UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ROKU INC.,
Petitioner,

v.

VIDEOLABS, INC.,
Patent Owner.

Case No. IPR2025-00071
Patent No. 7,440,559

PETITION FOR *INTER PARTES* REVIEW
UNDER 35 U.S.C. §§ 311-319 AND 37 C.F.R. § 42.1 *et seq.*

TABLE OF CONTENTS

TABLE OF AUTHORITIES

CASES

TABLE OF EXHIBITS

Exhibit	Description
1001	U.S. Patent No. 7,440,559 ("'559")
1002	File History of U.S. Patent Application No. 10/690,692 ("'559FH")
1003	Declaration of Bruce McNair ("McNair")
1004	CV of Bruce McNair
1005	U.S. Patent No. 6,341,316 ("Kloba")
1006	U.S. Patent Application Publication No. 2003/0079038 ("Robbin")
1007	*Starz Entm't, LLC v. VL Collective IP, LLC*, 1:21-cv-01448, Dkt. 88, pages 1-4 (Claim Construction Order) (D. Del. Jan. 10, 2023)
1008	*Netflix, Inc. v. VideoLabs, Inc*, IPR2023-00630, Paper 2 (Petition) (P.T.A.B. Feb. 23, 2023)
1009	*Netflix, Inc. v. VideoLabs, Inc*, IPR2023-00630, Paper 10 (Institution Decision) (P.T.A.B. Oct. 3, 2023)
1010	Federal Judicial Caseload Statistics (June 30, 2024)
1011	*VideoLabs, Inc. v. Roku, Inc.*, 1:23-cv-01136, Dkt. 36 (Defendant Roku Inc.'s Motion to Stay) (D. Del. July 18, 2024)
1012	*VideoLabs, Inc. v. Roku, Inc.*, 1:23-cv-01136, Dkt. 37 (Defendant Roku, Inc.'s Opening Brief in Support of Its Motion to Stay Pending Inter Partes Review) (D. Del. July 18, 2024)
1013	*VideoLabs, Inc. v. Roku, Inc.*, 1:23-cv-01136, Dkt. 41 (Order) (D. Del. July 26, 2024)
1014	Reserved
1015	*Netflix, Inc. v. VideoLabs, Inc*, IPR2023-00630, Paper 31 (Final Written Decision) (P.T.A.B. Oct. 2, 2024)
1016	*VideoLabs, Inc. v. Roku, Inc.*, 1:23-cv-01136, Dkt. 6 (Summons) (D. Del. Dec. 22, 2023)
1017	*AviaGames, Inc. v. Skillz Platform, Inc.*, IPR2022-00530, Paper 14 (Decision on Director Review) (P.T.A.B. Mar. 2, 2023)
1018	U.S. Patent Application Publication No. 2003/0163431 ("Ginter")

Exhibit	Description
1019	U.S. Patent No. 5,943,422 ("Van Wie")

TABLE OF CHALLENGED CLAIMS

U.S. 7,440,559		
#	**Sub-part**	**Text**
1	**[pre]**	An apparatus comprising:
	[a]	a processor configured to
	[b.i]	receive, from a terminal located remote from the apparatus, a content status including terminal status information, and
	[b.ii]	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,
	[c]	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
	[d]	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.
2	**[pre]**	An apparatus according to claim 1,
	[a]	wherein the terminal comprises a memory, and
	[b]	wherein the processor is configured to send, to the terminal, a response to the content status that instructs the terminal to at least one of delete at least one piece of content from the memory of the terminal, or
	[c]	download at least one piece of content from the source.
3	**[pre]**	An apparatus according to claim 2,
	[a]	wherein the terminal status information comprises a listing of at least one piece of content stored in the memory of the terminal, and
	[b]	wherein the processor is configured to send, to the terminal, 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.
4	**[pre]**	An apparatus according to claim 2,
	[a]	wherein the server status information comprises a listing of at least one piece of available content from the source, and
	[b]	wherein the processor is configured to send, to the terminal, a response to the content status that instructs the terminal to download at least one piece of content from the source based upon the listing of at least one available piece of content from the source.
5	**[pre]**	An apparatus according to claim 2,
	[a]	wherein the processor is configured to determine if the memory of the terminal includes at least one piece of content to delete, and
	[b]	wherein the processor is configured to send, to the terminal, a response to the content status that instructs the terminal to delete at least one piece of content when the processor determines that the memory of the terminal includes at least one piece of content to delete.
6	**[pre]**	An apparatus according to claim 5,
	[a]	wherein the processor is further configured to determine if source includes at least one available piece of content for the terminal to download, and
	[b]	wherein the processor is configured to send, to the terminal, a response to the content status that instructs the terminal to download at least one available piece of content when the processor determines that the source includes at least one available piece of content for the terminal to download.
7	**[pre]**	An apparatus comprising:
	[a]	a controller operable with a terminal including a memory configured to store at least one piece of content,
	[b]	wherein the controller is configured to send a content status including terminal status information comprising a listing of at least one piece of content stored in the memory,
	[c]	wherein the controller is configured to send the content status to a remote network entity, and

	[d]	receive a response to the content status from the network entity that instructs the controller to perform one or more actions to thereby control a flow of content to the terminal based upon the terminal status information, and
	[e]	wherein the at least one piece of content stored in the memory, and the content for which the network entity is configured to control the flow, comprise multimedia content.
8	**[pre]**	An apparatus according to claim 7,
	[a]	wherein the controller is configured to receive a response that instructs the controller to at least one of delete at least one piece of content from the memory of the terminal, or
	[b]	download at least one piece of content from a source of content.
9	**[pre]**	An apparatus according to claim 8, and
	[a]	wherein the controller is configured to receive a response that instructs the controller 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.
10	**[pre]**	An apparatus according to claim 8,
	[a]	wherein the controller is configured to receive a response that instructs the controller to download at least one piece of content from the source based upon server status information comprising a listing of at least one available piece of content from the source.
11	**[pre]**	An apparatus according to claim 8,
	[a]	wherein the controller is configured to send the content status such that the network entity determines if the memory of the terminal includes at least one piece of content to delete, and
	[a]	wherein the controller is configured to receive a response that instructs the controller to delete at least one piece of content when the network entity determines that the memory of the terminal includes at least one piece of content to delete.
12	**[pre]**	An apparatus according to claim 11,
	[a]	wherein the controller is configured to send the content status such that the network entity further determines if the source includes at least one available piece of content for the terminal to download,

	[b]	wherein the controller is configured to receive a response that further indicates if the source includes at least one available piece of content, and
	[c]	wherein the controller is further configured to download the at least one available piece of content when the network entity determines that the source includes at least one available piece of content.
13	**[pre]**	A method for controlling a flow of content, the method comprising:
	[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
	[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,
	[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.
14	**[pre]**	A method according to claim 13,
	[a]	wherein sending a response comprises sending a response that instructs the terminal to at least one of delete at least one piece of content from the memory of the terminal, or
	[b]	download at least one piece of content from a source of content.
15	**[pre]**	A method according to claim 14, and
	[a]	wherein sending a response comprises sending a response 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.
16	**[pre]**	A method according to claim 14,
	[a]	wherein sending a response comprises sending a response that instructs the terminal to download at least one piece of content from the source based upon server status information comprising a listing of at least one available piece of content from the source.
17	**[pre]**	17. A method according to claim 14 further comprising:

	[a]	determining if the memory of the terminal includes at least one piece of content to delete,
	[b]	wherein sending a response comprises sending a response that instructs the terminal to delete at least one piece of content when the memory of the terminal is determined to include at least one piece of content to delete.
18	**[pre]**	A method according to claim 17 further comprising:
	[a]	determining if the source includes at least one available piece of content for the terminal to download,
	[b]	wherein sending a response comprises sending a response that further instructs the terminal to download at least one available piece of content when the source is determined to include at least one available piece of content.
19	**[pre]**	A computer-readable storage medium having computer-readable program code portions stored therein, the computer-readable program code portions comprising:
	[a]	a first executable portion configured to receive, 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
	[b]	a second executable portion configured to send, 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,
	[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.
20	**[pre]**	A computer-readable storage medium according to claim 19,
	[a]	wherein the second executable portion is configured to send a response that instructs the terminal to at least one of delete at least one piece of content from the memory of the terminal, or
	[b]	download at least one piece of content from a source of content.
21	**[pre]**	A computer-readable storage medium according to claim 20,

	[a]	wherein the second executable portion is configured to send a response 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.
22	**[pre]**	A computer-readable storage medium according to claim 20,
	[a]	wherein the second executable portion is configured to send a response that instructs the terminal to download at least one piece of content from the source based upon server status information comprising a listing of at least one available piece of content from the source.
23	**[pre]**	A computer-readable storage medium according to claim 20 further comprising:
	[a]	a third executable portion configured to determine if the memory of the terminal includes at least one piece of content to delete,
	[b]	wherein the second executable portion is configured send a response that instructs the terminal to delete at least one piece of content when the second executable portion determines the memory of the terminal includes at least one piece of content to delete.
24	**[pre]**	A computer-readable storage medium according to claim 23,
	[a]	wherein the third executable portion is further configured to determine if the source includes at least one available piece of content for the terminal to download, and
	[b]	wherein the second executable portion is configured to send a response that further instructs the terminal to download at least one available piece of content when the second executable portion determines the source includes at least one available piece of content.

I. INTRODUCTION

Petitioner Roku, Inc. ("Petitioner") respectfully requests *inter partes* review ("IPR") of claims 1-24 ("Claims") of U.S. Patent No. 7,440,559 (EX1001; "'559") assigned to VideoLabs, Inc. ("PO") in accordance with §§311-319 and §42.100 et seq. There is a reasonable likelihood that at least one challenged claim is unpatentable as explained herein. Petitioner requests review of the Claims and judgment finding them unpatentable under 35 U.S.C. §103.

