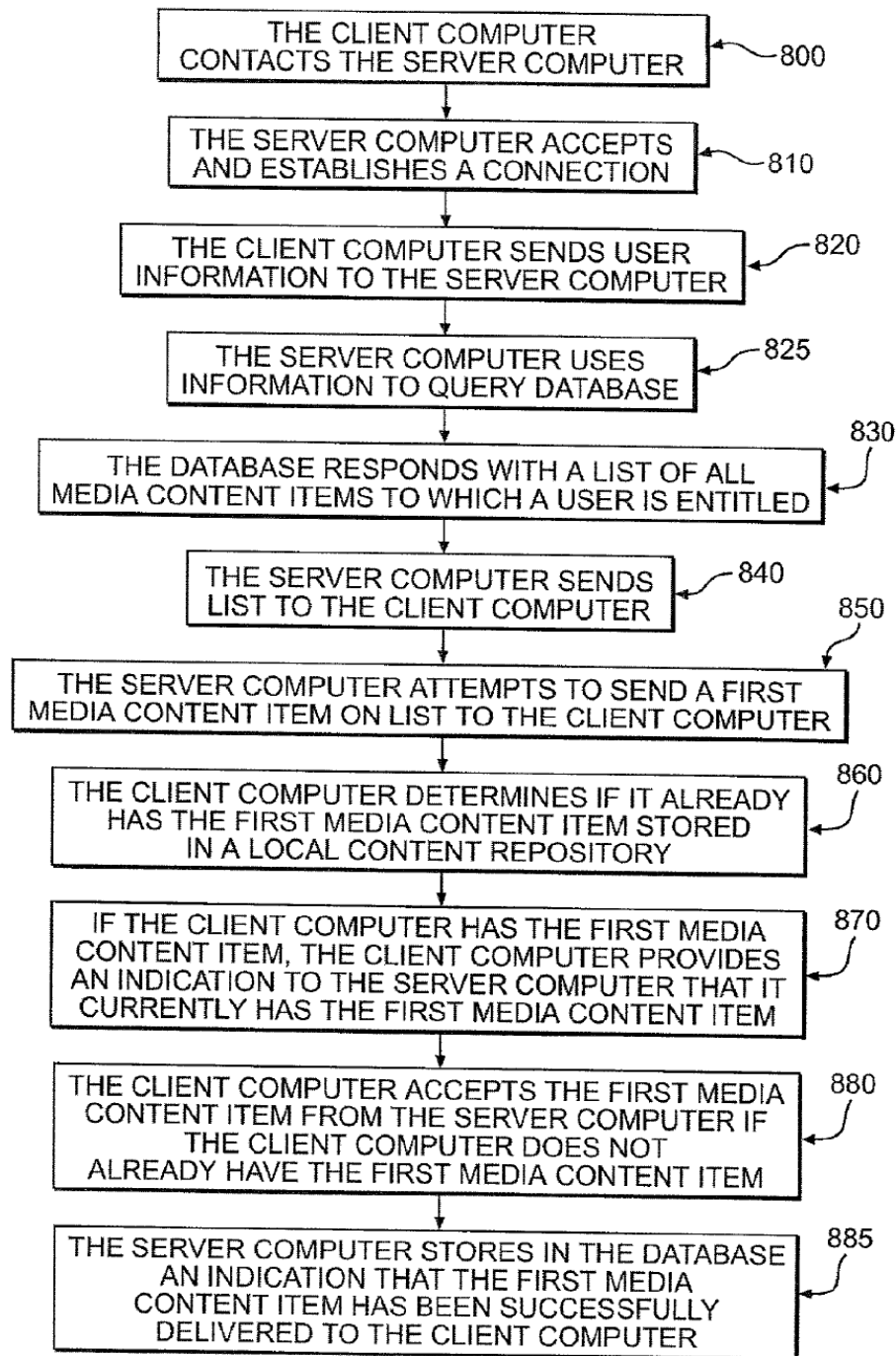


FIG. 8

EX1004, Fig. 8.

Cassin's Fig. 9 (shown below) illustrates a ***second*** protocol for communications between server computer system 610 and client computer system 620. *See id.*, ¶0166. In the second protocol, client computer system 620 sends user information to the server computer system 610. *See id.* Server computer system 610 queries database 640 to retrieve a list of all the media content that the user is entitled based on the user information. *See id.* Server computer system 610 sends the list to client computer system 620. *See id.* Client computer system 620 transmits a second list to server computer system 610 that indicates the media content that is not stored at client computer system 620. *See id.* Server computer system 610 transmits the media content from the second list to client computer system 620. *See id.*, ¶¶0166-0167; EX2002, ¶51.

