

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

ROKU, INC.

Petitioner

v.

ANONYMOUS MEDIA RESEARCH HOLDINGS, LLC.

Patent Owner

Case IPR2024-01056
U.S. Patent 10,572,896

PATENT OWNER'S PRELIMINARY RESPONSE

TABLE OF CONTENTS

TABLE OF CONTENTS	2
EXHIBIT LIST	4
CONVENTIONS	5
I. Introduction.....	6
II. Level of Ordinary Skill In The Art.....	7
III. Claim Construction.....	9
A. content identification results corresponding to a sequence of submitted data samples.....	10
B. play lists and play streams.....	16
IV. Ground 1 Does Not Render Any Challenged Claims Unpatentable	22
1. Feininger does not teach or suggest content identification results corresponding to a sequence of submitted data samples or sample sequence data corresponding to a sequence of the content identification results – Claims [1.1], [1.3], [9.1], [9.3]	22
2. Feininger does not teach or suggest comparing a progression of log times associated with capture timing of each of the particular plurality of submitted data samples and a progression of content offsets of the corresponding content identification results – Claims [1.5], [9.6].....	25
3. Feininger does not teach or suggest utilizing sample sequence data together with corresponding channel data to deduce play- altering actions of a monitored audience member [Claim 3]	28
4. Feininger does not teach or suggest wherein a clean play list is generated including deduced play-altering actions [Claim 4].....	29
5. Feininger does not teach or suggest a raw play stream [Claim 5] and a clean play stream [Claim 6].....	31
V. Ground 2 Does Not Render any Challenged Claims Unpatentable	35
A. Petitioner fails to show the asserted combination would have been obvious.	35
B. Ramaswamy-Conklin Does Not Disclose All Limitations of the Challenged Claims	41

1.	Ramaswamy-Conklin does not teach or suggest video data samples or submitted data samples – Claims [1.1], [9.1].....	41
2.	Ramaswamy-Conklin does not teach or suggest utiliz[ing] sample sequence data and content offset data to deduce play-altering actions of the monitored audience member – Claims [1.2], [9.2].....	42
3.	Ramaswamy-Conklin does not teach or suggest content offsets ... indicating a time position of a content identification result within identified content – Claims [1.4], [9.5]	49
4.	Ramaswamy-Conklin does not teach or suggest comparing a progression of log times associated with capture timing of each of the particular plurality of submitted data samples and a progression of content offsets of the corresponding content identification results – Claims [1.5], [9.6].....	49
5.	Ramaswamy-Conklin does not teach or suggest generating a play list including content offset data as part of deducing media play-altering actions of a monitored audience member [Claim 2]	58
6.	Ramaswamy-Conklin does not teach or suggest utilizing sample sequence data together with corresponding channel data to deduce play-altering actions of a monitored audience member [Claim 3].....	59
7.	Ramaswamy-Conklin does not teach or suggest wherein a clean play list is generated including deduced play-altering actions [Claim 4].....	61
8.	Ramaswamy-Conklin does not teach or suggest determining if one or more discontinuities exist in the progression of content offsets - Claim 7	64
9.	Ramaswamy-Conklin does not teach or suggest Claims 12 or 13 because it does not disclose including content offset data in one or more of a play list or play stream.....	65
VI.	Ground 3 Does Not Render Any Challenged Claim Unpatentable.....	65
A.	Petitioner fails to identify any reason to “improve” Feininger with Conklin.....	65
B.	Ground 3 does not disclose the limitations of Claims 2, 7, 12, or 13.	66
VII.	Conclusion	66

EXHIBIT LIST

Exhibit No.	Description
2001	Expert Declaration of W. Leo Hoarty
2002	<i>Curriculum Vitae</i> of W. Leo Hoarty
2003	U.S. Patent No. 6,990,453 B2
2004	U.S. Patent Application Publication No. US 2002/0083060 A1
2005	U.S. Patent No. 7,469,283
2006	U.S. Patent No. 8,972,481
2007	Defendant Roku, Inc.'s Identification of Proposed Terms for Construction, <i>Anonymous Media Research Holdings, LLC v. Roku, Inc.</i> , Case No. 1:23-cv-01143-DII (W.D. Tex., Austin Division, July 3, 2024)

CONVENTIONS

1. Patent Owner is referenced herein as “AMR”.
2. All bold emphasis is added unless otherwise noted.
3. All emphasis in italics is in original unless otherwise noted.

I. Introduction

Ground 1 (Feininger) does not teach or suggest *generating a play stream of content identification data corresponding to a sequence of submitted data samples*, required by the independent claims, because it relies on embedded tags or codes that reside in the media data. Therefore, it never submits any data samples for corresponding content identification. It also fails to teach or suggest *log times associated with capture timing* of the data samples, because Feininger's log times are not associated with the time the data samples are captured. Feininger also does not teach or suggest the limitations of multiple dependent claims.

Ground 2 (Ramaswamy-Conklin) does not teach or suggest multiple elements of the independent claims, including the use of *content offsets*. While Petitioner asserts that the calculation of content offsets was obvious, the Petition provides no rationale for **why** a POSA would undertake such calculations in the combination. Indeed, neither system relies on such offsets (and even their calculation was impossible in Conklin's system). Ramaswamy-Conklin also fails to teach or suggest the limitations of multiple dependent claims. Moreover, Conklin teaches away from combination with systems relying on program signatures like Ramaswamy.

Ground 3 (Feininger-Conklin) relies solely on Conklin's disclosure as teaching or suggesting the elements of dependent claims 2, 7, 12, and 13, which each require *content offsets* that Conklin does not use (and which cannot be determined in Conklin's

system). Petitioner also fails to identify any valid reason why a POSA would have combined Conklin with Feininger.

Accordingly, the Petition fails to show that any challenged claims are unpatentable.¹

II. Level of Ordinary Skill In The Art

Petitioner's proposed definition is unduly narrow because it would preclude a person having considerable industry experience designing and implementing automated content recognition ("ACR") systems, but with no formal degree, from qualifying as a person having ordinary skill in the art ("POSA"). Ex. 2001 ¶ 45. Patent Owner's technical expert Mr. Hoarty has two years of college education in the field of electrical engineering, several decades of industry experience designing and implementing ACR systems, and numerous patents to his name covering ACR technologies. Ex. 2001 ¶ 47. He qualifies as a POSA under any reasonable proposed definition based on his industry experience, notwithstanding his lacking a formal degree.

¹ As none of Petitioner's grounds, as asserted, render any challenged claim unpatentable, Patent Owner ("AMR") does not address the priority date of the '896 Patent. *See* Petition, 15-17. AMR reserves the right to prove priority to the provisional application should it become necessary.