The '559 explains that technological advancements have allowed client devices to be used for a variety of functions, for example video calls and the playback of multimedia applications that are comprised of audio and video clips. '559, 1:22-25. The '559 explains that the standard techniques employed by client devices to retrieve such content may not provide network administrators sufficient control over the flow of content to the client devices. '559, 2:25-35. The '559 thus describes techniques for controlling the flow content to client devices that take into account "user preferences, capabilities of the" client device "and/or previous contents stored or otherwise received by the" client device. '559, 2:40-44.

But controlling content flow based on these types of client characteristics was well-known in the art. For example, Kloba describes techniques for synchronizing content between a client device and a remote server or content provider based on "state information" sent by the client to the remote server to begin the

synchronization process. Kloba, FIG. 63B, 21:19-20, 23:40-45. The remote server responds to the client with "instructions" generated based on the state information to synchronize the content stored on the client with the content stored either on the remote server itself or on a separate content provider. Kloba, FIG. 63B, 19:1-24, 19:64-20:11. Kloba's server system therefore controls the flow of content to client devices, like the '559, and renders obvious the independent claims of the '559.

Additional features claimed in dependent claims include, *e.g.*, particular instructions in the server response to instruct the client to delete content items. To the extent these additional features are not present in Kloba, they are taught by Robbin and a POSITA would have found it obvious to incorporate them into Kloba's system. *See, e.g.,* Robbin, Abstract, [0057], [0066]; *see* §VII.B.2.

The USPTO did not consider Kloba, alone or in combination with Robbin during the '559's prosecution or in any subsequent proceeding.

Accordingly, Petitioner respectfully requests that the Board institute trial and cancel the Claims.

II. MANDATORY NOTICES UNDER 37 C.F.R. §42.8

A. Real Party-In-Interest

Petitioner identifies Roku, Inc. as real party-in-interest. No other party had access to or control over the present Petition, and no other party funded or participated in preparation of the present Petition.

B. Related Matters

The '559 is currently asserted in the following district court proceedings:

Title	No.	Court	Filing Date
VideoLabs, Inc. v. Roku, Inc.	1:23-cv-01136	D. Del.	2023-10-11
VideoLabs, Inc. v. Netflix Inc.	1:22-cv-00229	D. Del.	2022-02-23
Starz Entertainment, LLC v. VL Collective IP, LLC	1:21-cv-01448	D. Del.	2021-10-13

The '559 is also the subject of *Netflix, Inc. v. VideoLabs, Inc.*, IPR2023-00630, which was filed 2023-02-23 and reached a final written decision 2024-10-02.

C. Lead and Back-up Counsel, and Service Information

Lead Counsel	Scott A. McKeown, Reg. No. 42,866	
Backup Counsel	Libbie A. DiMarco (*pro hac vice* forthcoming) Victor Cheung, Reg. No. 66,229 Daniel D. Smith, Reg. No. 71,278	
Service Information	E-Mail	SMcKeown-PTAB@WolfGreenfield.com Elizabeth.DiMarco@WolfGreenfield.com VCheung-PTAB@WolfGreenfield.com DSmith-PTAB@WolfGreenfield.com
	Post & Hand-Delivery	Wolf, Greenfield & Sacks, P.C. 600 Atlantic Avenue Boston, MA 02210-2206
	Telephone	(617) 646-8000
	Facsimile	(617) 646-8646

Petitioner consents to electronic service of documents to the email addresses of the counsel identified above.

III. REQUIREMENTS FOR IPR

A. Grounds for Standing (37 C.F.R. §42.104(a))

Petitioner certifies the '559 is available for IPR and Petitioner is not barred or estopped from requesting IPR on the following grounds. *See* EX1016 (complaint in the related litigation was served on 2023-12-22, less than one year prior to filing this Petition).

B. Identification of Challenge (37 C.F.R. §42.104(b))

Petitioner requests IPR of the Claims and that the Board cancel the same as unpatentable.

1. Specific Art on Which the Challenge is Based

Petitioner relies upon the following prior art:[1]

Name	Ex.	Patent / Publication	Filed	Published	Prior Art Under §102[2]
Kloba	1005	U.S. 6,341,316	2000-04-28	2002-01-22	(e), (b)
Robbin	1006	U.S. 2003/0079038	2002-10-21	2003-04-24	(e), (a)

[1] The '559 was filed 2003-10-22 and does not claim any earlier priority date. *See* '559, Face.

[2] Listed references qualify as prior art under, at least, the listed sections of pre-AIA 35 U.S.C. § 102.

2. Statutory Grounds on Which the Challenge is Based

Petitioner respectfully requests cancellation of the Claims on the following grounds:

Ground	Basis	Claims	Reference(s)
1	§103	1, 7, 13, 19	Kloba
2	§103	2-6, 8-12, 14-18, 20-24	Kloba in view of Robbin

IV. BACKGROUND

A. '559 Patent

The '559 patent is purportedly directed to an improved system and associated terminal, method and computer program product for controlling the flow of content. '559, 2:57-62. As the '559 patent acknowledges, "[d]igital broadband data broadcast networks [were] known," including the goal to achieve "efficient delivery of digital services." '559, 1:58-67, 2:8-11. The specification of the '559 patent admits that the concept of downloading content to client devices was well-known in the art, including when to deliver new pieces of content to the client device and what new pieces of content to deliver. '559, 2:25-39. The '559 patent alleges that "current techniques for downloading content can suffer from inefficient content flow control between the mobile terminal and the server or content provider." '559, 2:47-49.

The ’559 patent purports to solve this alleged problem using “a terminal capable of sending a content status including terminal status information” to a content flow manager, which can control the flow of content to the terminal. Id., 3:10-20. The “content status” includes status information regarding the terminal. Id., 10:60-67. The “terminal status information” includes information that accounts for user preferences, capabilities of the terminal and/or previous content stored by the terminal. *See* ’559, 3:1-4, 12:18-30.

In addition, the ’559 patent discloses that “the control flow manager can be capable of controlling the terminal to download one or more pieces of content from the source of content based upon server status information including a listing of available piece(s) of content from the source.” ’559, 3:31-36. For example, the source of content (such as origin server 24 or digital broadcast receiver 28) is associated with the network entity operating the content flow manager. ’559, 12:37-43.

Based upon the terminal status information and/or the server status information, the content flow manager can control the flow of content to the terminal, such as by instructing the terminal to delete at least one piece of content from the memory of the terminal and/or download at least one piece of content from the source of content. ’559, 3:18-36.

B. Prosecution History of the ’559

The ’559 patent issued from U.S. Patent Appl. No. 10/690,692 (“the ’692 application”), which was filed on October 22, 2003. ’559, Face.

During prosecution of the ’692 application, the Examiner rejected claims 1-24 as being anticipated by U.S. Patent No. 5,450,482 to Chen et al. (“Chen”) in a Non-final Office Action. ’559FH, 90. In response, the applicant amended independent claim 1, adding the limitation “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 consumable by the terminal.” ’559FH, 71.[3] Similarly, the applicant amended independent claims 7, 13, and 19 to recite “wherein the at least one piece of content stored in the memory, and the content for which the network entity is configured to control the flow, comprise multimedia content consumable by the terminal.” ’559FH, 72-75.

The applicant asserted that “Chen discloses a network automatic call distribution system (ACD) for a network including a number of switches interconnecting a number of telephones and operator switches.” ’559FH, 78. In distinguishing the prior art, the applicant contended that “Chen discloses switch status including a listing of switches and services available from those switches[,]”

[3] All emphasis added unless indicated otherwise.

not "a server status including a listing of content available from the source, similar to the claimed invention." '559FH, 79-80 (emphasis in original). Therefore, the applicant contended, Chen purportedly did not "teach or suggest a network entity controlling the flow of content to a terminal based on terminal status information, as well as server status information for a source of content, the server status information including a listing of one or more pieces of content available from the source" as recited in claim 1. '559FH, 78. The applicant also argued that Chen purportedly did not disclose a terminal status including a listing of content stored in memory of the terminal as recited in claims 7, 13 and 19. '559FH, 79-80.

In a Final Office Action, the Examiner maintained the rejection that claims 1-24 are anticipated by Chen. '559FH, 64-65. To overcome the prior art rejection, the applicant amended claims 1, 7, 13 and 19 to recite a terminal that is remote, and argued that the cited prior art purportedly did not disclose "multimedia content." '559FH, 42-47, 51-53. In response to the prior art rejection in the second Non-final Official Action, the applicant argued, without any amendment, that the cited reference, Aubault (U.S. Patent Application Publication No. 2005/0086318), did not qualify as prior art. '559FH, 22-23.

The Examiner subsequently allowed the '690 application, and the '559 patent issued on October 21, 2008. '559, Face.

V. LEVEL OF ORDINARY SKILL IN THE ART

On or before the earliest effective filing date of the '559 (2002-10-22, hereinafter the "Critical Date"), a person of ordinary skill in the art ("POSITA") would have had a bachelor's degree in computer science, computer or electrical engineering, or a related field, and approximately two or more years of experience with digital multi-media content distribution and management and associated system infrastructures. Additional education could substitute for professional experience, and vice versa. McNair, [24]-[25].

VI. CLAIM INTERPRETATION

Claim terms subject to IPR are to be construed in accordance with their ordinary and customary meaning as understood by a POSITA in light of the specification and prosecution history. 37 C.F.R. §42.100(b). Only terms necessary to resolve the controversy need to be construed. *Nidec Motor Corp. v. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013 (Fed. Cir. 2017).

No constructions are necessary in this proceeding at this time, as the prior art discloses the claimed limitations under any interpretation.