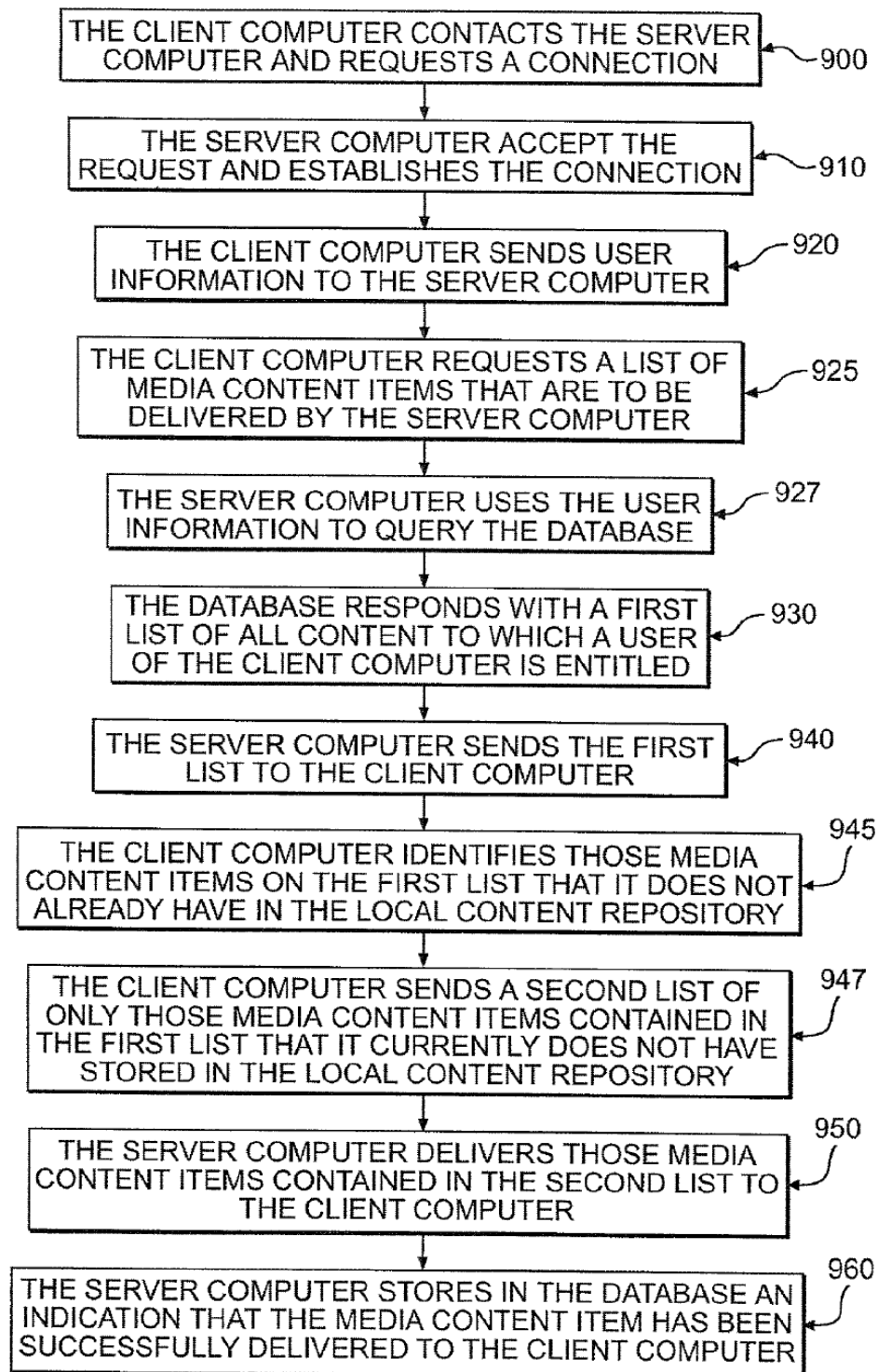


FIG. 9

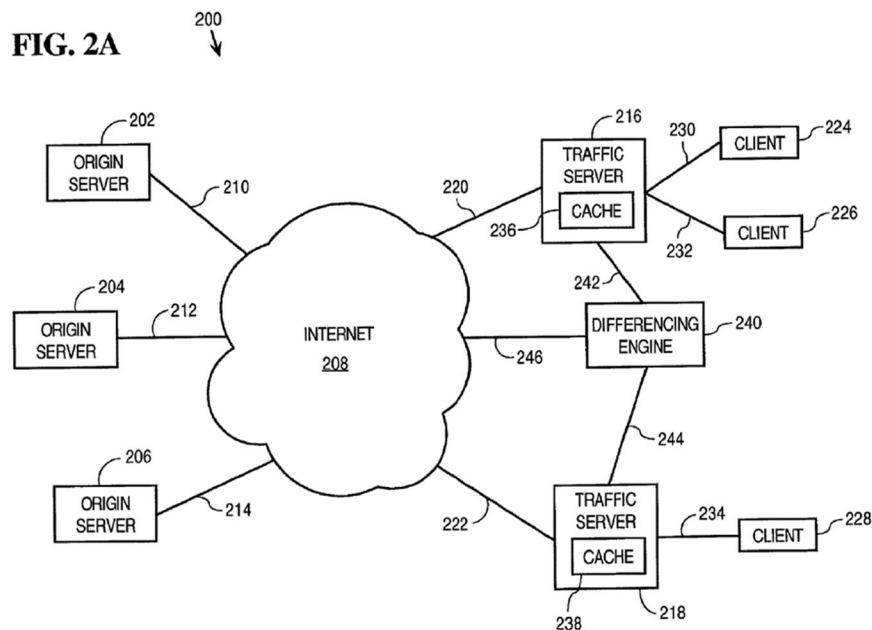
EX1004, Fig. 9.

In both protocols, the server computer system directly sends the content to the client computer. Cassin's server computer system never transmits a response that instructs the client computer system to perform one or more actions to control the flow of content. EX2002, ¶52.

D. Overview of Huston (EX1005)

Huston is U.S. Patent No. 7,243,136, entitled "Approach For Managing And Providing Content To Users," which was filed January 17, 2001. EX2002, ¶53

Huston is directed to selectively refreshing a device's cache. EX1005, Abstract. Huston's Fig. 2A (reproduced below) illustrates an exemplary system.



EX1005, Fig. 2. Origin servers 202, 204, 206 host content. *See id.*, 5:42-55. Content may be delivered from origin servers 202, 204, 206 to traffic servers 216 or 218. *See id.*, 5:56-67. Traffic servers 216 or 218 may be in communication with one or more clients 224, 226, or 228. *See id.*; EX2002, ¶54-55.

Differencing engine 240 may selectively cause content on traffic servers 216 and 218 to be refreshed. *See* EX1005, 6:15-21. Huston states the “differencing engine 240 is configured to selectively cause content to be deleted from traffic servers 216, 218 and/or replaced with newer versions of the deleted content from origin servers 202, 204, 206.” *Id.*, 6:21-25. For example, a first version of content may be provided to traffic server 216. *See id.*, 7:50-60. Differencing engine 240 may detect that a second version of content is available at one of the origin servers. *See id.*, 7:60-67. Differencing engine 240 may cause the first version of content to be deleted from traffic server 216’s cache 236. *See id.*, 7:60-67; EX2002, ¶56.

Differencing engine 240 may detect that the second version of content is available at the origin servers using different techniques. *See* EX1005, 11:13-15. For example, Huston states “content is available on origin server 206. For example, differencing engine 240 may periodically poll origin server 206 to determine what new content has been stored on origin server 206. Differencing engine 240 may also be periodically informed by origin server 206 about new data stored on origin server 206.” *See id.*, 11:15-20; EX2002, ¶57.

In contrast to the claims of the '559 patent, Huston never discloses that the differencing engine receives a content status (or any information) from the traffic servers. Furthermore, Huston only initiates deletion of older content on a traffic server in response to new content becoming available at the origin server and not in response to a content status. *See e.g.*, EX1005, 6:21-25, 7:29-33, 7:60-8:2; EX2002, ¶58.

III. PRIORITY DATE OF THE '559 PATENT

Petitioner assumes a priority date of October 22, 2003, which is the filing date of the '559 patent. Pet., 6, 10. Patent Owner reserves all rights to establish its preferred priority date in this or other proceedings, but does not dispute this assumed priority date for purposes of its Response. The Petition fails regardless of the priority date of the '559 patent. EX2002, ¶59.

IV. LEVEL OF ORDINARY SKILL

Petitioner sets forth a level of ordinary skill. Pet., 10. Patent Owner's expert proposes a similar level, but regardless this Response establishes that Petitioner's arguments fail even under its preferred definition of the level of ordinary skill. EX2002, ¶¶60-63.

V. RELATED MATTERS

Patent Owner is asserting or has asserted the '559 patent in the following district court actions:

- *VideoLabs, Inc. v. Netflix Inc.*, Case 1:22-cv-00229 (D. Del.)
- *Starz Entertainment, LLC v. VL Collective IP, LLC*, Case 1:21-cv-01448 (D. Del.)
- *VideoLabs, Inc. v. Roku, Inc.*, Case 1:23-cv-01136 (D. Del.)
- *VideoLabs, Inc. v. Amazon.com Inc.*, Case 6:22-cv-00079 (W.D. TX), Case 2:22-cv-01451 (W.D. WA), Case 6:22-cv-01167 (W.D. TX)

VI. CLAIM CONSTRUCTION

Petitioner has not presented a proposed construction for any of the terms. *See* Pet. at 15-17. Patent Owner submits that no express construction of any term is required at this stage. For the purposes of the Petition and Patent Owner Response (POR) Patent Owner applies the plain and ordinary meaning of the claim terms. *See Phillips v. AWH Corp.*, 415 F.3d 1303, 1312–13 (Fed. Cir. 2005) (en banc).

Petitioner acknowledges that the Court in a related case construed the term “download” to mean “copy and store in memory of the terminal for subsequent use.” Petitioner has not shown that any claim is unpatentable under either this construction or the plain and ordinary meaning of “download.” EX2002, ¶¶64-65.