Petitioner's proposal is also overly broad in that it only requires industry experience in "digital media content delivery," which relates broadly to *delivering* media content rather than *recognizing* or *identifying* media content. The level of ordinary skill in the art should require industry experience designing or implementing ACR systems, which is a fundamental aspect of the claimed invention and the technology described in the '896 Patent. Indeed, it is the *recognition* or *identification* of delivered media content that is key, not simply delivering media content. Mr. Hoarty has decades of experience with such systems, and thus also meets this narrower requirement. Ex. 2001 ¶¶ 45-47.

Further, while Petitioner's expert Dr. Russ may have expertise in various electrical fields, Petitioner provides no evidence that he has any industry experience with media content identification or recognition. Dr. Russ discusses his experience: (1) managing a cable set-top box design group; (2) acting as "a staff expert in home networking;" (3) managing a hardware design team that completed the design of a payment terminal; (4) teaching circuit board design and two-way interactive video classes; and (5) authoring articles and conference papers "on digital video recording." EX1003 ¶¶ 10-14. It does not appear that any of this experience relates to or involves the recognition or identification of media content. Indeed, it appears that Petitioner broadly proposes "digital media content **delivery**" as the relevant industry experience because its proffered expert has no experience in the relevant field of media content

recognition. AMR reserves the right to challenge Dr. Russ’s qualifications as a POSA should the Board institute trial.

III. Claim Construction

Petitioner bears the burden of identifying “[h]ow the challenged claim is to be construed.” 37 C.F.R. § 42.104(b)(3). Failure to construe a claim term in controversy, or providing an incorrect claim construction, is grounds for denial. *Jiawei Tech. (HK) Ltd. v. Richmond*, IPR2014-00937, Paper 22 at 7-8 (PTAB Dec. 16, 2014) (noting that “[i]t is Petitioner’s burden to explain how the challenged claims are to be construed and how they read on the prior art” and finding that petitioner failed to meet that burden for an unconstrued claim term); *United Microelectronics Corp. v. Lone Star Silicon Innovations LLC*, Case IPR2017-01513, Paper 8, 16-18 (PTAB Jan. 31, 2018) (reiterating the Board may decline to institute when patentability challenges depend on an incorrect claim construction). In related litigation, Petitioner identified multiple terms for construction: “video data samples,” “data samples,” “raw play stream,” “play stream,” and “monitoring device.” Ex. 2007. Despite this, Petitioner identifies no terms requiring construction here, asserting that it merely gives all terms their “plain and ordinary meaning.” Petition, 18.

AMR identifies certain terms below as requiring construction given Petitioner’s application of the asserted grounds.

A. content identification results corresponding to a sequence of submitted data samples

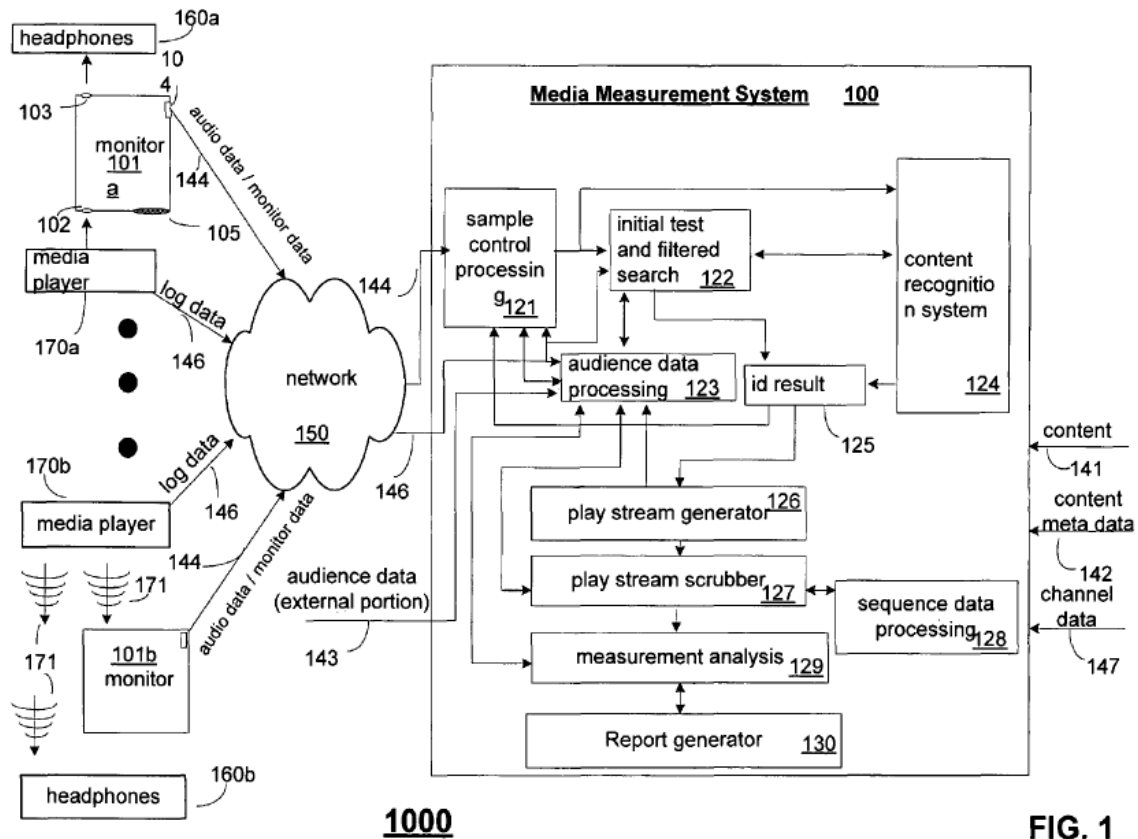
Claims 1 and 9 require generating “a play stream of content identification results corresponding to a sequence of submitted data samples,” with “sample sequence data corresponding to a sequence of the content identification results,” and “submitted data samples and corresponding content identification results.” EX1001, 15:12-35, 16:10-36. The submitted data samples correspond to “video data samples captured at a monitoring device associated with a monitored audience member.” *Id.* Claim 8, depending from claim 1, requires the “content offset data [be] derived from a comparison ... comparing the submitted data samples against known content such that content offset data is obtainable without using time codes accompanying the captured video data samples.” *Id.*, 16:4-9.