Petitioner notes that a claim construction order involving the '559 issued in litigation not involving Petitioner. *See* EX1007. The order construed the following:

Term	Court's Construction
"download" [Claims 2, 4]	"copy and store in memory of the terminal for subsequent use"

VII. GROUNDS OF UNPATENTABILITY

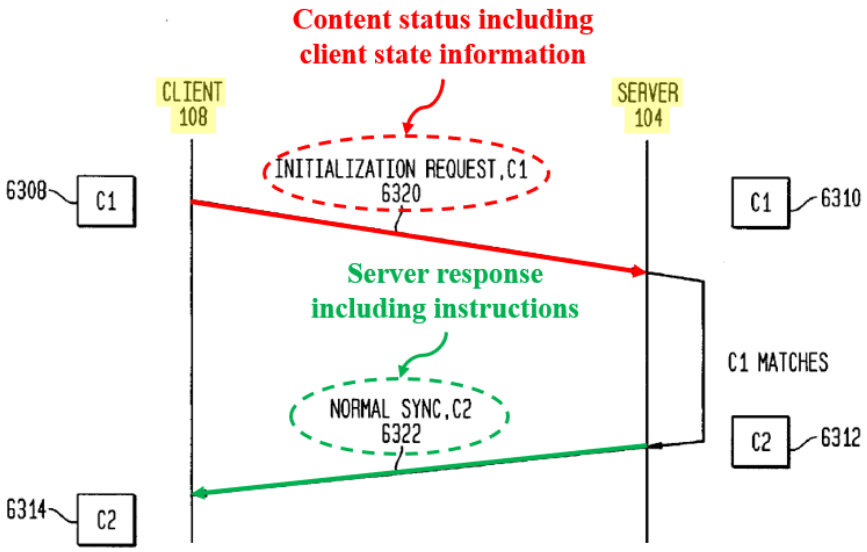
This Petition is supported by the Declaration of Bruce McNair, which describes the prior art's scope and content at the time of the '559. McNair, [46]-[184]. The prior art renders the Claims unpatentable for reasons discussed below.

A. Ground 1: Claims 1, 7, 13, 19 are Rendered Obvious by Kloba

1. Overview: Kloba

Kloba describes a system and method for "synchronizing content between a server and a client based on state information," in which transfers of content (*e.g.*, music, movies, books, photos, etc.) are based on the current state of content stored on a client device. EX1005 ("Kloba"), Title, Abstract, 25:30-32. Kloba seeks to "enabl[e] web content … to be loaded on mobile devices, and for users of mobile devices to operate with Such web content on their mobile devices in an interactive manner while in an off-line mode." Kloba, 1:54-57. Kloba describes "synchronization processes that can collect information from the Internet to a server, and to the client." Kloba, 5:30-31.

For example, Kloba discloses the synchronization process shown in FIG. 63B, annotated below:



Kloba, Detail of FIG. 63B (annotated)

As shown, the client 108 sends request 6320 to the server 104 to begin the synchronization process (annotated in red above). Kloba, 21:19-20, 23:40-45. The request 6320 contains "state information" regarding the state of the client 108, including a "data marker" indicating the state of content stored by the client 108. Kloba, 18:28-33, 19:18-21, 21:19-20, 23:40-45. The server 104 receives the request 6320 and determines a set of "differences" (or "deltas") between the version of content stored by the server 104 and that stored by the client 108 by comparing the received "data marker" to one representing the server's version of the content. Kloba, 19:1-15, 19:37-45. In some cases, the server 104 determines the differences between content stored by an external provider 128 and the client 108, rather than between content stored by the server 104 itself. Kloba, 19:64-20:3.

Based on the determined differences, the server 104 determines a set of instructions for the client 108 to execute in order to bring its stored version of

content up-to-date with that stored by the server 104 or provider 128. Kloba, 19:1-15, 19:64-20:3. The server 104 then transmits a response to the client 108 including the determined instructions, which are then executed by the client 108. Kloba, 19:1-24, 19:64-20:11.

Kloba discloses that the status message received by the server 104 includes state information about the client 108, including a data marker that provides information about the state of content stored on the client 108. Kloba, 18:28-33, 19:18-21. A POSITA would have recognized that Kloba discloses that the data marker identifies at least one piece of content stored on the client 108, because Kloba discloses that the data marker is used to identify differences between pieces of content stored on the client 108 and pieces of content stored on the server 104 or provider 128. McNair, [52]; *see, e.g.*, Kloba, 19:1-15, 19:37-45, 19:66-20:3. The data marker must identify a piece of content stored on the client 108 in order for the server 104 to compare it to the corresponding piece of content stored on the server 104 or provider 128, and identify the differences between the pieces of content. McNair, [52]; *see, e.g.*, Kloba, 19:1-15, 19:37-45, 19:66-20:3. From at least these disclosures, a POSITA would have recognized that Kloba discloses a content status (e.g., the status message from the client 108 to the server 104) including terminal status information (e.g., the data marker identifying content stored on the client 108). McNair, [52].

Kloba further discloses that the server 104 receives status information (*i.e.,* server status information) from a provider 128 (*i.e.,* a source of content) specifying pieces of content available from the provider 128. Kloba, 7:66-8:7, 12:35-40, 21:10-11, FIG. 1A. From at least these disclosures, a POSITA would have recognized that Kloba discloses receiving server status information regarding a source of content (*e.g.*, server 104 receiving status information from provider 128) including a listing of at least one piece of content available from the source (*e.g.*, the information specifying pieces of content available from the provider 128). McNair, [53].

Kloba discloses that the server 104 responds to the content status from the client 108 with a set of instructions to synchronize the client 108 with the provider 128. Kloba, 19:64-20:3; *see also* 19:1-24. The server 104 compiles the set of instructions based on the client state information (*i.e.,* the terminal status information) received from the client 108, and based on the information regarding content available from the provider (*i.e.,* the server status information). Kloba, 19:64-20:3; *see also* 19:1-24. Kloba discloses that the server 104 sends a response to the content status message including the compiled instructions to the client 108. Kloba, 20:1-2, 7-11. From at least these disclosures, a POSITA would have recognized that Kloba discloses the server 104 sending, to the terminal (*e.g.*, client 108), a response to the content status (*e.g.*, the response including the compiled instructions) that instructs the terminal to perform one or more actions to thereby

control the flow of content to the terminal (*e.g.,* the instructions included in the response instruct the client 108 to perform actions on the stored content) based upon the terminal status information and the server status information (*e.g.*, the server 104 compiles the instructions based on the information from the provider 128 and the client state information). McNair, [54]. Kloba further discloses that the client 108, server 104 and provider 128 store various types of multimedia content, including music, images, and movies. Kloba, 4:1-20, 7:13-17, 21:38-44, 25:31-35, 25:38-49, FIG. 1M; *see also* 4:1-20.

Because Kloba discloses a system and method for controlling a flow of content by synchronizing versions of content between clients and remote servers, Kloba discloses the systems and methods for controlling a flow of content in the '559. McNair, [55]. To the extent PO would argue that Kloba's disclosure does not explicitly describe certain aspects of its system in an anticipatory manner, such variations would have been obvious to POSITAs, as discussed in detail below.

And, to the extent PO argues that various relied-on features of Kloba are from distinct and unrelated embodiments—and would allegedly preclude a finding of anticipation—Kloba teaches that its disclosed features are combinable, and a POSITA would have thus found the claims obvious as well. McNair, [56]

For example, Kloba explicitly states that its "synchronization embodiments can be used individually or in combination, as will be appreciated by persons skilled

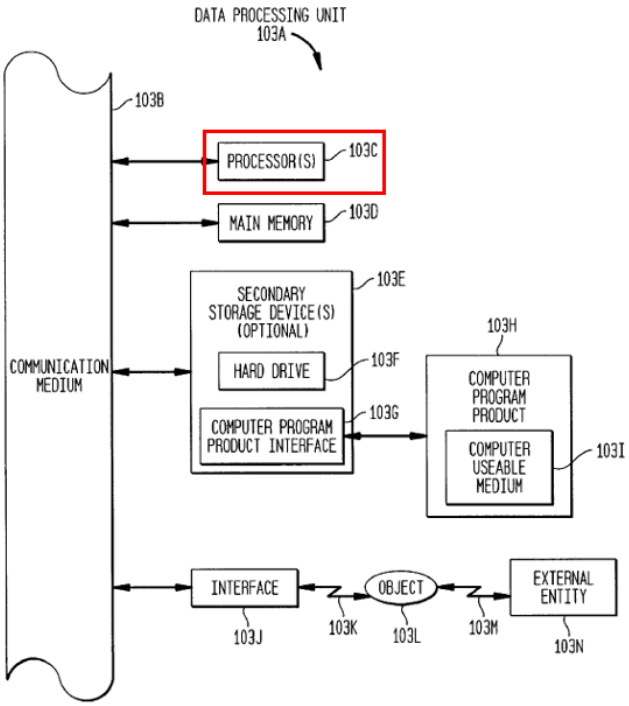
in the relevant art(s)." Kloba, 20:57-60.[4] Indeed, the portions of Kloba generally referenced herein are directly related to each other, and each describe and cross-reference functions and configurations of the same server 104, client 108, and provider 128. *See, e.g.*, Kloba, 19:1-24, 19:64-20:11. Accordingly, even without Kloba's explicitly disclosing the use of the synchronization embodiments in combination, a POSITA would have exercised routine experimentation in implementing a system based on Kloba's disclosure and combined features from related "examples" or "embodiments" to result in a desired system or functionality. *Boston Scientific Scimed, Inc. v. Cordis Corp.*, 554 F.3d 982, 991 (Fed. Cir. 2009) ("Combining two embodiments disclosed adjacent to each other in a prior art patent does not require a leap of inventiveness.") McNair, [57].

Accordingly, claims 1, 7, 13, and 19 are obvious over Kloba. McNair, [58].

2. Claim Charts (1, 7, 13, 19)

'559	Kloba (EX1005)
1[pre]. An apparatus comprising:	Kloba discloses **an apparatus** (*e.g.,* server 104). As discussed in greater detail below, Kloba discloses that "the server 104 maintains a collection of channels," each including "a collection of objects…that can be transferred to a client 108." Kloba, 7:13-16. The objects may include multimedia such as "images, movies, [and] music." Kloba, 7:16-17; McNair, [60].