VII. ASSERTED GROUNDS

The Petitioner asserts the following grounds in the Petition:

Ground	Challenged Claims	Basis	References
--------	-------------------	-------	------------

1	1, 2, 4, 7, 8, 10, 13, 14, 16, 19, 20, 22	§102	Cassin
2	1-24	§103	Cassin in view of Huston
3	1-24	§103	Huston

VIII. PETITIONER HAS NOT SHOWN THAT ANY CHALLENGED CLAIMS ARE UNPATENTABLE

A. Ground 1: Cassin Does Not Anticipate Claims 1-14

1. Cassin Does not Disclose a Response to the Content Status that Instructs the Terminal or Controller to Perform One or More Actions to Thereby Control the Flow of Content to the Terminal Based Upon the Terminal Status Information, as Required by Claims 1, 7, 13, and 19.¹

Petitioner has not (and cannot) show that Cassin discloses a “response to the content status” “that instructs the terminal [or “controller”] to perform one or more actions to thereby control the [or “a”] flow of content to the terminal based upon the terminal status information,” as required by claims 1, 7, 13, and 19. The Petition alleges five different “response[s] to the content status.” Pet., 24-26. Additionally,

¹ Claims 1, 13, and 19 recite “terminal,” while claim 7 recites a “controller operable with a terminal.” This difference is not relevant to the analysis in this Response. Petitioner also treats the “controller operable with a terminal” as equivalent to the “terminal.”

the Petition alleges four different “content status[es] including the terminal status information,” which the “response” must be “in response” to and “based upon.” *Id.*, 19-23. Due to Petitioner’s scattershot approach, Petitioner’s arguments are first summarized below. Each alleged “response” is analyzed in detail below, but each share a fundamental flaw: the server does not send *instructions* for the terminal to take an action, but takes the action itself. Cassin therefore does not disclose a “response to the content status that instructs to thereby control the flow of content to the terminal” EX2002, ¶66.

a. Summary of Petitioner’s Arguments.

i. “response to the content status that instructs”

The Petition alleges that Cassin discloses five “responses to the content status that instructs,” which are listed below. Pet., 24-26. As explained above, Cassin discloses two separate, distinct protocols. Petitioner’s alleged “responses” are either from the first protocol or the second protocol. EX2002, ¶68. Petitioner’s expert, Mr. Wechselberger, identifies which protocol the claimed “responses” apply to. EX1003, ¶¶73-76; EX2002, ¶¶66-68.

First Protocol:

1. “sending the client computer the list of content the user is entitled” Pet., 25; EX2002, ¶68;

2. “attempting to send the first media content item on the list to the client”
Pet., 25; EX2002, ¶68;
3. sending the second media content item on the list to the client, in the
event “the client does not have the second media content item stored in
its local content repository” Pet., 25-26; EX2002, ¶68;
4. “not delivering the first media content item to the client computer” Pet.,
25; EX2002, ¶68;

Second Protocol:

5. “delivering from its database 640 and first content repository 645 those
media content items from the client’s list” EX1003, ¶76; EX2002, ¶68.

**ii. “content status including terminal status
information”**

The Petition alleges that Cassin includes four “content status[es] including
terminal status information,” which are listed below along with the protocol they fall
within:

1. First and Second Protocol: “user information” (information about the
user of the client computer) Pet., 21; EX1004, ¶¶0164, 0166; EX2002,
¶¶69-71;
2. Second Protocol: a “request for a list of content” Pet., 21; EX1004,
¶0166, Fig. 9; EX2002, ¶¶69-70, 72;

3. First Protocol: “an indication . . . to the server computer that it [the client computer] currently has the media ‘content’ item stored” Pet., 22; EX1004, ¶0164, Fig. 8; EX2002, ¶¶69-70, 73;
4. Second Protocol: “list of content items that it does not have” Pet., 23; EX1004, ¶0166, Fig. 9; EX2002, ¶¶69-70, 74.

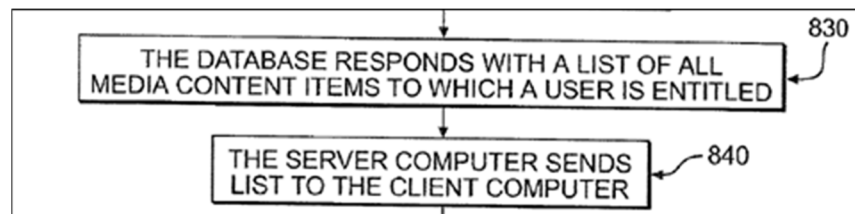
See also Pet., 24-25 (re-listing the four allegedly disclosed “content status”).

a. Cassin Does Not Disclose the “response to the content status that instructs . . .” limitations.

Cassin does not disclose a response to the content status that instructs the terminal or controller to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information. Each alleged “response” is analyzed in detail below, but each shares a common core flaw. EX2002, ¶75. Cassin only discloses the server computer sending the content itself (either the media content item or the list of media content items) to the client computer—and doing so “automatically.” EX1004, ¶¶0140-0143, 0162-0168, Figs. 8, 9; *see also id.*, ¶012 (“The device delivers the media content to the remote device at the second predetermined time.”), ¶0050 (“According to another aspect of the ninth embodiment, the device delivers the media content.”), ¶0103, ¶0108 (“The first device 110 delivers the media content to the remote device at the second predetermined time.”), ¶0130, cl. 2 (“The device defined in claim 1, wherein said

device delivers the media content”), cls. 13, 104, 118. Cassin does not disclose sending *instructions* for the client computer to then do something to control the flow of content. In other words, Cassin only discloses the server computer directly performing the “actions to thereby control the flow of content to the terminal,” not sending instructions to do so. *Id.*, ¶¶0012, 0050, 0103, 0130, cls. 2, 13, 104, 118; EX2002, ¶75.

First, Cassin’s disclosure of “sending the client computer the list of content the client is entitled to” does not disclose the claims “response to the content status that instructs” EX2002, ¶¶76-87. This is shown in the excerpt from the flowchart of Figure 8 below.



EX1004, Fig. 8.

As an initial matter, the claimed “response” must be both a “response *to the content status*” and “based upon the *terminal status information*.” As summarized above, Petitioner alleges four different “content statuses.” *See* Section VIII.A.1.a.i. Of those, the only alleged “content status” that the alleged “response” is “in response to” is the client sending “user information.” EX2002, ¶77. That is because “sending the client computer the list of content the client is entitled to” occurs as part of the

first protocol, while both the “request for a list of content” and the “a second list of only those media content items contained in the first list that it does not already have in the local content repository” are part of the second protocol. *Id.* Petitioner cannot rely on a combination of two separate embodiments to satisfy the claim requirements. *Microsoft Corp. v. Biscotti, Inc.*, 878 F.3d 1052, 1068, 1074-75 (Fed. Cir. 2017) (holding that combining distinct teachings from the same reference was insufficient to anticipate a claim). Here, Cassin is clear that the first and second protocols are distinct, EX1004, ¶¶0162-0167, and a POSITA would not understand them to be combinable. EX2002, ¶52. Therefore, the first and second protocols cannot be combined to meet an anticipation ground. *See Chamberlain Grp., Inc. v. Techtronic Indus. Co.*, 935 F.3d 1341, 1350 (Fed. Cir. 2019) (“[T]he relevant question is whether the reference is sufficiently clear in disclosing the combinability of those elements such that a skilled artisan would ‘at once envisage’ the claimed combination.”); *see also Kennametal, Inc. v. Ingersoll Cutting Tool Co.*, 780 F.3d 1376 (Fed. Cir. 2015).

Next, “an indication . . . to the server computer that ‘it [the client computer] currently has the media content item’ stored” *is* a part of the first protocol, *see* Section VIII.A.1.a.i, but it occurs *after* the “sending the client computer the list of content the client is entitled to.” *See* EX1004, ¶0164, Fig. 8; EX2002, ¶77. Indeed, “the client computer send[ing] user information to the server computer” is the only

alleged “content status including terminal status information” that occurs in the first protocol before the server computer sends “the client computer the list of content the client is entitled to.” EX2002, ¶77. Therefore, the only alleged “content status” that the alleged “response” is “in response to” is the client sending “user information.” *Id.*

However, “user information” is not “content status including terminal status information.” Cassin only gives one example of “user information”: “The sponsor for this content item knows from the statistical user information gathered for that user, that the user is a woman between 35-50 years of age and making between \$25,000 to \$35,000 a year income.” EX1004, ¶0191. From this (and from the term “user data” itself), a POSITA would understand that “user data” is data about the user. EX2002, ¶78. A POSITA would understand that “user data” is not “content status including terminal status information” applying the plain and ordinary meaning of that term, as it includes no information about content or terminal status. *Id.* Cassin’s other references further do not disclose that “user information” discloses “content status including terminal status information.” EX1004, ¶¶0035, 0122, 0154, 0164, 0166, 0185, cl. 68.