The specification explains that prior art systems which focused on monitoring channels did not allow for tracking consumption of particular media content, and that prior art systems using codes to tag or track particular content could not identify content that had not been encoded. EX1001, 1:18-38. Thus, the specification recites a system and method “that enhances recognition ... of the content of a media sample” via various means, as well as embodiments that deduce play-altering activities “from content offset values of identifications corresponding to captured samples.” *Id.*, 1:64-2:22.

An exemplary system practicing the claimed methods is illustrated in Fig. 1,

reproduced below.² The specification describes capturing “audio data” (which alternatively comprises video data as referenced above) at monitoring devices (101) and then uploading that data to the system (100). *Id.*, 3:45-62. The data is described as alternatively “raw” data or “data that has undergone one of a variety of levels of processing prior to upload.” *Id.* However, this data is not itself described as a “data sample.” *See id.*

² Although the specification generally describes embodiments in the context of media samples comprising audio data, it makes clear that these disclosures apply equally to alternative embodiments that include video data. EX1001, 14:17-29 (“In alternative embodiments to what is illustrated and described in FIGS. 1-6, a record is created of individual media usage over time by means of an open-ended measurement system that employs portable and stationary detection devices to generate digital fingerprints of the video signal portion of the media content itself and identifies media played in the environment by means of matching such fingerprints against a database of known media. ... Thus, various alternatives to the exemplary methods discussed in the context of the audio data-based embodiments illustrated in FIGS. 1-6 rely instead on video data.”)



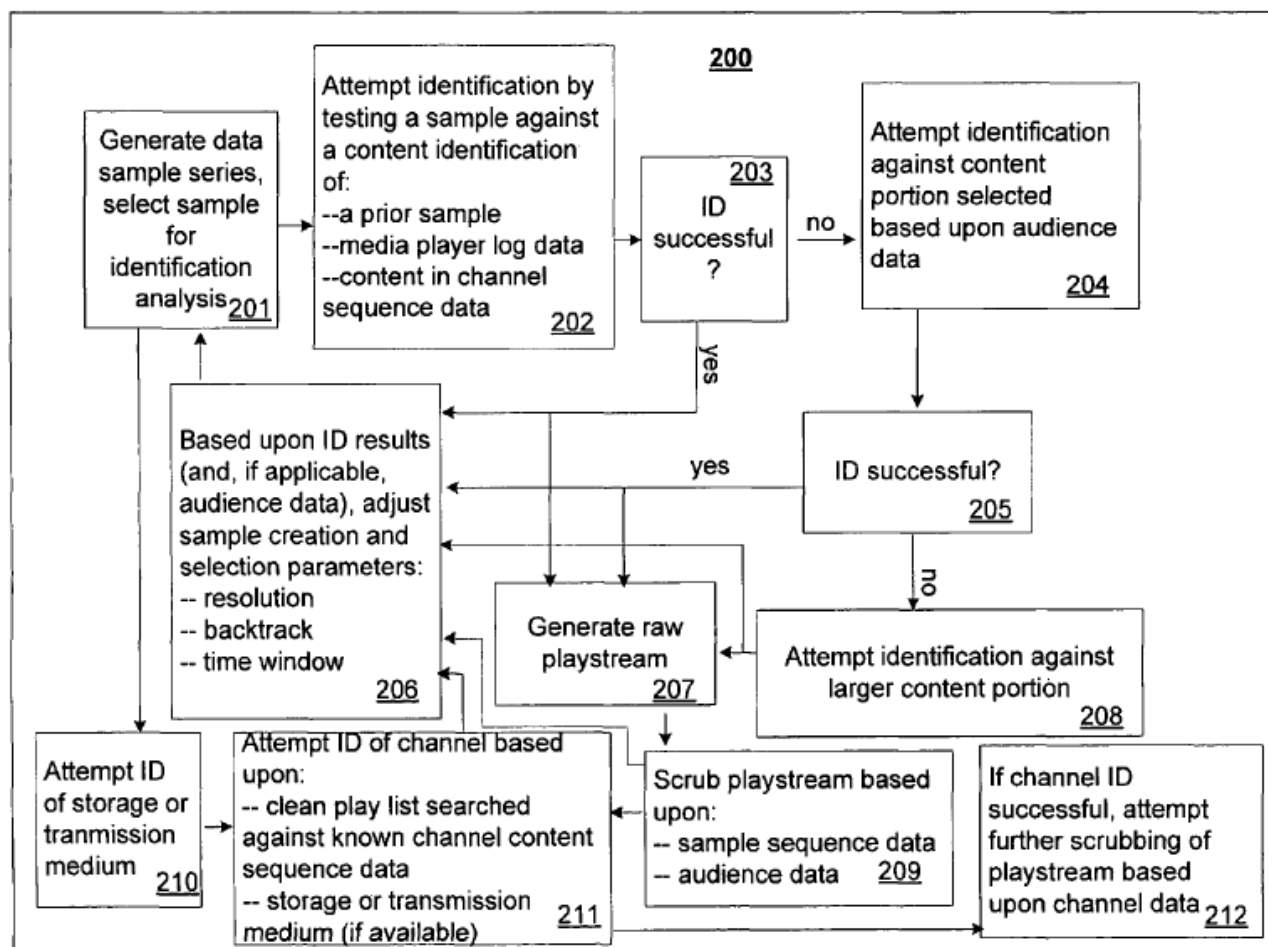
The system contains a “sample control processing module” (121) that first receives the audio data and which then **“divides the audio data into samples and determines which samples should be submitted for further content recognition testing.”** *Id.*, 4:58-5:5. Alternative embodiments are described wherein the “audio data may already have been **divided into samples** prior to receipt” by the system. *Id.* Thus, the captured data is always described as being divided into samples. *Id.*

This module submits such samples to either an “initial test and filtered search module” (122) or a “content recognition system” (124). *Id.* The test/search module “conducts one or more initial identification attempts.” *Id.*, 5:6-38. “After completing the initial identification attempt,” the module “passes the identification result” to an “ID

result” module (125). *Id.*, 5:39-6:23. If there was a successful identification, the result is passed to a play stream generator (126); if no identification was made, the content recognition system makes another attempt by performing “a search using a larger portion of its content to identify the data sample.” *Id.* The content recognition system uses known content recognition methods that “identify content by extracting parameters from a media signal and applying an algorithm to those parameters to search for a content match.” *Id.*

This search result, whether successful in identifying the content or not, is passed to the ID result module, which passes the result to the play stream generator and to the “sample control module.” *Id.*, 5:39-6:23, 7:26-48. The play stream generator generates “a raw play stream relating a sequence of samples to content identification results.” *Id.*, 6:24-37.