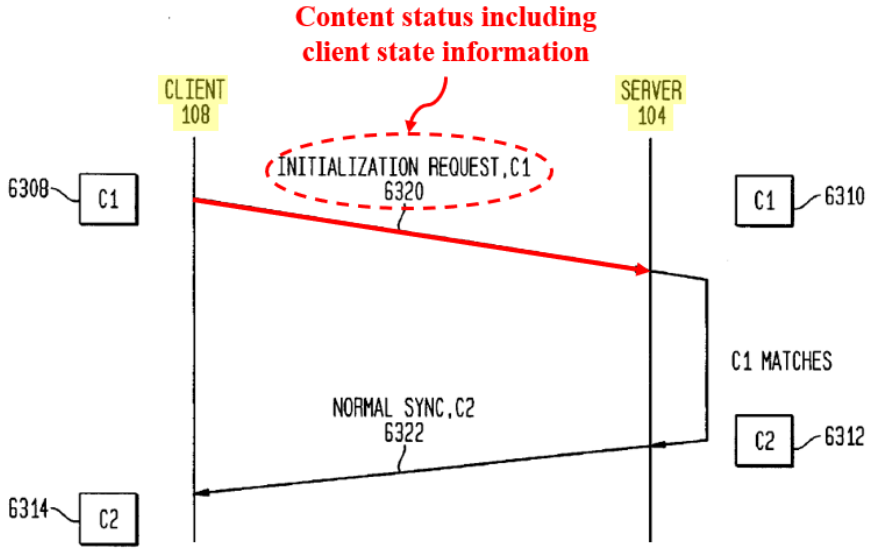
[4] All emphasis added unless otherwise indicated.

’559	Kloba (EX1005)
1[a]. a processor configured to	Kloba discloses that the **apparatus** (*e.g.*, server 104) comprises a **processor.** Kloba discloses that the “server 104” can be implemented as a “data processing unit” including a “processor.” Kloba, 13:4-21, 7:6-12, FIGS. 1A, 1B1; McNair, [61]. • 13:18-21 (“Data processing unit 103A includes one or more processor (s) 103C, and a main memory 103D. Main memory 103D may be RAM, ROM, or any other memory type, or combinations thereof.”) • FIG. 1B1 (showing the data processing unit 103A including processor 103C):  Kloba, Detail of FIG. 1B1 (annotated) • 13:4-6 (“FIG. 1B1 illustrates a block diagram of a data processing unit 103A that can be used to implement the entities shown in FIGS. 1A and 1B”)

’559	Kloba (EX1005)
	• 7:6-12 (“FIG. 1A is a block diagram of a data processing environment 102” that “includes a server 104[.]”)
1[b.i]. receive, from a terminal located remote from the apparatus, a content status including terminal status information, and	Kloba discloses that the processor of server 104 is configured to **receive, from a terminal located remote from the apparatus, a content status including terminal status information** (*e.g.*, the server 104 receives state information from the client 108 that includes information regarding content stored by the client 108 including a data marker or “content status including terminal status information.”). Kloba discloses that the client 108[5] initializes a content synchronization session with the server 104 by transmitting a status message to the server 104. Kloba, 18:28-33, 21:19-31, 22:15-26, 23:40-45, FIG. 63B. The client 108 transmits the status message to the server 104 via a wireless or wired network. Kloba, 12:6-13, FIGS. 1A, 1V. The client 108 encodes and transmits the status message over the network using Hypertext Transfer Protocol (HTTP) and Transmission Control Protocol (TCP) / Internet Protocol (IP). Kloba, 12:6-13, 22:15-21. Kloba discloses that the server 104 receives the status message from the network. Kloba, 23:40-45. From at least these disclosures, a POSITA would have recognized that Kloba discloses that the client

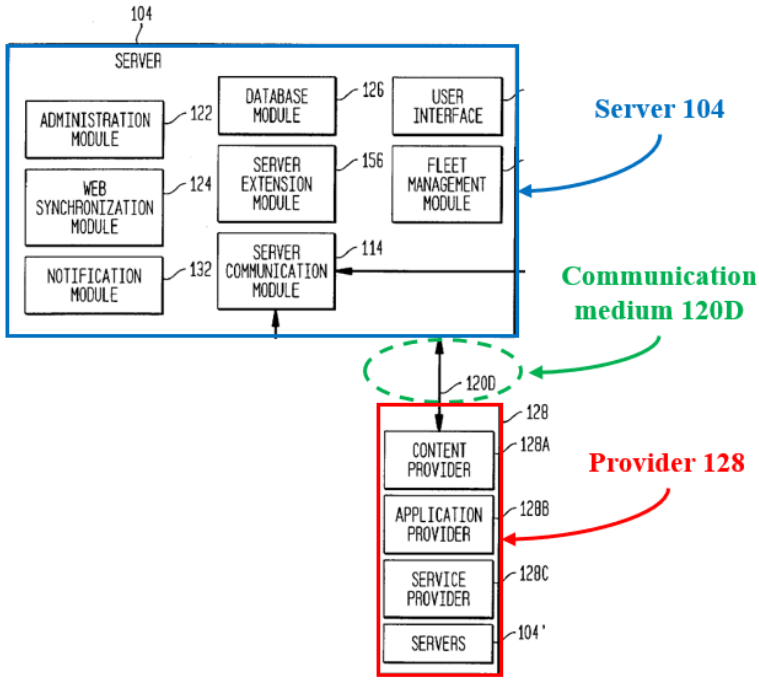
[5] Kloba uses the terms “client” and “device” interchangeably. *See, e.g.,* Kloba, 10:42-48 (“[T]he devices 106 include software, hardware, and/or combinations thereof related to client functionality (such client functionality is described herein). When a device 106 includes such software, hardware, and/or combinations thereof, the device 106 is referred to herein as a client 108.”).

’559	Kloba (EX1005)
	108 is located remote from the server 104. McNair, [64]-[65]. In addition, Kloba teaches that the state information in the content status message can include user preferences, such as particular channels to which the user of the client devices is subscribed. Kloba, 34:23-27; McNair, [66]. The '559 patent explains that “terminal status information” can include information that accounts for user preferences, capabilities of the terminal and/or previous content stored by the terminal. *See* ’559, 3:1-4, 12:18-30. As discussed above (*see* §VII.A.1), Kloba discloses a **content status** (*e.g.*, the status message from the client 108 to the server 104) including terminal status information (*e.g.*, the data marker identifying content stored on the client 108, and the user preferences associated with a user of the client device). McNair, [67]. Regarding the server 104 receiving the status message from the remote client including the client state information: • FIG. 1A (showing the server 104 in communication with the remote client 108 via communication medium 102B) • FIG. 63B (showing the content status transmitted from the client 108 to the server 104):

’559	Kloba (EX1005)
	 Kloba, Detail of FIG. 63B (annotated) • 21:19-20 (“[C]lient 108 provides state information regarding the nature of its resources.”) • 23:40-45 (“[C]lient 108 sends state information to server 104 via client communications module 110…. In step 176B, server communications module 114 receives client's state information[.]”) • 12:6-13 (“Client communications module 110 enables the client 108 to interact with external entities, such as server 104. In embodiments, the client communications module 110 enables TCP/IP traffic, although the invention is not limited to this example. More generally, the client communications module 110 enables communication over any type of communication medium 120, such as wireless, wired, etc[.]”) • FIG. 63B (showing client 108 sending an initialization request including a data marker C1 to server 104) • FIG. 3C (showing example synchronization process between client 108 and server 104) Regarding the client state information:

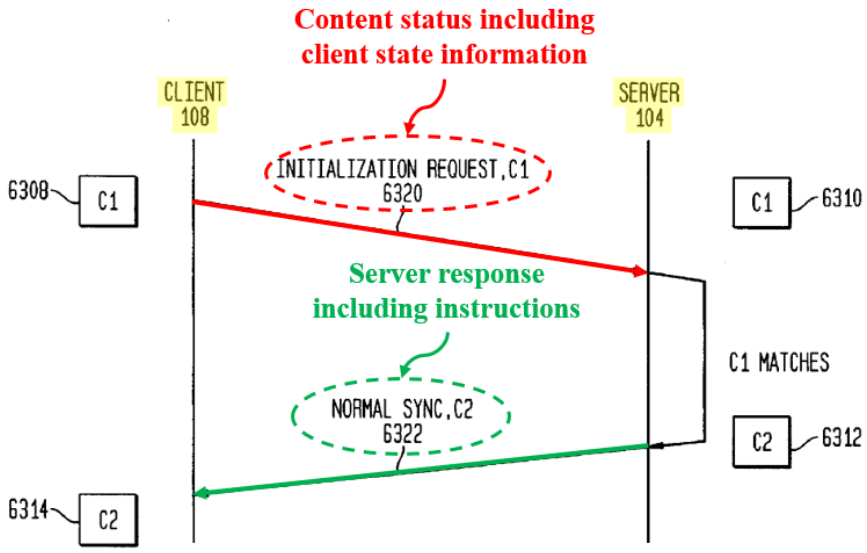
’559	Kloba (EX1005)
	• 18:28-33 (“[C]lient communication module 110 of client 108 initializes a synchronization session (step 170A). Client control module 110 of client 108 sends a current data marker C1 to web synchronization module 124 on server 104 (step 170B).” • 19:18-21 (“In one embodiment, a data marker is a synchronization token which is specifically constructed **to provide information about the state of information on a client**.”) • 19:1-15 (“In step 170F, the server 104 compares the latest data marker received from the client 108 (C2 in the example of 63B) with ones stored in the server 104 for the client 108. Essentially, the server 104 attempts in step 170F to “roll back” to a previous known state of client 108…. [T]he server 104 determines what instructions are needed to cause the client 108 to roll back to the known state associated with data marker C2 identified in step 170F, and what instructions are needed to cause the client 108 to move forward from the previous state associated with data marker C2 to the current state associated with data marker C3.”) • 19:37-45: (“Control module 142 identifies the deltas in the client databases identified by server 104 during initialization in step 168B (step 172A). In one embodiment of the present invention, a delta is a set of differences between versions of content or, more generally, objects (*i.e.*, different versions of the same pages, documents, links, images, applications, services, etc.). In other words, deltas are sets of differences in the state of the objects currently being offered and the state of the objects in client 108.”) • 34:23-27 (“When in the off-line mode, a user of the client 108 can elect to subscribe to channels listed in the channel subscription page. In an embodiment, the