“[U]ser information” is not “content status including terminal status information.” Therefore, “sending the client computer the list of content the client is entitled to (step 840)” cannot disclose a response *to the content status* that

instructs the terminal or controller to perform one or more actions to thereby control the flow of content to the terminal *based upon the terminal status information*. EX2002, ¶¶77-82.

Moreover, “sending the client computer the list of content the client is entitled to” does not disclose sending a response to the content status that instructs the terminal or controller to perform one or more actions to thereby control the flow of content to the terminal. Petitioner does not explain how the “list of content the client is entitled to” is a response that instructs—it is not. The “list of content the client is entitled to” does not instruct the client to perform any action, much less an action to control the flow of content. EX2002, ¶83.

Cassin explains that “[t]he server computer then sends this entire list [“for all content to which that user is subscribed”] to the client” and goes on to clarify that “[t]his is done so that the client computer can display on the graphical user interface 700 of the first software playback module 660 a progress bar, which indicates how many items have been received and how many are yet to come.” EX1004, ¶0162.² From this, a POSITA would understand that sending the list of content is simply “done so that the client computer can display on the graphical user interface 700 of

² As Mr. Wechselberger testified at his deposition, the cited paragraph (paragraph 0162) provides “a general description of the operation” of both the first and the second protocol. Deposition Transcript of Anthony Wechselberger (EX2003), 16:7-21. Thus, the description in paragraph 0162 applies to both protocols.

the first software playback module 660 a progress bar” not to “instruct” the client computer to take an action to control the flow of content. *Id.*; EX2002, ¶84.

Additionally, the next “action” that Cassin discloses is that the “server computer attempts to send a first media content item on the list to the client computer.” Ex1004, Fig. 8, ¶0164. Thus, next “action” is not even taken by the client computer and is not done at any “instruct[ion].” EX2002, ¶85. And the next “action” taken by the client computer is “determine[ing] if it already has the first media content item stored in a local content repository” and “if the client computer has the first media content item, the client computer provides an indication to the server computer that it currently has the first media content item.” Ex1004, Fig. 8, ¶0164. But again, a POSITA would understand that the client computer was not instructed to take these actions, but that the actions are based purely on whether or not the client determines it already has the media content. EX2002, ¶86.

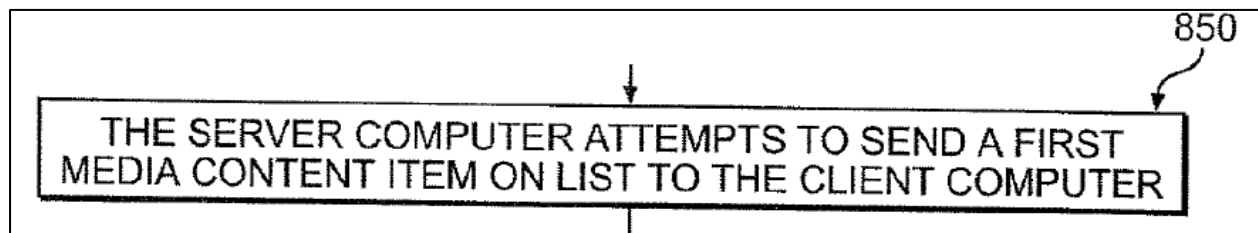
Petitioner’s expert, Mr. Wechselberger, acknowledged that the client computer’s actions are based “on determining the media contents that [the client] does not have in the local content repository 685.” EX2003, 27:6-16; *see also id.*, 23:11-15. Not the alleged “instructions.” A POSITA would understand that a simple list of the content that the client is entitled to in no way instructs the client computer to take any action. EX2002, ¶86. For these reasons, Cassin’s disclosure of “sending the client computer the list of content the client is entitled to (step 840)”

does not disclose a response to the content status that instructs the terminal or controller to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information.”

An analogy illustrates the difference between Cassin’s first protocol and the ’559 patent. Cassin’s first protocol operates like a restaurant with a fixed “vegetarian chef’s special” menu and a fixed “continental chef’s special” menu, both of which include a wine pairing, such as “red” or “white.” In this analogy, the server computer system is a waitress and the client computer system is a customer. When the customer arrives, he tells the waitress whether or not he is a vegetarian (e.g., the client’s user information) and the waitress provides the appropriate “chef’s special” menu, including the type of wine that it includes (e.g., the list of media items to which the user is entitled), to the customer. However, this restaurant also allows customers to bring their own wine, for a small corkage fee; hence, a customer might already have the appropriate wine pairing (e.g., “red” or “white”) when the customer comes to the restaurant. Thus, when the waitress attempts to bring the paired house wine to the customer, he may determine that he already has the appropriate wine in his backpack and he may then tell the waitress no and reject the wine (e.g., an indication to the server that the client currently has the first media item). In this analogy, as in the disclosure in Cassin, the waitress does not *instruct* the customer to retrieve his selected menu items. EX2002, ¶87. In this first protocol, the server

attempts to send the first media content to the client regardless of any alleged instructions from the client. *Id.*; EX2002, ¶¶76-87.

Second, Cassin’s disclosure of “attempting to send the first media content item on the list to the client” does not disclose a response to the content status that instructs the terminal or controller to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information. EX2002, ¶¶88-93. This is shown in the excerpt from the flowchart of Figure 8 below.



EX1004, Fig. 8.

As noted above, the claimed “response” must be both a “response to the content status” and “based upon the terminal status information.” Of the four alleged “content statuses,” the only one that the server “attempting to send the first media content item on the list to the client” is in response to is again the “user information.” EX2002, ¶89. That is because “attempting to send the first media content item on the list to the client” occurs as a part of the first protocol, while the “request for a list of content” and the “a second list of only those media content items contained in the first list that it does not already have in the local content repository 685” are both part of the second protocol. As explained above, Petitioner cannot rely on a

combination of two separate embodiments to satisfy the claim requirements. *Microsoft Corp.*, 878 F.3d at 1068, 1074-75; *Chamberlain Grp., Inc.*, 935 F.3d at 1350; EX2002, ¶52. And again, while the “an indication . . . to the server computer that ‘it [the client computer] currently has the media content item’ stored” is a part of the first protocol, it occurs after the “sending the client computer the list of content the client is entitled to.” See EX1004, ¶0164, Fig. 8. Indeed, “the client computer send[ing] user information to the server computer” is the only alleged “content status including terminal status information” that occurs in the first protocol before the server computer sends “the first media content item on the list to the client.” EX2002, ¶89. Therefore, the only alleged “content status” that the alleged “response” is “in response to” is the client sending “user information.”

However, as discussed above, Cassin’s “user information” does not disclose “content status including terminal status information.” Therefore, “attempting to send the first media content item on the list to the client” (see Pet., 25; EX1004, Fig. 8 step 850) cannot disclose a response *to the content status* that instructs the terminal or controller to perform one or more actions to thereby control the flow of content to the terminal *based upon the terminal status information*. EX2002, ¶90.

Moreover, “attempting to send the first media content item on the list to the client” does not disclose sending a response to the content status that instructs the terminal or controller to perform one or more actions to thereby control the flow of

content to the terminal. Petitioner does not explain how “the first media content item on the list to the client” is a response that instructs—it is not. The “first media content item on the list to the client” does not instruct the client to perform any action, much less an action to control the flow of content. EX2002, ¶91.