A disclosed method is illustrated in Figure 2, reproduced below for reference. The method recites as a first step (201) the receipt of “audio data from a monitoring device.” *Id.*, 7:1-13. This step (201) is labeled in Figure 2 as “Generate data sample series, select sample for identification analysis” and caches a series of samples. *Id.*, Fig. 2, 7:1-13.



The method (step 202) describes initial identification attempts, which if successful are passed on to subsequent steps described below. *Id.*, 7:26-48. If unsuccessful, additional identification attempts are made. *Id.* (step 204).

Successful initial identification attempts are used to generate a raw play stream “indicating a series of samples and corresponding identification results” (step 207). *Id.*, 8:13-34. Subsequent steps relate to scrubbing this raw play stream based on sample sequence and audience data, as well as further attempts to correctly identify the content

based on known channel data. *Id.*

The raw play stream itemizes a sequence of analyzed samples and relates that sequence to corresponding content identification results in a data structure (e.g., a table) containing the sample sequence data. *Id.*, 9:25-45, Fig. 4. A content identification result corresponds to a particular piece of content, or indicates that no identification was made. *Id.* The sample sequence data includes log times (the time at which capture of the analyzed sample began by a monitor), which may be relative to an arbitrary value (the beginning of the sequence) or represent absolute times (actual time the sample started to be captured). *Id.*, 9:25-45, 11:29-46.

From the foregoing, it is first apparent that the specification distinguishes between video data and “video data samples” and “submitted data samples.” Data samples are **generated from** video data. A POSA would understand from the plain language of the specification that “video data” and “data samples” are not equivalents, but that data samples are generated from the video data. Ex. 2001 ¶ 61.

Second, it is also clear that the specification exclusively uses “content identification results” to refer to the results of a content identification process, one having the express purpose of recognizing the content of the submitted data samples. Both the specification and the claim language repeatedly require that the content identification results correspond to the submitted data samples: “content identification results corresponding to a sequence of submitted data samples,” “the sample sequence

data corresponding to a sequence of the content identification results,” and “submitted data samples and corresponding content identification results.” EX1001, 15:12-35. This may be a successful result (the content is identified) or unsuccessful (no identification is made). But in all cases, “content identification results” in the specification entails more than merely using the disparaged prior art systems relying solely on embedded tags or encoding, which “are limited in that they can only track content that has been properly encoded.” EX1001, 1:39-43. Such systems do not require any process to identify content, or have any need to generate and submit data samples for content identification. In these systems, the content was identified when encoded or tagged, and that identification accompanies the data. Ex. 2001 ¶ 62.

Thus, a POSA would understand that *content identification results corresponding to a sequence of submitted data samples* refers to the results of an attempt to accurately identify the content of submitted data samples generated from video data. This requires first generating data samples, which are then submitted for content identification, yielding a resulting content identification corresponding to the submitted sample. *Id.* ¶ 63.

B. play lists and play streams

As discussed in the analysis of the asserted grounds (Sections IV.-VI.), in discussing these terms Petitioner repetitively cites the specification’s statement that “the particular organization, division, and content of the data may vary considerably from

that illustrated ... without departing from the spirit and scope of the present invention” (EX1001, 10:51-58). Petition, 36 n.1, 62, 66 (referencing *play stream*); *id.*, 51, 73, 75 (referencing *play list*). While this passage indicates that *play stream* and *play list* may vary in form and content, it does not mean that these terms have no meaning and can therefore be met by any data set whatsoever. While they need not conform to the disclosed embodiments, these different terms—while related—are not interchangeable. While these data sets may be represented in different ways and have varying additional content they are not the same and correspond to clearly different data sets in the claimed invention. EX1001, 10:51-11:7.

Claims 1 and 9 require generation of a “play stream of content identification results.” EX1001, 15:12-35, 16:10-36. Claim 2 requires “generating a play list including content offset data.” *Id.*, 15:36-39. Claim 4 specifies a “clean play list” “including deduced play-altering actions.” *Id.*, 15:44-46. Claims 5 and 6 alternatively require a “raw play stream” and a “clean play stream.” *Id.*, 15:47-50.

The specification explains that the disclosed methods conduct multiple identification attempts. *Id.*, 7:26-48. Both successful and (if present) unsuccessful results are ultimately provided for inclusion in the raw play stream. *Id.* (steps 207, 208). The “raw play stream” relates a sequence of samples to their corresponding content identification results. *Id.*, 6:24-37, 8:13-34. This raw play stream is then “scrubbed” to generate a “clean play stream” which is then converted to a “play list” listing the content

in the order captured. *Id.*, 8:13-34.

“Scrubbing” refers to the process of reviewing, and then correcting, the play stream or play list for missing or incorrect data by using additional data. *Id.*, 8:42-61; Ex. 2001 ¶¶ 67-69. The specification explains that contents of the play stream may be compared with known channel data to identify a channel. *Id.*, 8:62-9:16. A play stream or play list may be scrubbed with channel and/or non-channel data to generate a clean play list. *Id.*, 8:62-9:24. When deduced user actions are sought, these actions may be included on the clean play list. *Id.*, 9:17-24.

As described with reference to the embodiment of Figure 4, the scrubbing process for generating a clean play stream and clean play list begins with the raw play stream which lists the sequence of analyzed samples and relates them to their corresponding identification results (including whether no identification was made). *Id.*, 9:25-45. As described, this involves locating and correcting any likely incorrect identifications, or supplying missing identifications where they do not yet exist. *Id.*, 9:46-51. This is done by analyzing the pattern of identified samples, where unidentified samples may be reasonably assumed as containing the same content as preceding or succeeding samples. *Id.*, 9:52-65. Thus, the clean (“scrubbed”) play stream fills in this content identification for the identifications missing from the raw play stream. *Id.* Based on the sequence of content, a play list can be generated. *Id.*, 9:66-10:19. Channel and audience data may be used to correct unidentified and misidentified samples. *Id.*, 9:66-10:50.

The difference between a play stream and a play list is illustrated in Figures 4 and 5. While a play stream contains the content identification result and log (capture) times of each sample (with one or more samples for each specific item of media content), the play list condenses this information to correspond to the start/end time of each separate piece of media content (such as a particular song, ad, or television program). *Id.*, Fig. 4 (reproduced and highlighted below). Thus, as is consistent with its plain and ordinary meaning, a *play list* lists each item of media content once per event.

Ex. 2001 ¶¶ 71.