’559	Kloba (EX1005)
	selected channels are loaded on the client 108 during the next synchronization operation.”)
1[b.ii]. 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,	Kloba discloses that the processor of the server 104 is **configured to receive server status information regarding a source of content comprises a listing of at least one piece of content available from the source** (*e.g.*, the server 104 receives information regarding the content available from a provider 128). Kloba discloses that the server 104 receives information from a provider 128 specifying pieces of content available from the provider 128. Kloba, 7:66-8:7, 12:35-40, 21:10-11; McNair, [71]-[73]. • 12:35-40 (“Providers 128 are sources of various types of objects, such as … content (content providers 128A)… Providers 128 may also include servers 104 (similar to server 104), which may provide objects such as but not limited to content, applications, services, etc.”) • FIG. 1A (showing provider 128, including content provider 128A, in communication with server 104 via communication medium 120D):

’559	Kloba (EX1005)
	 Kloba, Detail of FIG. 1A (annotated) • 21:10-11 (“In step 314, server 104 receives information regarding the sets of content available from provider(s) 128.”) • 7:66-8:7 (“During a synchronization process, the server 104 loads a device 108 with the channels associated with the client 108. Generally, the server 104 does this by obtaining from providers 128 the objects defined by the channels, and causing those objects to be stored on the client 108. Thus, during the synchronization process, the server 104 will load the client 108 with the selected channels. More particularly, the server 104 will load the client 108 with the objects associated with the channels.”) *See also:* • 4:1-20 (Table 1 listing examples of “Internet content” that can be synchronized using Kloba’s techniques, including “Multimedia: Images (e.g., JPEG, GIF, PNG, vector graphics, etc.),” “Audio Files (e.g.

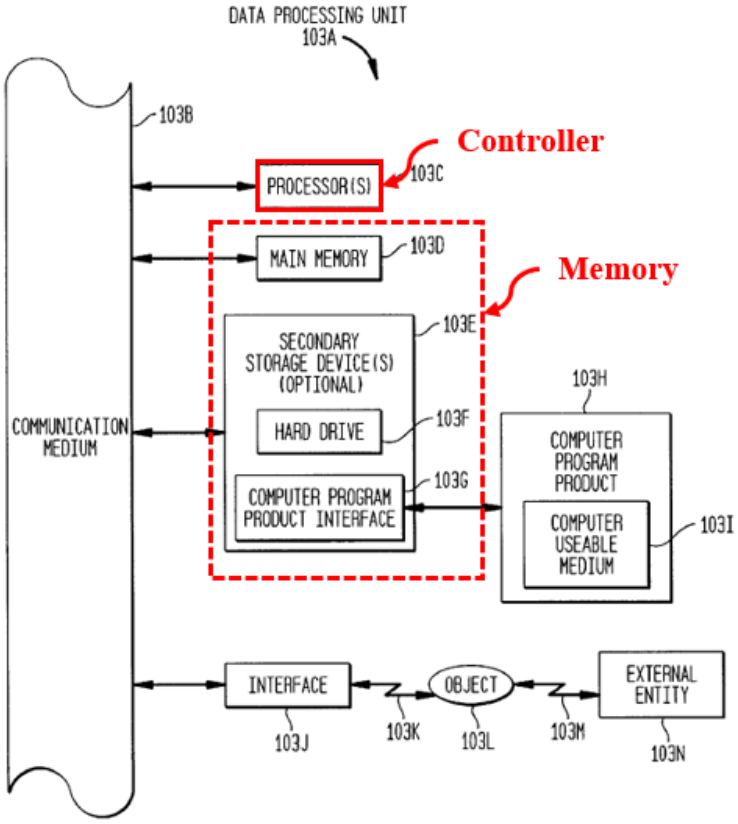
’559	Kloba (EX1005)
	MP3),” “Video (e.g. AVI),” and “Streaming Content: Voice/Data/Video”)
1[c]. wherein the processor is configured to send, to the terminal, a response to the content status that instructs the terminal to perform one or more actions to thereby control the flow of content to the terminal based upon the terminal status information and the server status information, and	Kloba discloses that **the processor** (*e.g.*, the processor in server 104) **is configured to send, to the terminal** (*e.g.*, the client 108)**, 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** (*e.g.*, the message from server 104 to client 108 including instructions to synchronize the client 108) **based upon the terminal status information and the server status information** (*e.g.*, the instructions compiled based on the state information received from the client 108 and the information regarding the available content received from the provider 128). McNair, [75]-[76]. As discussed at 1[b.i], Kloba discloses that the server 104 receives the content status from the client 108 (*i.e.,* the terminal) including the client state information (*i.e.,* the terminal status information). As discussed at 1[b.ii], Kloba discloses that the server 104 also receives, from provider 128 (*i.e.,* the source of content), information regarding content available from the provider (*i.e.,* the server status information). McNair, [77]. Kloba discloses that the server 104 responds to the content status from the client 108 with a set of instructions to synchronize the client 108 with the provider 128. Kloba, 19:64-20:3; *see also* 19:1-24. Kloba further discloses that the server 104 compiles the set of instructions based on the client state information (*i.e.,* the terminal status information) received from the client 108, and based on the information regarding content available from the provider (*i.e.,* the server status information). Kloba, 19:64-20:3; *see also* 19:1-24. Kloba discloses that the server 104 sends a response to the content status message including the compiled instructions to the client 108. Kloba, 20:1-2, 7-11. Kloba further discloses that the instructions can direct the

’559	Kloba (EX1005)
	client to load new content not previously stored on the client. Kloba, 7:14-17 (“a channel comprises a collection of objects” which are “any entity that can be transferred to a client 108, such as but not limited to content, applications, services, images, movies, music, links, etc.”), 25:31-37, 34:25-27 (“the selected channels are loaded on the client 108 during the next synchronization operation”); *see also* 7:66-8:7; McNair, [78]. Regarding the synchronization instructions: • 19:37-45 (“Control module 142 identifies the deltas in the client databases identified by server 104 during initialization in step 168B (step 172A). In one embodiment of the present invention, a delta is a set of differences between versions of content or, more generally, objects (i.e., different versions of the same pages, documents, links, images, applications, Services, etc.). In other words, deltas are sets of differences in the state of the objects currently being offered and the state of the objects in client 108.”) • 19:64-20:3 (“Synchronization modules 155 synchronize the deltas from client 108 with providers 128 (step 172C). Based on the information from provider(s) 128, synchronization modules 155 compile instructions to synchronize the client 108 with providers 128 (step 172D). Synchronization module 155 sends such instructions to client 108, plus updated data marker (step 172E).”) • 25:31-37 (“3.3.4. Syncing Music, Movies, Books, Photo Albums, and Other Collections of Objects… The invention supports channels which comprise web sites having collections of objects, such as collections of music, images, books**, movies,** applications, services, etc. By selecting such a channel, the client 108 can be populated with such collections of objects.”)

’559	Kloba (EX1005)
	• 34:23-27 (“When in the off-line mode, a user of the client 108 can elect to subscribe to channels listed in the channel subscription page. In an embodiment, the selected channels are loaded on the client 108 during the next synchronization operation.”) Regarding the server 104 transmitting the response including the synchronization instructions to the client 108: • FIG. 63B (showing the server 104 transmitting the response to the client 108):  Kloba, Detail of FIG. 63B (annotated) • 20:7-11 (“The instructions are transmitted via any reliable transport medium. For example, in one embodiment, HTTP is used. Control module 142 on the client 108 then executes the instructions (step 172F).”) • 12:6-10 (“Client communications module 110 enables the client 108 to interact with external entities, such as server 104. In embodiments, the client

’559	Kloba (EX1005)
	communications module 110 enables TCP/IP traffic, although the invention is not limited to this example.”) *See also:* • 19:1-24 (describing the process of comparing pieces content stored on the client 108 with pieces of content stored on the server 104 or the provider 128) • FIG. 63B (showing the above process)
1[d]. 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.	Kloba discloses that **the at least one piece of content available from the source** (*e.g.,* the content specified in the information on content available from provider 128)**, and the content for which the processor is configured to control the flow** (*e.g.,* the content stored on client 108)**, comprise multimedia content** (*e.g.,* the content stored on the client 108 and the content available from the provider include music, images, and movies). McNair, [83]. Kloba discloses that both the client 108 and the provider 128 store various types of content, including music, images, and movies. Kloba, 7:13-17, 21:38-44, 25:31-35, 25:38-49, FIG. 1M; *see also* 4:1-20; McNair, [84]. • 4:1-20 (Table 1 listing examples of “Internet content” that can be synchronized using Kloba’s techniques):

’559	Kloba (EX1005)
	TABLE 1. Internet Content Internet content includes but is not limited to: HTML JavaScript ™ Channels Java ™ ActiveX Multimedia: Images (e.g., JPEG, GIF, PNG, vector graphics, etc.) Audio Files (e.g. MP3) Video (e.g. AVI) Streaming Content: Voice/Data/Video Binary files XML Applications Data Objects Documents Anything that can be delivered via a “browser” Kloba, 4:1-10 (Table 1) (annotated) • 25:31-37 (“3.3.4. Syncing Music, Movies, Books, Photo Albums, and Other Collections of Objects… The invention supports channels which comprise web sites having collections of objects, such as collections of music, images, books**, movies,** applications, services, etc. By selecting such a channel, the client 108 can be populated with such collections of objects.”)
7[pre]. An apparatus comprising:	*See* 1[pre].
7[a]. a controller operable with a terminal including a memory configured to store	Kloba discloses **a controller operable with a terminal** (*e.g.*, the processor 103C in the client 108) **including a memory configured to store at least one piece of content** (*e.g.*, the memory 103D and secondary storage devices 103E in the client 108). McNair, [88].