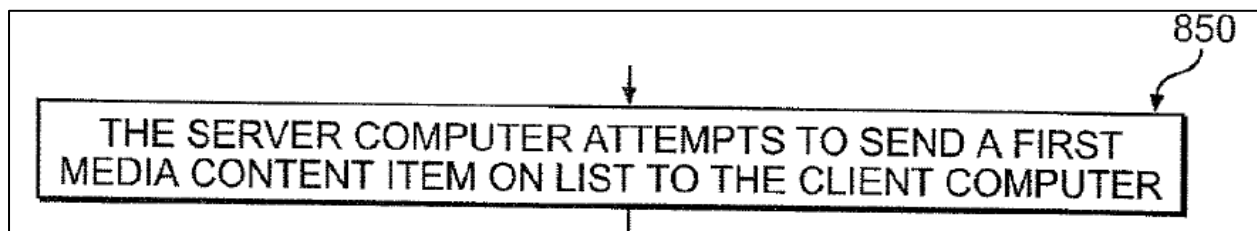
Cassin explicitly discloses that it is the server computer that is “sending” the content, not the client computer taking any action. EX1004, ¶¶0140-0143, 0162-0165, Fig. 8; *see also id.*, ¶¶0012, 0050, 0103, 0108, 0130, cls. 2, 13, 104, 118. Since the server computer has already sent the content, a POSITA would understand that there is no action for the user to then take, and certainly none to control the flow of content (since the content has already been sent) and not at the instruction of the server computer. EX2002, ¶¶91-92.

Additionally, the next “action” taken by the client computer is “determin[ing] if it already has the first media content item stored in a local content repository” and “if the client computer has the first media content item, the client computer provides an indication to the server computer that it currently has the first media content item.” EX1004, ¶0164, Fig. 8. But again, a POSITA would understand that the client computer was not instructed to take these actions, but that the actions are based purely on whether or not the client determines it already has the media content. EX2002, ¶93. Mr. Wechselberger acknowledged that the client computer’s actions are based “on determining the media contents that [the client] does not have in the

local content repository 685.” EX2003, 26:22-27:16; *see also id.*, 23:11-15. Not the alleged “instructions.” EX2002, ¶93.

The restaurant analogy from above applies here as well. A POSITA would understand that sending the first (or subsequent) content that the client is entitled to in no way instructs the client computer to take any action. EX2002, ¶¶93, 87. For these reasons, a POSITA would understand that Cassin’s disclosure of “attempting to send the first media content item on the list to the client” does not disclose a response to the content status that instructs the terminal or controller to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information. EX2002, ¶88-93.

Third, Cassin’s disclosure of sending the second media content item on the list does not disclose a “response to the content status that instructs the terminal or controller to perform one or more actions to thereby control the flow of content.” This is shown in the excerpt from the flowchart of Figure 8 below (when the flow is repeated).



EX1004, Fig. 8.

Instead, as explained above, a POSITA would understand it discloses the server itself taking the action—sending the content. EX2002, ¶¶94-97. As Mr. Wechselberger agreed during his deposition: “then after that [the client has indicated that it already has the first piece of content], *the server computer sends the next piece of content.*” EX2003, 24:1-12 (emphasis added) (“Q. So then after that, the server computer sends the next piece of content, right? A. Yes. There will either have been an acceptance of the first offer or not, and the last sentence of paragraph 164 says: Thereafter, Steps 850 through 880 are repeated for each media content as necessary.”). The point of Cassin is how to manage data flow between the server and the client so that the server can only send the necessary content—something that is only necessary because it is the server itself sending the content. EX2002, ¶95.

As discussed above, a POSITA would not understand the *server* sending content to be the *client* taking action to control the flow of action and in no way an instruction for the client to do so. EX2002, ¶96. As also explained above, the user’s only actions are to say whether or not it has content already, which it does not do at the server computer’s instruction.

For these reasons, Cassin’s disclosure of sending the second media content item on the list to the client in the event “the client does not have the second media content item stored in its local content repository” does not disclose a response to the content status that instructs the terminal or controller to perform one or more

actions to thereby control the flow of content to the terminal based upon the terminal status information. EX2002, ¶¶94-97.

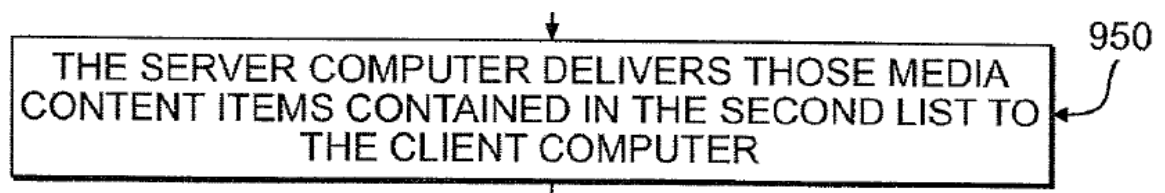
Fourth, “not delivering the first media content item to the client computer” does not disclose a response to the content status that instructs the terminal or controller to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information. EX2002, ¶98.

Cassin discloses that the server computer sends the second media content item on the list and, if the client computer already has that content, then it is not delivered to the user. *Id.* Thus, “not delivering” content does not disclose the claimed element for the same reason as sending the content in the first place does not—as explained above, the server computer is taking the action itself (not delivering the content after the user says it already has it). *Id.* Moreover, “not delivering” something cannot be sending a response of any type.

For these reasons, a POSITA would not understand Cassin’s disclosure of not delivering the first media content item to the client computer to disclose a response to the content status that instructs the terminal or controller to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information. *Id.*; EX2002, ¶98.

Fifth, Cassin’s disclosure of sending the media content items from the client’s list does not disclose a “response to the content status that instructs the terminal or

controller to perform one or more actions to thereby control the flow of content.” This is shown in the flowchart of Figure 9 below. Instead, a POSITA would understand Cassin discloses that the server itself takes the action, sending the content. EX2002, ¶99. Indeed, there are *no further client actions* after the server sends the media content items from the client’s list. EX2002, ¶99. As Mr. Wechselberger testified, “the server computer delivers those media content items contained in the second list to the client computer.” EX2003, 27:17-28:1, 29:16-30:7 (“The text in there says that Step 947, the client computer sends a second list of only those media content items contained in the first list that it currently does not have stored in local content repository 685. In Step 950, the server computer delivers those media content items contained in the second list, so it doesn’t say anything about other information.”). Cassin explicitly discloses that it is the server that “delivers those media content items.” EX1004, Fig. 9, ¶0166. A POSITA would not understand the *server* sending content to be the *client* taking action to control the flow of action and in no way is it an instruction for the client to do so. EX2002, ¶99. The user’s only actions are to say whether or not it has content already, which it does before this alleged “response . . . that instructs.”



EX1004, Fig. 9.

Again, an analogy regarding Cassin's second protocol is illustrative. *See* EX2002, ¶100. Cassin's second protocol operates like a restaurant with a single fixed "chef's special" menu that includes a wine pairing, such as "red" or "white," where the server computer system is a waitress, and the client computer system is a customer. When the waitress arrives, she provides the "chef's special" menu of items, including the type of wine that it includes (e.g., the first list of media items), to the customer. However, this restaurant also allows customers to bring their own wine, for a small corkage fee; hence, a customer might already have the appropriate wine pairing (e.g., "red" or "white") when the customer comes to the restaurant. Thus, when the waitress brings the "chef's special" menu to the customer, he may determine that he already has the appropriate wine in his backpack and he may then tell the waitress that he only needs the other items on the "chef's special" menu (e.g., the second list of only those media content items contained in the first list that the client does not have stored in the local content repository). The waitress then brings the menu items to the customer absent the house wine. In this analogy, as in the disclosure in Cassin, the waitress does not *instruct* the customer to retrieve his selected menu items. EX2002, ¶¶99-100.

For these reasons, Cassin's disclosure of "delivering from its database 640 and first content repository 645 those media content items from the client's list" does

not disclose a response to the content status that instructs the terminal or controller to perform one or more actions to thereby control the flow of content to the terminal or controller based upon the terminal status information. *Id.*, ¶101.