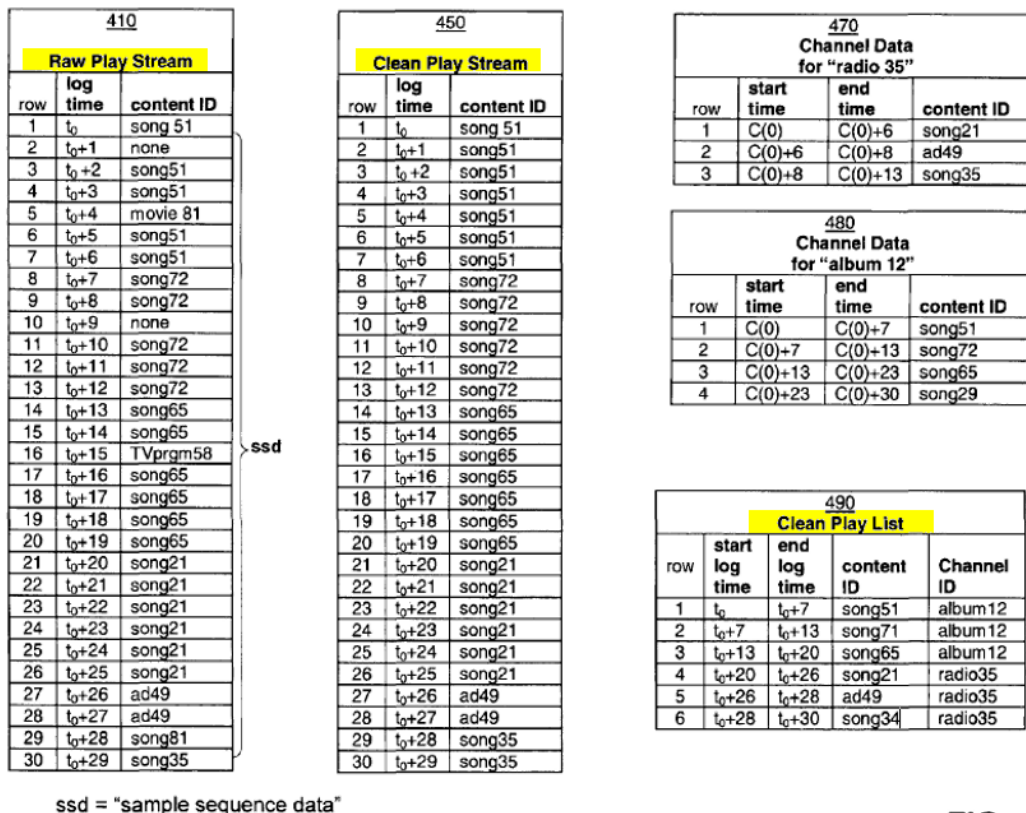


FIG. 4

Clean play streams and play lists may show deduced play-altering actions. *Id.*, 11:8-28. Such actions include pausing, reversing, fast-forwarding, and skipping

backwards or forwards. *Id.* The clean play stream contains log times (the time at which capture of the analyzed sample began), content identifiers, and content offset values for a series of samples. *Id.*, 11:29-46. The content offset value for each sample represents a time offset from the beginning of a particular piece of content. *Id.*

A play list may thus provide a start and end time for each “internally continuous media consumption event.” *Id.*, 11:47-58. That is, when an entire piece of media content is consumed without interruption, the play list notes it as a single entry. *Id.* If there is an interruption or “discontinuity” in consumption, it forms a boundary to the consumption event, and the play list will note the interruption in a separate entry. *Id.*, Fig. 5 (reproduced and highlighted below).

510			
Clean Playstream			
row	log time	content ID	content offset
1	t_0	TVprgm 27	(0)
2	t_0+1	TVprgm 27	(0)+1
3	t_0+2	none	n/a
4	t_0+3	none	n/a
5	t_0+4	none	n/a
6	t_0+5	TVprgm 27	(0)+2
7	t_0+6	TVprgm 27	(0)+3
8	t_0+7	TVprgm27	(0)+9
9	t_0+8	TVprgm27	(0)+10
10	t_0+9	TVprgm27	(0)+9
11	t_0+10	TVprgm27	(0)+10
12	t_0+11	none	n/a
13	t_0+12	TVprgm27	(0)+29
14	t_0+13	TVprgm27	(0)+30
15	t_0+14	ad23	(0)
16	t_0+15	none	n/a
17	t_0+16	ad25	(0)+2
18	t_0+17	TVprgm28	(0)
19	t_0+18	TVprgm28	(0)+1
20	t_0+19	TVprgm28	(0)+2

560				
Clean Play List				
row	start log time	end log time	content ID	content offset
1	t_0	t_0+2	TVprgm27	(0)
2	t_0+2	t_0+5	none	n/a
3	t_0+5	t_0+7	TVprgm27	(0)+2
4	t_0+7	t_0+9	TVprgm27	(0)+9
5	t_0+9	t_0+11	TVprgm27	(0)+9
6	t_0+11	t_0+12	none	n/a
7	t_0+12	t_0+14	TVprgm27	(0)+29
8	t_0+14	t_0+15	ad23	(0)
9	t_0+15	t_0+16	none	n/a
10	t_0+16	t_0+17	ad25	(0)+2
11	t_0+17	t_0+20	TVprgm28	(0)

570			
Channel Data for TV C4			
row	start time	end time	content ID
1	C(0)	C(0)+31	TVprgm27
2	C(0)+31	C(0)+34	ad23
3	C(0)+34	C(0)+37	ad24
4	C(0)+37	C(0)+40	ad25
5	C(0)+40	C(0)+50	TVprgm28

590				
Clean Play List with Deduced Actions				
row	log time	end log time	content ID and <deduced action>	content offset
1	t_0	t_0+2	TVprgm27	(0)
2	t_0+2	t_0+5	<pause for 3 units>	n/a
3	t_0+5	t_0+7	TVprgm27	(0)+2
4	t_0+7	t_0+7	<skip frwrd 5 units>	n/a
5	t_0+7	t_0+9	TVprgm27	(0)+9
6	t_0+9	t_0+9	<skip bckwrd 2 units>	n/a
7	t_0+9	t_0+11	TVprgm27	(0)+9
8	t_0+11	t_0+12	<fast frwrd 18 units>	n/a
9	t_0+12	t_0+14	TVprgm27	(0)+29
10	t_0+14	t_0+15	ad23	(0)
11	t_0+15	t_0+16	<fast frwrd 7 units>	n/a
12	t_0+16	t_0+17	ad25	(0)+2
13	t_0+17	t_0+20	TVprgm28	(0)

FIG. 5

These discontinuities are identified by reference to the progression of sample log times and the corresponding progression of content offsets in the play stream. *Id.*, 11:59-12:10. An equivalent progression of log times and content offsets signifies continuous consumption, while a difference in the progression of log times and content offsets shows a discontinuity. *Id.* Based on this information, a clean play list may be constructed with deduced play-altering actions. *Id.*, 12:17-22. Thus, when log time values in the clean play stream reflect advancing time units with no corresponding content ID, and the preceding and succeeding content offset values in the clean play stream indicate advancing no more than a single time unit within the same content ID,

it can be deduced that the program was paused for those time units. *Id.*, 12:23-35. Similar analyses are used to deduce fast-forwarding, rewinding, and skipping, which may also use channel data and gaps in content offset times with no corresponding gap in log times. *Id.*, 13:7-31.