’559	Kloba (EX1005)
at least one piece of content	Kloba discloses that the client 108 (*i.e.,* a terminal, *see* 1[b.i]) includes a processor 103C (*i.e.,* a controller), a memory 103D, and secondary storage devices 103E. Kloba, 13:4-33, FIG. 1B1. Kloba further discloses that the client 108 stores pieces of content. Kloba, 8:1-3, 25:31-37, 25:28-49. From these disclosures, a POSITA would have recognized that the client 108 stores these pieces of content in the memory 103D and/or the secondary storage devices 103E. McNair, [89]. Regarding the processor and memory included in the client 108: • FIG. 1B1 (showing a data processing unit 103A including a processor 103C, a memory 103D, and secondary storage devices 103E):  Kloba, Detail of FIG. 1B1 (annotated)

’559	Kloba (EX1005)
	• 13:4-6 (“FIG. 1B1 illustrates a block diagram of a data processing unit 103A that can be used to implement the entities shown in FIGS. 1A and 1B.”) • 7:6-12 (“FIG. 1A is a block diagram of a data processing environ ment 102 according to an embodiment of the invention. The data processing environment 102 includes a server 104 (although only one server 104 is shown, in practice the data processing environment 102 may include a plurality of servers), one or more devices 106, one or more adapters 118, and one or more providers 128.”) Regarding content stored in the memory of client 108: • 8:1-3 (“Generally, the server 104 does this by obtaining from providers 128 the objects defined by the channels, and causing those objects to be stored on the client 108.”) • 25:31-37 (“3.3.4. Syncing Music, Movies, Books, Photo Albums, and Other Collections of Objects… The invention supports channels which comprise web sites having collections of objects, such as collections of music, images, books, movies, applications, services, etc. By selecting such a channel, the client 108 can be populated with such collections of objects.”) • 25:38-49 (“For example, if a channel having a collection of music is selected, then it is possible to turn the client 108 into a "jukebox' once the music collection is stored on the client 108 during the synchronization process. Similarly, a client 108 can become a photo album, a book library, a movie theater, an application library, etc., by selecting appropriate channels. This process is represented by FIG.1M. It is noted that this process is applied to collections of music, but it is also applicable to collections of any types of objects. It is also noted that a given channel

’559	Kloba (EX1005)
	may have combinations of different types of objects, such as combinations of music, movies, applications, images, services, etc.”)
7[b]. wherein the controller is configured to send a content status including terminal status information comprising a listing of at least one piece of content stored in the memory,	*See* 1[b.i] (the client 108 sends the content status including a data marker specifying at least one piece of content for synchronization to the server 104)
7[c]. wherein the controller is configured to send the content status to a remote network entity, and	*See* 1[b.ii] (the client 108 sends the content status to the remote server 104 over a network)
7[d]. receive a response to the content status from the network entity that instructs the controller to perform one or more actions to thereby control a flow of content to the terminal based upon the terminal	*See* 1[c] (the client 108 receives the response to the content status sent by the server 104)

’559	Kloba (EX1005)
status information, and	
7[e]. wherein the at least one piece of content stored in the memory, and the content for which the network entity is configured to control the flow, comprise multimedia content.	*See* 1[d].
13[pre]. A method for controlling a flow of content, the method comprising:	Kloba discloses **a method for controlling a flow of content** (*e.g.*, a method for synchronizing content stored on a client 108 with content stored by remote entities including server 104 and provider 128). McNair, [97]. As previously discussed with respect the Claim 1, Kloba describes techniques for synchronizing content between client 108 and remote network entities, such as server 104 and provider 128. Kloba, Abstract. Content objects are organized into channels, which a client 108 may select to cause the content objects therein to be stored on the client 108 and synchronized with versions of the channels stored on remote network entities (*e.g.*, server 104 and provider 128). Kloba, 7:13-17; McNair, [98]. • Abstract (“Described herein are systems, methods, computer program products, and combinations and sub-combinations thereof, for enabling web content (as well as other objects) to be loaded on mobile devices (as well as other types of devices), and for users of mobile devices to operate with such web content on

’559	Kloba (EX1005)
	their mobile devices in an interactive manner while in an off-line mode.”) • 7:13-17 (“Generally, the server 104 maintains a collection of channels. In an embodiment, a channel comprises a collection of objects. An object is any entity that can be transferred to a client 108, such as but not limited to content, applications, services, images, movies, music, links, etc.”)
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	*See* 1[b.i].
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	*See* 1[c].

’559	Kloba (EX1005)
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.	*See* 1[d].
19[pre]. A computer-readable storage medium having computer-readable program code portions stored therein, the computer-readable program code portions comprising:	Kloba discloses **a computer-readable storage medium having computer-readable program code portions stored therein** configured to implement the techniques described above with respect to Claims 1, 7, and 13. Kloba discloses that server 104 includes “computer usable mediums [sic]” that store control logic (software) to implement the techniques described with respect to the claims above. Kloba, 13:22-47; McNair, [104]. • 13:34-39 (“The computer program products 103H include computer useable mediums in which objects may be stored, such as but not limited to optical mediums, magnetic mediums, etc. Control logic or software may be stored in main memory 103D, secondary storage device(s) 103E, and/or computer program products 103H.”) • 13:40-48 (“[T]he term “computer program product’ refers to any device in which control logic (software) is stored, so in this context a computer program product could be any memory device having control logic

’559	Kloba (EX1005)
	stored therein. The invention is directed to computer program products having stored therein software that enables a computer/ processor to perform functions of the invention as described herein.”)
19[a]. a first executable portion configured to receive, 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	*See* 1[b.i]
19[b]. a second executable portion configured to send, 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	*See* 1[c].

’559	Kloba (EX1005)
upon the terminal status information,	
19[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.	*See* 1[d].

B. Ground 2: Claims 2-6, 8-12, 14-18, 20-24 are Obvious Over Kloba in view of Robbin

1. Overview: Robbin

Robbin describes techniques for “interaction between a host computer and a media player, such as automatic synchronization of media contents stored on a media player with media contents stored on a host computer[.]” Robbin, Abstract. Robbin discloses that “management of media items residing on a media player can be performed at and by a host computer for the media player.” Robbin, Abstract. For example, if “a particular media item is resident on the media player … but is not resident on the” host computer, “the particular media item can be … removed (deleted) from the media player” at the instruction of the host computer. Robbin, [0036], [0057].

Robbin describes that the media player "has limited or no capability to manage media items on the media player[.]" Robbin, [0066]. Thus, a "management module … within the host computer … can indirectly manage the media items residing on the media player[.]" Robbin, [0066]. For example, "to 'delete' a media item from the media player 704, the management module 706 serves to identify the media item to be deleted from the media store 708 and then causes the identified media item to be deleted from the media player 704." Robbin, [0066].

Robbin further teaches that the synchronization process "is performed by a media device that interacts with a host computer over a network." Robbin, [0045].

Robbin further describes that the synchronized "media items" can include multimedia content, such as "audio items (e.g., audio files or songs)," "videos (e.g., movies) or images (e.g., photos)[.]" Robbin, [0090].

2. Motivation: Kloba in view of Robbin

Kloba and Robbin are in the same field as the '559—*e.g.*, digital content distribution and management—and reasonably pertinent to the problem alleged therein—*e.g.*, controlling the transfer of content between networked devices (*e.g.*, between Robbin's "host computer" and "media player"). '599, 1:8-13, 2:40-53; *see* Kloba, Abstract, 1:53-57, 19:1-45 (describing techniques to synchronize content between a client and a remote server or content provider); Robbin, Abstract, [0036], [0057] (describing techniques to synchronize content between a host computer and

a media player); *see also* [0045] (describing that during the synchronization process the "media device…interacts with [the] host computer ***over a network***"); McNair, [114].

As explained in §VII.A.1, Kloba discloses a system in which a server 104 synchronizes a set of "channels" each containing a collection of content items—*e.g.*, "music, images, books, movies, applications, services, etc."—to a client 108. *See*, *e.g.*, Kloba, 25:31-37. Kloba discloses that the client 108 provides the server 104 with information describing the state of its locally stored content, which the server 104 compares with content stored by the server 104 itself or by a provider 128. *See, e.g.*, Kloba, 18:28-33, 21:19-31, 22:15-26, 23:40-45, 19:37-45, FIG. 63B. If the content stored by the client 108 is out of sync—*e.g.*, because the content at the server 104/provider 128 has been modified, new content has been added, or pieces of content deleted—the server 104 compiles a set of instructions for execution by the client 108 to synchronize the content. *See, e.g.*, Kloba, 19:1-20:3. To the extent Kloba generally discloses the server 104 instructing the client to perform actions on its stored content to control content flow, (*see*, *e.g.*, Kloba, 19:1-20:3, claim 1) but leaves to the POSITA implementation details for situations where content is deleted from the server 104 / provider 128 (as claimed in claims 2-6, 8-12, 14-18, and 20-24 of the '599), a POSITA would have been motivated to modify Kloba to include

those implementation details, taught by Robbin, for the reasons below. McNair, [115].

A POSITA would have been motivated to implement a content deletion mechanism for Kloba's content synchronization process based on the known techniques described in Robbin. McNair, [116]. As discussed above, Robbin describes that "management of media items residing on a media player can be performed at and by a host computer for the media player." Robbin, Abstract. For example, if "a particular media item is resident on the media player … but is not resident on the" host computer, "the particular media item can be … removed (deleted) from the media player" at the instruction of the host computer. Robbin, [0036], [0057]. Robbin, like Kloba, discloses that the host computer performs a comparison between content stored on the media player and that stored on the host computer to determine differences between the sets of stored content as part of the synchronization process. Robbin, [0059]; McNair, [116].