2. Cassin Does Not Disclose Receiving Server Status Information that Comprises a Listing of At Least One Piece of Content Available from the Source

Claim 1 requires “server status information regarding a source of content, wherein the server status information comprises a listing of at least one piece of content available from the source.” Cassin does not disclose this limitation.

Petitioner states that “the server computer system has a first content repository 645, that stores multimedia content in the form of media files, which is accessed via database 640.” Pet., 23 (citing EX1004, ¶0146). Petitioner goes on to state that “the server then queries database 640 based on the user information” and that the “server then receives from the database a list of all content to which the user is entitled, which is ‘server status information regarding a source of content.’” *Id.* That is not correct. EX2002, ¶¶102-105.

Cassin does not disclose that the server computer system can “access[]” the “content repository 645” “via database 640.” Instead, Cassin discloses that the system accesses *the database 640*—not the “content repository 645”—to provide a list of content that the user is entitled to, irrespective of the content’s availability at the source. *See* EX1004, ¶¶0146, 0164, 0166. Cassin discloses that the media files

(i.e., the alleged content) are stored in first content repository 645 (i.e., the alleged source). *See id.* Database 640 stores “information related to the system in the form of metadata. The information must include metadata about the media content available to the system, including for example, descriptions of packages, series and episodes. The information should also include metadata that relates to users of the system. This metadata may include, for example, account information, billing history and statistics.” *Id.*, ¶0146. Cassin does not disclose that the Database 640 can store the media files themselves. *Id.*, ¶0146 (“The database need not contain the media files themselves.”). In both the first and second protocol, the server computer system queries database 640 for a list of content that the user is entitled to. *See id.*, ¶0164, 0166. Database 640 determines the list of content that the user is entitled to based on the user information and metadata stored in database 640. *See id.*, ¶¶0146, 0164, 0166. EX2002, ¶¶104.

Cassin does not disclose that Database 640 queries the first content repository 645 to identify the content available at first content repository 645 when preparing the list. From this, a POSITA would understand that the availability of the media files may change in first content repository 645. EX2002, ¶¶104-105. For example, the media files in first content repository 645 may be deleted, corrupted, or otherwise unavailable. Thus, when queried, database 640 will return a list of media content items that the user is entitled to based on the user information and metadata,

irrespective of whether the media files are available from first content repository 645. *See id.* In this scenario, the list returned by database 640 will include content that is not available at the source. As such, the list returned by database 640 does not truly indicate the availability of the content from the *source*. EX2002, ¶¶104-105.

**3. Cassin Does Not Disclose Terminal Status Information
Comprising a Listing of At Least One Piece of Content
Stored in the Memory**

Independent claims 7, 13, and 19 (and claims depending from those claims) require that “terminal status information comprising a listing of at least one piece of content stored in” memory. EX1001, cls. 7, 13, 19. Cassin does not disclose this limitation. EX2002, ¶¶106-114.

The Petition is unclear as to what exactly in Cassin allegedly discloses “terminal status information comprising a listing of at least one piece of content stored in” memory. The Petition only explicitly identifies “an indication . . . to the server computer that it [the client computer] currently has the media “content” item stored.” Pet., 22. However, Petitioner’s expert, Mr. Wechselberger also identifies “user information (information about the user of the client computer).” EX1003, ¶90. Both arguments are addressed below.

First, a POSITA would understand that Cassin’s disclosure of a “an indication . . . to the server computer that it [the client computer] currently has the media

‘content’ item stored” does not disclose that it includes a “listing of at least one piece of content stored in” memory. *See* Pet., 22; *see also* EX1003, ¶91. The Petition asserts that “[t]he indication can include a list of the media content items already stored on the client computer (*i.e.*, a ‘listing of at least one piece of content stored in a memory of the terminal’).” Pet., 22. But Cassin does not disclose what is included in the “indication” and a POSITA would understand that Cassin does not disclose that this “indication” includes “a list of the media content items already stored on the client computer.” EX2002, ¶109.

Indeed, Mr. Wechselberger agreed that Cassin does not “define the structure or contents of the indication,” (EX2003, 32:13-19) and that Cassin does not specify “how [the indication] accomplishes it, what data it sends,” EX2003, 33:22-34:5. A POSITA would understand that the disclosure of the “implementation” does not disclose “content status including terminal status information.” EX2002, ¶110. Moreover, a POSITA would understand that for the sake of communication and computing efficiency, the natural interpretation of such an indication is that it is a simple “yes” or “no,” or equivalent. EX2002, ¶110.

Further, Mr. Wechselberger testified that he did does not “have an opinion either way whether the indication could just be a message that says denied.” EX2003, 34:21-35:2. An “indication” that merely says “no” would not disclose to a POSITA a “listing of at least one piece of content stored in” memory because,

applying the plain and ordinary meaning of that term, the word “no” is not itself a listing of at least one piece of content. EX2002, ¶111. In the analogy discussed above, the customer could simply say “no” when offered a bottle of wine, instead of listing all of the wine the customer already has. Therefore, a POSITA would understand that a disclosure of the “indication” does not disclose a “listing of at least one piece of content” because it merely discloses the “indication” and is silent as to how that is accomplished. EX2002, ¶111.

Petitioner also relies on claims 130-133 to assert that the “indication” must disclose a “content status including terminal status information comprising a listing of at least one piece of content stored in the memory,” arguing that because “POSITA would have understood that Claim 133 narrows the scope of the second group of media content items to be ‘a group of media content items that the remote devices will accept for delivery from the device,’ which necessarily means that Claim 132, which is broader than Claim 133, can include either a group of media content items that the remote devices will or will not accept for delivery from the device.” EX1003, ¶¶92-93; *see also* Pet., 20-23. However, even if Claim 132 *could* include a group of media content items that the remote device will not accept for delivery from the device (and therefore media the client already had) that does not mean that it discloses it to a POSITA. EX2002, ¶112. In general, just because

something *can* perform a step does not mean that disclosure of that thing discloses the step to a POSITA. *Id.*

Second, as discussed above, a POSITA would understand that Cassin’s disclosure of “user information” does not disclose “content status including terminal status information” and certainly does not disclose that the “user information” would include a “listing of at least one piece of content stored in” memory. Cassin only gives one example of “user information”: “The sponsor for this content item knows from the statistical user information gathered for that user, that the user is a woman between 35-50 years of age and making between \$25,000 to \$35,000 a year income.” EX1004, ¶0191. From this (and from the term “user data” itself), a POSITA would understand that “user data” is data about the user. EX2002, ¶113. A POSITA would understand that “user data” is not “content status including terminal status information” applying the plain and ordinary meaning of that term, as it includes no information about content or terminal status and would further understand that “user data” does not include a “listing of at least one piece of content stored in” memory. *Id.* Cassin’s other references further do not disclose that “user information” discloses “content status including terminal status information” or includes a “listing of at least one piece of content stored in” memory. EX1004, ¶¶0035, 0122, 0154, 0164, 0166, 0185, cl. 68; EX2002, ¶113.

* * *

Therefore, Cassin does not disclose each limitation of at least claims 1, 7, 13, 19. Claims 2, 4, 8, 10, 14, 16, 20, and 22 are also not anticipated by Cassin by virtue of their dependency to claims 1, 7, 13, 19, respectively. EX2002, ¶115.

B. Ground 2: Cassin and Huston Do Not Render Obvious Any Claim of the '559 Patent.

Petitioner asserts that claims 1-24 are obvious over Cassin in view of Huston. They are not. EX2002, ¶116.

1. Cassin and Huston Do Not Teach Or Suggest a Response to the Content Status that Instructs the Terminal or Controller to Perform One or More Actions to Thereby Control the Flow of Content to the Terminal Based Upon the Terminal Status Information, as Required by Claims 1, 7, 13, and 19.