Thus, while *play stream* and *play list* are not required to conform exactly to the disclosed embodiments, it is apparent they have different meanings and contents. Ex. 2001 ¶ 75. A *play stream* contains content identifications and log (capture) times of each submitted data sample (where each specific piece of media content may be represented by multiple samples). *Id.* A *play list* is generated from a *play stream*, and condenses the information therein to correspond to the log times of each continuous media consumption event, for each separate piece of media content. *Id.*

IV. Ground 1 Does Not Render Any Challenged Claims Unpatentable

1. Feininger does not teach or suggest *content identification results corresponding to a sequence of submitted data samples or sample sequence data corresponding to a sequence of the content identification results* – Claims [1.1], [1.3], [9.1], [9.3]

Petitioner asserts that Feininger's presentation time retriever *generates a play stream of content identification results* when it "periodically creates a running list of log entry data structures 58" and populates the data structures with a program identifier retrieved from the packet data in a datastream of a played-back program. EX1005, 4:61-64, 5:28-42, 16:56-64; Petition, 36. Petitioner explains that Feininger's DVR receives broadcast program data which is "packetized" into datastreams of audio and video

packets. Petition, 36-37. A user's playback of this stored program data results in the creation of the log entry data structures created in response to the playback. *Id.* Petitioner thus asserts that the "user's viewing activity corresponds to the claimed *sequence of submitted data samples.*" *Id.*, 37.

Feininger's log entry data structures optionally include, for each section of the recorded program, the playback date and time, recording date and time, and program identifier. EX1005, 5:3-42. A program identifier included in the data structures definitively identifies the program being played back. EX1005, 5:32-37 ("The program identifier may be any form of information that identifies the program being played back"). There is no indication in Feininger that the program identifier is generated by any portion of Feininger's system, but it is merely received and then extracted from the "packet data" of the program as it is played. *Id.*; Petition, 36-37.

Thus, while Feininger does generate data structure entries populated with a content identifier, this is not the same as *generat[ing] a play stream of content identification results corresponding to a sequence of submitted data samples*. As explained above, the claim requires data samples submitted for the purpose of content identification (which is done using various described processes). Section III.A. But Feininger never submits any samples for content identification. Rather, when program data is first received, Feininger's recording device "packetizes **the received program data** into one or more datastreams," including writing "other data" such as the program

identifier into the datastreams. EX1005, 16:1-21 (referencing Fig. 9 block 206). This identifier is a code or tag received along with the program data; there is no indication that Feininger performs any content identification process to somehow derive it. Ex. 2001 ¶¶ 83-84. Feininger's system merely reads the identifier embedded in the datastream, and writes it in the log entry data structures. EX1005, 16:39-17:2 (referencing Fig. 10 blocks 228, 230). Thus, Feininger never submits any data samples to obtain content identification results for those samples, because its system receives their content identification **before any samples are even created** and, just as in the prior art, simply reads a program identifier encoded in the datastream. EX1001, 1:39-43; Ex. 2001 ¶¶ 84-85. Like the prior art, these identifications are limited in that they cannot track content unless it is properly encoded. They are not *content identification results* and cannot *correspond to a sequence of submitted data samples* because they are not results of any content identification process, and because no data samples are ever submitted for content identification. Instead, there are merely "various sections of a received broadcast program" that require no identification. EX1005, 4:20-22; Section III.A.; Ex. 2001 ¶ 86.

Further, because Feininger's system does not submit any data samples to generate content identification results, its log entry data structures do not contain any *content identification results* from any *submitted data samples*. Therefore, Feininger's log entry data structures do not contain *sample sequence data corresponding to a*

*sequence of the content identification results.*³

2. Feininger does not teach or suggest *comparing a progression of log times associated with capture timing of each of the particular plurality of submitted data samples and a progression of content offsets of the corresponding content identification results* – Claims [1.5], [9.6]

Petitioner asserts that Feininger’s playback times are *log times*, and maps the “creation time” of its log entry data structures to *the capture timing of each of the ... submitted data samples* because the data structures “are created at substantially the same time” as playback. Petition, 44-45. Petitioner further asserts that the playback/log time is *associated with the capture timing* because the playback times are recorded in the data structures. *Id.*

Petitioner misidentifies the claimed *log times associated with capture timing of each of the ... submitted data samples* in the ’896 Patent. The claims specify that the data samples are *captured at a monitoring device associated with a monitored audience member*. EX1001, 15:14-17, 16:14-16. The specification explains that the system captures data via monitors which can “capture[] audio energy through [a] media player

³ Feininger’s failure to disclose *content identification results* also precludes it from disclosing or suggesting elements [1.4] and [9.5] (requiring *content offsets associated with the content identification results*) and [1.5] and [9.6] (requiring *comparing a progression of content offsets of the corresponding content identification results*).

port.” *Id.*, 3:10-32. Alternatively, a monitor may capture samples via wireless transmission or “direct electrical connections and optical sensors.” *Id.*, 3:33-44, 14:24-26. However captured, the captured data from the monitoring device is uploaded to the system where it is then divided into samples, some subset of which are then selected for submission for content recognition. *Id.*, 3:45-62, 4:58-5:5, 7:1-13. The submitted data samples are consistently referred to as “captured samples.” *Id.*, 5:3-38, 8:13-34, 11:29-46. These samples are captured at the time an audience member plays the media content. *Id.*, 3:10-62.

The specification also explains that the sample log time recorded in the play stream(s) may be relative to an arbitrary value or reflect an actual time, but in each case “indicates a time at which capture of the analyzed sample began.” *Id.*, 11:29-46; *see also id.*, 9:25-45. Discontinuities in consumption (playback) are identified by comparing the progression of sample log times to the progression of corresponding content offset times. *Id.*, 11:59-12:10.