A POSITA would have been motivated to modify Kloba to implement a content deletion instruction based on Robbin's teachings in order to accommodate scenarios where content items that are stored by a client 108 have been deleted from the server 104 / provider 128. McNair, [117]. That is, where Kloba discloses a general content synchronization scheme based on identifying differences or "deltas" between two versions of a content item, a POSITA would have additionally

recognized a need to incorporate a mechanism for handling situations where the server 104 / provider 128 version of the content item has been deleted, such as that taught by Robbin. *See* Robbin, Abstract, [0057]-[0059]; McNair, [117]. A POSITA would have been motivated to incorporate such a deletion mechanism to ensure that the content items stored by the client 108 are the most up to date by removing outdated objects from a client to ensure users are accessing the most current content, or in the case of media subscription models, is still within a license (i.e., rental) term. Kloba teaches a broad applicability to movie, book, and audio content, which is well known to benefit from more granular DRM controls to ensure copyright protections. *See, e.g.*, Ginter, [0191], [2234]-[2238], [2472] (describing multimedia content rental and deleting content after expiration of DRM license defining the rental period); Van Wie, 12:28-36 (describing multimedia content rental and preventing playback after expiration of DRM license defining the rental period); McNair, [117]. Thus, a POSITA would have recognized the well-known benefits of incorporating such a deletion mechanism given the broad content applicability of Kloba, and been motivated to perform the proposed modification to obtain those benefits. McNair, [117].

Furthermore, the implementation of the deletion mechanism above would have been, at least, the application of known techniques (*e.g.*, Robbin's deletion mechanism for content items) to a known system ready for improvement (*e.g.*, Kloba

content synchronization system) to yield predictable results (*e.g.*, deleting content items from clients in a content delivery system, a result taught by Robbin itself). *See* MPEP §2143 (citing *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 415-421 (2007)); McNair, [118].

A POSITA would have had a reasonable expectation of success in implementing a deletion mechanism in Kloba's content synchronization process because Robbin teaches a system operating in the proposed manner. *See, e.g.*, Robbin, Abstract, [0036], [0045], [0057]-[0059], [0066]; McNair, [119].

3. Claim Charts (2-6, 8-12, 14-18, 20-24)

'559	Kloba (EX1005) + Robbin (EX1006)
2[pre]. An apparatus according to claim 1,	*See* Ground 1, Claim 1.
2[a]. wherein the terminal comprises a memory, and	*See* Ground 1, 7[a].
2[b]. wherein the processor is configured to send, to the terminal, a response to the content status that instructs the terminal to at least one of delete at	As previously discussed at Ground 1, 1[c], Kloba discloses that **the processor** (*e.g.*, the processor in server 104) **is configured to send, to the terminal** (*e.g.*, the client 108)**, a response to the content status that instructs the terminal to perform one or more actions** (*e.g.*, the message from server 104 to client 108 including instructions to synchronize the client 108). McNair, [123]. Robbin discloses a response that **instructs the terminal to delete at least one piece of content from the memory of the terminal** (*e.g.*, response from the host computer

’559	Kloba (EX1005) + Robbin (EX1006)
least one piece of content from the memory of the terminal, or	instructing the media player to delete media items). McNair, [124]. As discussed in at least §VII.B.2, a POSITA would have been motivated to modify the instructions in Kloba’s synchronization process to include content deletion functionality based on the teachings of Robbin, for example, to ensure that the content items stored by the client 108 are the most up to date by removing outdated items, or, in the case of media subscription models, to ensure that content items are removed at the end of a license (i.e., rental) term. McNair, [125]. As discussed above, Robbin discloses that a host computer identifies pieces of content (*e.g.*, “media items”) that are stored on a media device (*i.e.,* a **terminal**) but not on the host computer. Robbin, [0057]. Robbin discloses that the host computer then instructs the media device to delete those identified media items from its memory. Robbin, [0057], [0066]; McNair, [126]. • Robbin, Abstract (“Improved techniques for interaction between a host computer (e.g., personal computer) and a media player are disclosed. According to one aspect, interaction between a host computer and a media player, such as automatic synchronization of media contents stored on a media player with media contents stored on a host computer, can be restricted. According to another aspect, management of media items residing on a media player can be performed at and by a host computer for the media player.”) • Robbin, [0045] (“The media device synchronization processing 500 is performed by a media device that interacts with a host computer over a network.”) • Robbin, [0057] (“Additionally, although not illustrated in FIGS. 6A and 6B, according to another embodiment, the host computer synchronization

’559	Kloba (EX1005) + Robbin (EX1006)
	processing 600 at operation 614 can also identify those of the media items on the media device that are not on the host computer. Then, the host computer can operate to interact with the media device to remove (e.g., **delete**) those media items stored on the media device that are not stored at the host computer.”) • Robbin, [0066] (“[I]n one embodiment, the media player 704 has limited or no capability to manage media items on the media player 704. However, the management module 706 within the host computer 702 can indirectly manage the media items residing on the media player 704. For example, … to “delete” a media item from the media player 704, the management module 706 serves to identify the media item to be deleted from the media store 708 and then causes the identified media item to be deleted from the media player 704.”) *See also:* Robbin, [0058]-[0059] (describing comparing content stored at host computer and media device to determine differences)
2[c]. download at least one piece of content from the source.	As previously discussed at Ground 1, 1[c], Kloba discloses that **the processor** (*e.g.*, the processor in server 104) **is configured to send, to the terminal** (*e.g.*, the client 108)**, a response to the content status that instructs the terminal to perform one or more actions** (*e.g.*, the message from server 104 to client 108 including instructions to synchronize the client 108). McNair, [129]. Kloba further discloses that the instructions included in the response include an instruction to **download at least one piece of content from the source** (*e.g.*, the server 104 instructs the client 108 to synchronize its content with provider 128). McNair, [130].

’559	Kloba (EX1005) + Robbin (EX1006)
	Kloba describes that server 104 compiles instructions to synchronize the client 108 with a provider 128 (a content source). Kloba, 19:63-20:3. A POSITA would have understood that such instructions would include an instruction for the client to download the content to be synchronized from the provider 128. McNair, [131]; *see, e.g.*, Kloba, 19:1-20:3. • Kloba, 19:63-20:3 (“Based on the information from provider(s) 128, synchronization modules 155 compile instructions to synchronize the client 108 with providers 128 (step 172D). Synchronization module 155 sends such instructions to client 108, plus updated data marker (step 172E).
3[pre]. An apparatus according to claim 2,	*See* Claim 1.
3[a]. wherein the terminal status information comprises a listing of at least one piece of content stored in the memory of the terminal, and	*See* 1[b.i].
3[b]. wherein the processor is configured to send, to the terminal, a	*See* 2[b].

’559	Kloba (EX1005) + Robbin (EX1006)
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.	
4[pre]. An apparatus according to claim 2,	*See* Claim 2.
4[a]. wherein the server status information comprises a listing of at least one piece of available content from the source, and	*See* Ground 1, 1[b.ii].
4[b]. wherein the processor is configured to send, to the terminal, a response to the content status that instructs the terminal to	*See* 1[b.ii], 2[c]. *See, e.g.*, Kloba, 14:64-15:2 (“In step 208, the objects retrieved in the preceding steps are compared with the objects already cached on device 106. ***Server 104 determines the set of changes that have occurred*** between the retrieved objects and the objects already cached on device 106 in step 210. Only the set of changes determined in step 210 are transmitted to device 106.”)

’559	Kloba (EX1005) + Robbin (EX1006)
download at least one piece of content from the source based upon the listing of at least one available piece of content from the source.	
5[pre]. An apparatus according to claim 2,	*See* Claim 2.
5[a]. wherein the processor is configured to determine if the memory of the terminal includes at least one piece of content to delete, and	*See* 2[b].
5[b]. wherein the processor is configured to send, to the terminal, a response to the content status that instructs the terminal to delete at least one piece of content when the processor	*See* 3[b].

’559	Kloba (EX1005) + Robbin (EX1006)
determines that the memory of the terminal includes at least one piece of content to delete.	
6[pre]. An apparatus according to claim 5,	*See* Claim 5.
6[a]. wherein the processor is further configured to determine if source includes at least one available piece of content for the terminal to download, and	*See* 4[b].
6[b]. wherein the processor is configured to send, to the terminal, a response to the content status that instructs the terminal to download at least one available piece of content when the processor	*See* 2[c].

’559	Kloba (EX1005) + Robbin (EX1006)
determines that the source includes at least one available piece of content for the terminal to download.	
8[pre]. An apparatus according to claim 7,	*See* Claim 7.
8[a]. wherein the controller is configured to receive a response that instructs the controller to at least one of delete at least one piece of content from the memory of the terminal, or	*See* 2[b].
8[b]. download at least one piece of content from a source of content.	*See* 2[c].
9[pre]. An apparatus according to claim 8,	*See* Claim 8.
9[a].	*See* 3[b].

’559	Kloba (EX1005) + Robbin (EX1006)
and wherein the controller is configured to receive a response that instructs the controller 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.	
10[pre]. An apparatus according to claim 8,	*See* Claim 8.
10[a]. wherein the controller is configured to receive a response that instructs the controller to download at least one piece of content from the source based upon server status information comprising a listing of at least one available	*See* 6[b].

’559	Kloba (EX1005) + Robbin (EX1006)
piece of content from the source.	
11[pre]. An apparatus according to claim 8	*See* Claim 8.
11[a]. wherein the controller is configured to send the content status such that the network entity determines if the memory of the terminal includes at least one piece of content to delete, and	*See* 5[a].
11[b]. wherein the controller is configured to receive a response that instructs the controller to delete at least one piece of content when the network entity determines that the memory of the terminal includes at least one piece of content to delete.	*See* 5[b].

’559	Kloba (EX1005) + Robbin (EX1006)
12[pre]. An apparatus according to claim 11,	*See* Claim 11.
12[a]. wherein the controller is configured to send the content status such that the network entity further determines if the source includes at least one available piece of content for the terminal to download,	*See* 6[a].
12[b]. wherein the controller is configured to receive a response that further indicates if the source includes at least one available piece of content, and	*See* 6[b].
12[c]. wherein the controller is further configured to download the at least one available	*See* 6[b].

’559	Kloba (EX1005) + Robbin (EX1006)
piece of content when the network entity determines that the source includes at least one available piece of content.	
14[pre]. A method according to claim 13	*See* Ground 1, Claim 13.
14[a]. wherein sending a response comprises sending a response that instructs the terminal to at least one of delete at least one piece of content from the memory of the terminal, or	*See* 2[b].
14[b]. download at least one piece of content from a source of content.	*See* 2[c].
15[pre]. A method according to claim 14, and	*See* Claim 14.
15[a].	*See* 3[b].