As explained above, Cassin does not teach or suggest a “response to the content status that instructs” *See* Section VIII.A.1. The Petition does not provide any additional analysis or assertions as to how this limitation is rendered obvious either by Cassin itself or in view of Huston. *Compare* Pet., 40-41 *with id.*, 24-26; EX1003, ¶138. Therefore, Cassin and Huston do not render obvious receiving server status information that comprises a listing of at least one piece of content available from the source for the same reasons as indicated above. EX2002, ¶117.

2. Cassin and Huston Do Not Teach Or Suggest Server Status Information that Comprises a Listing of At Least One Piece of Content Available from the Source

As explained above, Cassin does not teach or suggest server status information that comprises a listing of at least one piece of content available from the source. *See* Section VIII.A.2. The Petition does not provide any additional analysis or assertions as to how this limitation is rendered obvious either by Cassin itself or in view of Huston. Pet., 40; EX1003, ¶136. Therefore, Cassin and Huston do not render obvious server status information that comprises a listing of at least one piece of content available from the source for the same reasons as indicated above. EX2002, ¶118.

3. Cassin and Huston Do Not Teach Or Suggest Terminal Status Information Comprising a Listing of At Least One Piece of Content Stored in the Memory

As explained above, Cassin does not render obvious terminal status information comprising a listing of at least one piece of content stored in the memory. *See* Section VIII.A.3. Petitioner does not *explicitly* allege that “terminal status information comprising a listing of at least one piece of content stored in the memory” would be obvious in light of Cassin by itself. Pet., 31-32, 37-39. It would not, and any implication otherwise is no more than impermissible hindsight. “To draw on hindsight knowledge of the patented invention, when the prior art does not contain or suggest that knowledge, is to use the invention as a template for its own

reconstruction—an illogical and inappropriate process by which to determine patentability.” *Sensonics Inc. v. Aerosonic Corp.*, 81 F.3d 1566, 1570 (Fed. Cir. 1996) (citing *W.L. Gore & Assoc. v. Garlock, Inc.*, 721 F.2d 1540, 1553 (Fed. Cir. 1983). EX2002, ¶119.

Here, a POSITA would understand that Cassin’s “indications” should contain as little data as possible to increase speed and decrease storage requirements. EX2002, ¶¶120, 111. And since Cassin does not disclose what data is in the “indications,” it could simply be the word “no” or “denied.” EX2002, ¶¶120, 111; EX2003, 34:21-35:2. It would thus not be obvious to add additional, unnecessary information, such as “a listing of at least one piece of content stored in the memory.” *See Sensonics Inc.*, 81 F.3d at 1570 (“The invention must be viewed not after the blueprint has been drawn by the inventor, but as it would have been perceived in the state of the art that existed at the time the invention was made.”). EX2002, ¶1120.

Instead of alleging that Cassin by itself renders this element obvious, Petitioner asserts that Huston discloses it. Huston does not disclose that “the differencing engine 240 would receive an identification of the versions of the content stored on the caches 236, 238 (‘content status including terminal status information’) from the traffic servers 216, 218 (‘terminals’).” EX2002, ¶¶121-124. Instead, Petitioner notes that in Huston “that the differencing engine 240 instructs the traffic servers to delete content after the differencing engine compares the versions of

content stored on caches 236, 238 of the traffic servers 216, 218 with the versions of the corresponding content stored on the origin servers 202, 204, 206.” Pet., 39; *see also* EX1003, ¶161. From that, Petitioner reasons that “[i]n order to compare the versions of content, a POSITA would have understood that the differencing engine 240 would receive an identification of the versions of the content stored on the caches 236, 238 (‘content status including terminal status information’) from the traffic servers 216, 218 (‘terminals’), which are ‘remote’ from the differencing engine 240 (‘network entity’).” *Id.* Petitioner’s argument mischaracterizes Huston and what a POSITA would understand from it. EX2002, ¶¶121-124.

Huston discloses that the differencing engine can be aware of the content that is being sent to and stored on the traffic servers. EX1005, 7:46-49 (“In this situation, differencing engine 240 interacts directly with origin servers 202, 204, 206 to determine when new content is available and to ‘push’ the new content to traffic servers 216, 218.”), 7:38-49 (“For example, according to one embodiment of the invention, differencing engine 240 is co-located with, or at least communicatively coupled to, origin servers 202, 204, 206.”); EX1003, ¶161; EX2003, 58:17-60:8, 65:9-66:3 (acknowledging that the Differencing Engine can be aware of the content that is sent to the traffic server). From this a POSITA would understand that the differencing engine would *already* be aware of what content is on the traffic servers, and thus would not need to “receive an identification of the versions of the content

stored on the caches 236, 238 . . . from the traffic servers 216, 218 (“terminals”).” EX2002, ¶123. Since Huston discloses the differencing engine tracking the content stored in the traffic server’s cache when it transmits the content to the traffic server, the differencing engine has no need to receive a content status including terminal status information. EX2002, ¶¶123-124. Therefore, Huston does not disclose, or even suggest, receiving a content status including terminal status information of any sort, much less terminal status information comprising a listing of at least one piece of content stored in the memory. *Id.*

Petitioner does not provide any additional analysis as to how “terminal status information comprising a listing of at least one piece of content stored in the memory” is rendered obvious by Cassin in view of Huston. Pet., 37-40; EX1003, ¶161. Therefore, based on the reasons above, Cassin and Huston do not teach or suggest “terminal status information comprising a listing of at least one piece of content stored in the memory.” EX2002, ¶¶125.

4. Cassin and Huston Do Not Teach Or Suggest a Response to the Content Status that Instructs the Terminal to Delete at Least One Piece of Content from the Memory of the Terminal Based Upon the Listing of At Least One Piece of Content Stored in the Memory of the Terminal

Claims 2, 3, 5, 8, 9, 11, 14, 15, 17, 20, 21, 23 recite a response to the content status that instructs the terminal to delete at least one piece of content from the memory of the terminal based upon the listing of at least one piece of content stored

in the memory of the terminal. EX1001, cls. 7, 13, 19. Cassin and Huston do not teach or suggest this limitation. EX2002, ¶¶126-127.

Petitioner acknowledges that Cassin does not disclose this element, but instead argues that Huston discloses it. It does not. EX2002, ¶¶128-129. The claim element requires that the “response . . . that instructs the terminal to delete” is in response “to the content status” and “based upon” terminal status information. EX1003, ¶143. However, Huston discloses that the differencing engine only issues the “delete” command in response to it detecting a second version of the content in the origin server. EX1005, 7:60-63 (“In step 260, differencing engine 240 detects that a second version of the content has been created and is available. In step 262, differencing engine 240 causes the first version of content to be deleted from cache 236.”); Fig. 2B; 6:56-7:5 (“Sometime later, a second more recent version of the particular electronic document is created by a content provider and stored on origin server 202.”); EX2002, ¶128. Mr. Wechselberger acknowledged at his deposition that “the differencing engine causes content to be deleted from the cache shown in Step 262” “based on” “detect[ing] that a second version of the content has been created.” EX2003, 66:16-67:1; EX1003, ¶153 (“In response to this comparison, Huston discloses that the differencing engine 240 is configured to selectively cause content to be deleted from the traffic servers 216, 218 based on the comparison . . .”). Thus a POSITA would understand that the “Delete” command

issues in response to new content in the origin server, not in response to a content status from the terminal server and not based on terminal status information. EX2002, ¶128. Indeed, as explained above, Huston does not disclose receiving content status from the terminal at all. *Id.*, ¶¶128-129.