Thus, in the specification, *log time* always refers to the capture timing of the analyzed sample—the time when the sample’s data is first captured by the system. This is different from the times that the captured data is uploaded to the system from the monitors, divided into samples, or added to a play stream or play list. Thus, *log times associated with capture timing* of the data samples refers to the time the data was captured by the system—the time the content was consumed or played (or played back).

In Feininger, only the system's recording device captures the media data, which may be contained in a series of data packets. EX1005, 4:19-31. The recording times (also called the presentation times)—the “times at which various sections of a received broadcast program are received and/or stored”—are recorded by a time stamper, which may add the time to the data packet headers in the audio and/or video datastreams. *Id.*, 4:15-48. Thus, Feininger's system captures the data when it is first received and stored by the recording device.

When a previously recorded program is played back, the presentation time retriever “retrieves the recording/presentation times of some or all of the sections of the program being played back and stores those recording times in association with the playback time of the corresponding sections of the program.” *Id.*, 4:49-61. This is done via the log entry data structures. *Id.*, 4:61-5:2 (referencing Figs. 3A-3C). Although each data structure is “preferably created at substantially the time that a corresponding section of the recorded program is being played,” (*id.*), this is not the time at which the data or sample was captured. The presentation time retriever merely retrieves the recording/presentation time at playback and correlates it with each segment's playback time. But the data capture occurred earlier, at the recording/presentation time. Ex. 2001 ¶ 94.

This distinction highlights a fundamental difference between Feininger and the claimed invention. In the specification and claims, access to the captured samples at the

time of recording/presentation is not required as it is in Feininger. Rather, the capture occurs once, at the time of consumption (playback), and does not reflect the recording/presentation time. This is the *log time associated with capture timing* of each sample. In Feininger, the data is captured and, critically, time-stamped when received by the system (recording/presentation time). EX1005, 16:1-13. Feininger’s playback time (mapped as the *log time* by Petitioner)—is thus not *associated with the capture timing* of the data samples, because Feininger does not capture the data at the time of playback. Certainly, there is no data or sample capture at the time Feininger creates its log entry data structures, which Petitioner identifies as the *capture timing*. Petition, 45; Ex. 2001 ¶¶ 95-96. Feininger therefore does not teach or suggest *comparing a progression of log times associated with capture timing of each of the ... submitted data samples*.

3. Feininger does not teach or suggest *utilizing sample sequence data together with corresponding channel data to deduce play-altering actions of a monitored audience member* [Claim 3]

Petitioner explains that Feininger may include channel data in the data log entry for a program section along with the program identifier. Petition, 50. Petitioner asserts that a channel number can be used to identify a program schedule “to discern additional information about the broadcast programming on that channel,” and that this schedule is used by Feininger’s non-real time content identifier to determine the particular segments of a program that were affected by a play-altering action. *Id.*, 50-51. But this

is not the same as utilizing the channel data in the first instance to *deduce play-altering actions* of an audience member, as required by Claim 3.

In Feininger, the non-real time content identifier is used to “identify the segment(s) in the portion of the recorded program that were not played back in real time.” EX1005, 9:61-63. The non-real time content identifier “**responds** to detection of non-zero differences by the difference detector” “by searching a database to determine the identities of the program segments that were skipped, fast forwarded, and/or rewound.” *Id.*, 9:63-10:1. That is, **before** the non-real time content detector is activated, Feininger’s system has already determined that “portion[s] of the recorded program ... were not played back in real time”—meaning that Feininger’s system has **already** “deduced play-altering activities of the audience member” in playing back the recorded program. *Id.*, 9:25-60, 9:61-62. Feininger thus only uses any “channel data” to further identify the segments that were subject to the already deduced play-altering actions, not to deduce the play-altering actions themselves. Ex. 2001 ¶¶ 100-101. Feininger thus does not teach or suggest Claim 3.

4. Feininger does not teach or suggest *wherein a clean play list is generated including deduced play-altering actions* [Claim 4]

Petitioner identifies Feininger’s recapture table as a “clean play list” that includes deduced play-altering actions. Petition, 52. Feininger teaches that the recapture table stores the identities of segments identified as having been skipped or fast forwarded, and the identities of all segments of the recorded program that were rewound. EX1005,

12:40-48, Fig. 6.

Feininger's recapture table is not a *clean play list* as that term is used in the specification. While the specification does not limit the term to the preferred embodiments, it has a clear meaning. *See* Section III.B. A "play list" includes a list of each distinct piece of media content in the sequence in which it was played. *Id.* A "clean play list" refers to a play list that has been scrubbed. *Id.* Scrubbing either a play stream or a play list involves correcting a raw stream or list by analyzing the sequence contents and correcting any mistaken or missing identifications of the listed samples or content. *Id.*

Feininger's recapture table is, first, not a *play list* at all because it does not provide a sequence of the particular pieces of media content that have been consumed. Rather, it merely lists those program segments that have been subjected to play-altering. Ex. 2001 ¶¶ 104-105. This is not a "play list," but more accurately a "not played list." *Id.*

The recapture table is also not a *clean play list* because Feininger does not teach or suggest scrubbing or cleaning the data captured in any of its data structures. Petitioner does not identify any aspect of Feininger which corrects or adds missing or incorrect data. As discussed for Claim 6 below, this is because a POSA would have understood that Feininger would not have missing or incorrect data in its various listings of content segments. Ex. 2001 ¶ 106; Section IV.5. Because Feininger relies exclusively on pre-

embedded codes or tags in its datastreams that definitively identify the program being played back (*see* Section IV.1), every entry in Feininger’s log entry data structure (which is used as the sole basis for deducing play-altering activities) is already conclusively identified before inclusion in the data structure. Ex. 2001 ¶ 106. Thus, a POSA would have understood that there would be no unidentified or misidentified entries in Feininger’s data structures. *Id.* Likewise, Feininger’s recapture table is never scrubbed to remove missing or incorrect data. *Id.* ⁴

5. Feininger does not teach or suggest *a raw play stream* [Claim 5] and *a clean play stream* [Claim 6].

Petitioner identifies Feininger’s log entry data structures 58 as both a raw play stream and a clean play stream. Petition, 53-54. Petitioner first construes *raw play stream* as a play stream that is “not scrubbed.” *Id.*, 53. Petitioner admits that “Feininger

⁴ Although Petitioner does not assert that Feininger renders claim 4 obvious (Petition, 51-53), Dr. Russ testifies that “Feininger teaches, or at least renders obvious, this element,” with no further explanation. EX1003 ¶ 193 (uncited in Petition). Even if considered, this conclusory assertion cannot support any conclusion of obviousness, as it provides no “articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.” *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 418 (2007) (quoting *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006)).

does not describe that the running log entry data structures 58 are scrubbed,” and on that basis asserts that they comprise a raw play stream. *Id.*

Next, Petitioner construes “clean play stream.” Here, Petitioner does not define the term as it did with “raw play stream”—in relation to whether or not the play stream was scrubbed—but as merely “a play stream that contains no missing or incorrect data.” *Id.*, 53-54. Asserting that the sequence data in Feininger “ensures that the log entry data structures 58, 70 are transmitted, analyzed, and/or processed in the correct order” and therefore prevents such missing or incorrect data, Petitioner proclaims that the data structures are both a raw play stream and a clean play stream at the same time. *Id.*, 54.