’559	Kloba (EX1005) + Robbin (EX1006)
wherein sending a response comprises sending a response 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.	
16[pre]. A method according to claim 14,	*See* Claim 14.
16[a]. wherein sending a response comprises sending a response that instructs the terminal to download at least one piece of content from the source based upon server status information comprising a listing of at least one available	*See* 4[a], 4[b].

’559	Kloba (EX1005) + Robbin (EX1006)
piece of content from the source.	
17[pre]. A method according to claim 14 further comprising:	*See* Claim 14.
17[a]. determining if the memory of the terminal includes at least one piece of content to delete,	*See* 5[a].
17[b]. wherein sending a response comprises sending a response that instructs the terminal to delete at least one piece of content when the memory of the terminal is determined to include at least one piece of content to delete.	*See* 5[b].
18[pre]. A method according to claim 17 further comprising:	*See* Claim 17.

’559	Kloba (EX1005) + Robbin (EX1006)
18[a]. determining if the source includes at least one available piece of content for the terminal to download,	*See* 6[a].
18[b]. wherein sending a response comprises sending a response that further instructs the terminal to download at least one available piece of content when the source is determined to include at least one available piece of content.	*See* 6[b].
20[pre]. A computer-readable storage medium according to claim 19,	*See* Claim 19.
20[a]. wherein the second executable portion is configured to send a response that instructs the terminal to at least	*See* 2[b].

’559	Kloba (EX1005) + Robbin (EX1006)
one of delete at least one piece of content from the memory of the terminal, or	
20[b]. download at least one piece of content from a source of content.	*See* 2[c].
21[pre]. A computer-readable storage medium according to claim 20,	*See* Claim 20.
21[a]. wherein the second executable portion is configured to send a response 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* 3[a], 3[b].
22[pre].	*See* Claim 20.

’559	Kloba (EX1005) + Robbin (EX1006)
A computer-readable storage medium according to claim 20,	
22[a]. wherein the second executable portion is configured to send a response that instructs the terminal to download at least one piece of content from the source based upon server status information comprising a listing of at least one available piece of content from the source.	*See* 4[b].
23[pre]. A computer-readable storage medium according to claim 20 further comprising:	*See* Claim 20.
23[a]. a third executable portion configured to determine if the memory of the terminal includes	*See* 5[a].

’559	Kloba (EX1005) + Robbin (EX1006)
at least one piece of content to delete,	
23[b]. wherein the second executable portion is configured send a response that instructs the terminal to delete at least one piece of content when the second executable portion determines the memory of the terminal includes at least one piece of content to delete.	*See* 5[b].
24[pre]. A computer-readable storage medium according to claim 23,	*See* Claim 23.
24[a]. wherein the third executable portion is further configured to determine if the source includes at least one available piece of content	*See* 6[a].

’559	Kloba (EX1005) + Robbin (EX1006)
for the terminal to download, and	
24[b]. wherein the second executable portion is configured to send a response that further instructs the terminal to download at least one available piece of content when the second executable portion determines the source includes at least one available piece of content.	*See* 6[b].

VIII. NO BASIS FOR DISCRETIONARY DENIAL

A. *Fintiv*

Under the Director’s Interim Procedure for Discretionary Denials, the Board will not deny institution based on *Fintiv*[6] if there is compelling evidence of unpatentability, which this Petition provides, thereby concluding the *Fintiv* analysis. *Nokia of Am. Corp. v. TQ Delta, LLC*, IPR2022-00471, Paper 11 at 19-21 (Aug. 18, 2022). Regardless, the *Fintiv* factors also weigh against discretionary denial.

[6] *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11 (Mar. 20, 2020) (precedential).

1: The district court case, *VideoLabs, Inc. v. Roku, Inc.* (*see* §II.B above), is currently stayed until January 12, 2025 in light of pending proceedings on seven other patents at the PTAB (including IPRs on two patents that have not had decisions on institution; IPRs 2024-01023, -01024, 01025, and -01026) and at the Court of Appeals for the Federal Circuit. *See* EX1011, EX1012 §II.D, EX1013. Petitioner will also seek a stay pending resolution of this IPR.

2: D. Del.'s 33-month median time-to-trial statistic (EX1010, 14) would estimate trial for approximately July 2026, but with the case stayed, this is entirely uncertain. Trial could possibly take place in 2027, more than a year after a final written decision would be expected in this IPR.

3: To date, the court has not issued any substantive orders regarding the '559.

4: After the final written decision, the same grounds and arguments could not be presented in the litigation.

5: The parties are the same.

6: Petitioner is highly likely to prevail with respect to the Claims as shown herein.

B. §325(d)

1. '559 Prosecution

The Examiner during prosecution did not consider Kloba or Robbin or art with substantially the same disclosures (or the same or substantially the same arguments)

as those herein. To the extent the Examiner considered references that purportedly teach controlling content flow to a terminal (*see* §IV.B above), the Examiner erred in failing to reject the Claims over a combination of any of those references and art teaching content distribution systems.

2. Netflix IPR

The other IPR of the '559 (hereinafter the "Netflix IPR"), based on a petition filed by Netflix, Inc. on 2023-02-22 (s*ee* EX1008 (Petition), EX1009 (Institution Decision)) reached a final written decision on 2024-10-02 (*see* EX1013 (Final Written Decision)). The grounds of the Netflix IPR are based on the "Cassin" (U.S. Pub. No. 2003/0023427) and "Huston" (U.S. Patent No. 7,243,136) references. Cassin and Huston are unrelated to the Kloba and Robbin references, applied herein. Further, Cassin and Huston disclose or render obvious the '559 claims in different ways than the references applied herein, and thus the arguments in the Netflix IPR applying Cassin and Huston are different from those in the present IPR. For example, Cassini discloses a server providing a set of content items to a client one at a time, and waiting for a response from the client (*e.g.*, indicating that the client already has the provided content item) before providing the next content item in the set (*see* EX1008, pp. 10-12 (Cassin overview)), while Kloba discloses a server providing instructions to a client to synchronize a set of multiple content items, rather than

providing instructions for each content item separately (*see, e.g.,* Kloba, 19:64-20:3). McNair, [42].

Moreover, Netflix is a different party from Petitioner here, and no significant relationship exists between the parties. Petitioner and Netflix are accused of infringing the '559 patent with different products in different court proceedings (*see* §II.B) and have not coordinated regarding the '559 patent, the separately accused products, or the filing of petitions for IPR. Petitioner and Netflix may have further independent interests in pursuing IPR of the '559. Unique claim construction and infringement theories may arise from these different positions. There is no basis for denial under §325(d). *See Ford Motor Company v. Neo Wireless LLC*, IPR2023-00763, Paper 28 (Mar. 22, 2024) and *American Honda Motor Co., Inc. v. Neo Wireless LLC*, IPR2023-00797, Paper 27 (Mar. 22, 2024) (vacating denials of institution based on General Plastic[7] factors, finding no significant relationship between parties having different accused products that merely engage in court-ordered pretrial coordination).

Further, the final written decision in the Netflix IPR finding all claims of the '559 unpatentable (*see* EX1013 (Final Written Decision), 58) does not warrant the

[7] *General Plastic Industrial Co., Ltd. v. Canon Kabushiki Kaisha*, IPR2016-01357, Paper 19 at 16 (Sep. 6, 2017) (precedential as to §II.B.4.i).

exercise of §325(d) discretion in the present case. As the Director noted in reversing a panel's §325(d) denial in a similar circumstance in the *Aviagames* case, "the challenged claims have not yet been cancelled and remain in force until the opportunity to appeal has been exhausted," and "[b]y the time an appeal will have concluded, Petitioner will be barred under 35 U.S.C. § 315(b) from bringing a new challenge in an IPR petition." *Aviagames, Inc. v. Skillz Platform, Inc.*, IPR2022-00530, Paper 14 (Decision on Director Review), pp. 3-4 (P.T.A.B. March 2, 2023). In such circumstances, the Director instructed the Board to institute when the petition "presents a compelling, meritorious challenge," which the present Petition does as discussed above. *See id.*; §VIII.A.

Thus, discretionary denial under §325(d) is not warranted.

IX. CONCLUSION

The Board should institute IPR and cancel the Claims of the ’559.

Dated: November 11, 2024

Respectfully submitted,

By: /Scott A. McKeown/
Scott A. McKeown, Reg. No. 42,866
WOLF, GREENFIELD & SACKS, P.C.

Counsel for Petitioner
Roku, Inc.

CERTIFICATE OF SERVICE UNDER 37 C.F.R. § 42.6 (E)(4)

I certify that on November 11, 2024, I will cause a copy of the foregoing document, including any exhibits or appendices filed therewith, to be served via Overnight FedEx at the following correspondence address of record for the patent:

Workman Nydegger
60 East South Temple
Suite 1000
Salt Lake City, UT 84111

Courtesy copies of the same documents were also served at the following email addresses of record for Patent Owner's litigation counsel:

M. Elizabeth Day	eday@bdiplaw.com
Marc Belloli	mbelloli@bdiplaw.com
Jerry D. Tice II	jtice@bdiplaw.com
Aaron R. Hand	ahand@bidplaw.com
Hillary Bunsow	hbunsow@bdiplaw.com

Date: November 11, 2024

/MacAulay Rush/
MacAulay Rush
Paralegal
WOLF, GREENFIELD & SACKS, P.C.

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. §42.24, the undersigned certifies that the foregoing IPR petition contains 10,766 words excluding a table of contents, a table of authorities, Mandatory Notices under §42.8, a certificate of service or word count, or appendix of exhibits or claim listing. Petitioner has relied on the word count feature of the word processing system used to create this paper in making this certification.

Date: November 11, 2024

/MacAulay Rush/
MacAulay Rush
Paralegal
WOLF, GREENFIELD & SACKS, P.C.