Moreover, even if Huston did disclose this limitation (which it does not), it would not be obvious to a POSITA to modify Cassin in view of Huston. *See* EX1003, ¶¶151-153; EX2002, ¶129. Cassin explicitly discloses a type of deleting—the user being able to manually delete content. EX1004, ¶0169 (“To delete a media content item, a user would either drag the item using a mouse to the Trash folder 717, or select the content item and press the DELETE key on a keyboard associated with the client computer.”); EX2003, 17:8-10 (“Q. Cassin teaches that the user can manually delete content, right? A Yes, that’s disclosed in paragraph 169.”). Thus, a POSITA would understand that Cassin was aware of the need to delete content but implicitly taught away from the server being able to instruct the terminal to delete content. EX2002, ¶129; *In re Gurley*, 27 F.3d 551, 553 (Fed. Cir. 1994) (holding “[a] reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant.”); *Arctic Cat Inc. v. Bombardier Recreational Prod. Inc.*, 876 F.3d 1350, 1360 (Fed. Cir. 2017). EX2002, ¶129.

Further, modifying Cassin to add a delete command is contrary to the “intended purpose” of Cassin. *See Adidas AG v. Nike, Inc.*, 963 F.3d 1355, 1359 (Fed. Cir. 2020); *Masimo Corporation, v. Apple Inc.*, IPR2023-00807, Paper 10 at 20-21 (November 21, 2023). In Cassin, the server only checks whether a terminal already has a piece of content in order to ensure that duplicative content is not sent. *See* EX1004, ¶0164. The purpose is not that the server *controls* what content is stored at the terminal, only that it provides non-duplicative content. EX2002, ¶¶46, 129. Indeed, Cassin is directed to a “Playback Scheme,” (*id.*, Title), which further demonstrates that there was no motivation to allow the server computer to delete content from the client computer (i.e., a path divergent from the path that was taken by the Mehta). *See id.* Title, ¶0164; EX2002, ¶130.

Nor would it be obvious to a POSITA *how* to combine Cassin and Huston. EX2002, ¶129. The Federal Circuit has “held that where a party argues a skilled artisan would have been motivated to combine references, it must show the artisan ‘would have had a reasonable expectation of success from doing so.’” *Arctic Cat* 876 F.3d at 1360–61 (Fed. Cir. 2017) (quoting *In re Cyclobenzaprine Hydrochloride Extended–Release Capsule Pat. Litig.*, 676 F.3d 1063, 1068 (Fed. Cir. 2012)). Petitioner alleges that “modifying Cassin to include functionality for the server to send delete commands to delete content in the repository of the client computer, as disclosed in Huston, is simply using a known technique (deleting content) to

improve a similar device in the same way.” Pet., 35-36. But Cassin only describes that the server computer can send (1) lists of content and (2) content. *See* EX1004, ¶¶0164, 0166. There is no mechanism in Cassin to be able to send instructions to the client computer, and thus a POSITA would not have a reasonable expectation of success in combining Cassin and Huston. EX2002, ¶¶129-130.

* * *

Therefore, Cassin and Huston do not teach or suggest each limitation of at least claims 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, and 23. Claims 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, and 24 are also not rendered obvious by Cassin in view of Huston by virtue of their dependency to claims 1, 6, 7, 12, 13, 19, 20, 24. EX2002, ¶131.

C. Ground 3: Huston Does Not Render Obvious Any Claim of the '559 Patent.

1. Huston Does Not Disclose Each Element of Claims 1, 7, 13, 19.

Claims 1, 7, 13, and 19 (and claims depending from those claims) require receiving “from a terminal located remote from the apparatus, a content status including terminal status information.” As shown above, Huston does not disclose receiving any information from the terminals, including “a content status including terminal status information.” *See* Section VIII.B.3. Petitioner does not provide any additional analysis as to how this limitation is rendered obvious by Huston. Pet., 51-54; EX1003, ¶¶207-212; EX2002, ¶132.

Claims 1, 7, 13, and 19 (and claims depending from those claims) also require sending or receiving “a response to the content status that instructs the terminal to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information.” As shown above, Huston does not disclose “a response to the content status that instructs the terminal to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information.” *See* Section VIII.B.4. Petitioner does not provide any additional analysis as to how this limitation is rendered obvious by Huston. Pet., 51-54; EX1003, ¶¶207-212; EX2002, ¶133.

Claims 7, 13, and 19 (and claims depending from those claims) require that “terminal status information comprising a listing of at least one piece of content stored in” memory. As shown above, Huston does not disclose any “terminal status information,” including “terminal status information comprising a listing of at least one piece of content stored in” memory. *See* Section VIII.B.3. Petitioner does not provide any additional analysis as to how this limitation is rendered obvious by Huston. Pet., 51-54; EX1003, ¶¶207-212; EX2002, ¶134.

Therefore, Huston does not teach or suggest each limitation of claims 1, 7, 13, 19. The remaining claims are also not rendered obvious by Huston by virtue of their dependency to claims 1, 7, 13, 19. EX2002, ¶¶132-134.

2. Huston Does Not Disclose Each Element of Claims 2, 3, 5, 8, 9, 11, 14, 15, 17, 20, 21, 23.

Claims 2, 3, 5, 8, 9, 11, 14, 15, 17, 20, 21, 23 recite a response to the content status that instructs the terminal to delete at least one piece of content from the memory of the terminal based upon the listing of at least one piece of content stored in the memory of the terminal. As explained above, Huston does not disclose a response to the content status that instructs the terminal to delete at least one piece of content from the memory of the terminal based upon the listing of at least one piece of content stored in the memory of the terminal. *See* Section VIII.B.4. Petitioner does not provide any additional analysis as to how these limitations are rendered obvious by Huston. Pet., 54-58; EX1003, ¶¶214-224; EX2002, ¶¶135-136.

Therefore, Huston does not teach or suggest each limitation of claims 3, 5, 9, 11, 15, 17, 21, and 23. Claims 4, 6, 10, 12, 16, 18, 22, and 24 are also not rendered obvious by Huston by virtue of their dependency to claims 3, 5, 9, 11, 15, 17, 21, and 23. EX2002, ¶¶135-136.

IX. CONCLUSION

For the reasons stated above, the Petition fails to show that any challenged claim is unpatentable.

Dated: December 22, 2023

By: /Christine E. Lehman/
Christine E. Lehman
PTO Reg. No. 38,535
Counsel for Patent Owner

CERTIFICATE OF COMPLIANCE

The undersigned hereby certifies that Patent Owner's Response consists of 11,484 words, excluding the table of contents, certificate of service, this certificate of compliance, or table of exhibits. The Response complies with the type-volume limitation of 14,000 words as mandated in 37 C.F.R. § 42.24. In preparing this certificate, counsel has relied on the word count of the word-processing system used to prepare this paper (Microsoft Word).

Dated: December 22, 2023

Respectfully submitted,

/Christine E. Lehman/

Christine E. Lehman

PTO Reg. No. 38,535

Counsel for Patent Owner

CERTIFICATE OF SERVICE

The undersigned hereby certifies that a copy of the foregoing **PATENT OWNER'S RESPONSE** and **Associated Exhibits** was served on December 22, 2023, via electronic mail, as agreed to by counsel, upon the following counsel for Petitioner:

Aliza George Carrano, Reg. No. 70,637
Willkie Farr & Gallagher LLP
1875 K Street NW,
Washington, DC 20006
acarrano@willkie.com

Indranil Mukerji, Reg. No. 46,944
Willkie Farr & Gallagher LLP
1875 K Street NW,
Washington, DC 20006
imukerji@willkie.com

Stephen A. Marshall (*Pro Hac Vice*)
Willkie Farr & Gallagher LLP
1875 K Street NW,
Washington, DC 20006
smarshall@willkie.com

And to:
Netflix-VL_WFG@willkie.com

Dated: December 22, 2023

/Christine E. Lehman/
Christine E. Lehman
PTO Reg. No. 38,535