Petitioner tries to have it both ways by changing the basis of its construction of “raw” and “clean” play streams from whether the stream has been scrubbed to whether it contains missing or incorrect data. But consistency requires that if a raw play stream is not scrubbed, a clean play stream is one that has been scrubbed. Conversely, if a clean play stream is one containing no missing or incorrect data, a raw play stream is one that contains some missing or incorrect data. Either the log entry data structures are a raw play stream, or a clean play stream (or neither), but they cannot be both at the same time.

Apparently understanding that a clean play stream does require scrubbing data, Petitioner also advances an additional obviousness attack, asserting that both “scrubbing (or cleaning) data and the benefits thereof” was within the general

knowledge of a POSA and that it was therefore obvious to scrub Feininger's data structures "in order to ensure accurate results." Petition, 55. But Petitioner provides no reason **why** a POSA would add such scrubbing to "ensure accurate results" in Feininger.

Petitioner admits that "a POSA would have understood that the safeguards provided by utilizing sequence number field 72 would result in a running list of log entry data structures 70 **that do not include any missing entries.**" *Id.*, 54. This is correct—as discussed above, Feininger's datastreams definitively identify the media content therein, and therefore a POSA understood that the log entry data structures would not contain any missing or incorrect entries. *See* Section IV.1.; Ex. 2001 ¶ 115. A POSA had no reason to scrub Feininger's data structures "to ensure accurate results" because a POSA would have understood that there would be no missing or incorrect data to correct. Ex. 2001 ¶ 115.

Like the Petition, Dr. Russ's declaration contains no reasoning explaining **why** a POSA would add scrubbing to Feininger. Dr. Russ recites multiple references in an attempt to show that the general "concept" of data scrubbing was "notoriously well known." EX1003 ¶¶ 207-215. Even assuming this were true, nowhere to be found is any reason to apply data scrubbing to Feininger. Nor can there be any—even Petitioner acknowledges that Feininger's data structures will contain no missing entries, and its definitive identification of the content in the datastreams ensures there will be no incorrect data. Ex. 2001 ¶ 115. Petitioner's conclusory obviousness arguments amount

to a mere claim that a POSA **could have** scrubbed Feininger’s data, not that a POSA **would have** been motivated to do so. This is insufficient. “[O]bviousness concerns whether a skilled artisan not only *could have made* but *would have been motivated to make* the combinations or modifications of prior art to arrive at the claimed invention.” *Belden Inc. v. Berk-Tek LLC*, 805 F.3d 1064, 1073 (Fed. Cir. 2015); *see also Virtek Vision Int’l ULC v. Assembly Guidance Sys.*, 97 F.4th 882, 888 (Fed. Cir. 2024) (reversing obviousness finding where the prior art disclosed the claimed limitations, explaining that “[i]t does not suffice to simply be known. A reason for combining must exist”); *PersonalWeb Techs., LLC v. Apple, Inc.*, 848 F.3d 987, 993-94 (Fed. Cir. 2017) (“But [the PTAB’s] reasoning seems to say no more than that a skilled artisan, once presented with the two references, would have understood that they *could be* combined. And that is not enough: it does not imply a motivation to pick out those two references and combine them to arrive at the claimed invention.”).

The ’996 Patent makes clear that a “clean” play stream or list is one that has been scrubbed, entailing review and correction of any missing or incorrect data. *See* Section III.B.; Ex. 2001 ¶ 114. As Petitioner admits, nothing suggests that the log entry data structures are ever reviewed or corrected in any way, and thus they cannot be a clean play stream as that term is used in the patent. Feininger therefore does not teach or suggest a *clean play stream* (Claim 6). Ex. 2001 ¶¶ 114-115. Alternatively, if Petitioner’s alternative construction of “clean play stream” as merely one that contains

no missing or incorrect data is adopted, then Feininger’s play stream is a *clean play stream* but not a *raw play stream*. In this case, Feininger does not teach or suggest a *raw play stream* (claim 5). *Id.* ¶ 116.

V. Ground 2 Does Not Render any Challenged Claims Unpatentable

A. Petitioner fails to show the asserted combination would have been obvious.

Petitioner asserts that it would have been obvious to combine Ramaswamy with Conklin (“Ramaswamy-Conklin”), under the rationale that the combination is simply the application of known techniques to a known system or method ready for improvement, yielding predictable results. Petition, 56. Petitioner’s analysis is fundamentally flawed.

Petitioner first claims that Ramaswamy and Conklin “relate to monitoring media consumption” and to solving “the problem of monitoring consumption when the media is viewed in a time shifted manner.” *Id.* Even if true, this proves only that the references are analogous, which is insufficient to show a motivation to combine. *See, e.g., Trend Micro, Inc. v. Open Text Inc.*, IPR2023-00692, Paper 8 at 29-30 (PTAB November 6, 2023) (citing *K-TEC, Inc. v. Vita-Mix Corp.*, 696 F.3d 1364, 1375 (Fed. Cir. 2012) and *Microsoft Corp. v. Enfish, LLC*, 662 F. App’x 981 (Fed. Cir. 2016) and explaining that showing that references relate to the same field and provide solutions to the same problem relates to whether they are analogous, which is “merely a jumping-off point” in determining obviousness and “fall[s] short of providing a motivation to combine”).

Petitioner more specifically advances a motivation to combine based on the desire to provide Conklin's "ability to determine more complex time shifting viewing behaviors" to Ramaswamy. Petition, 57-58. Petitioner specifies that this would merely entail applying Conklin's sequence of events analysis to Ramaswamy's method. *Id.*, 58.

Conklin's system relates specifically to tracking events on a digital recording device, and requires access to the internal data bus that transmits information to the device's hard drive and media display devices. EX1010 [0022]-[0030]. Thus, Conklin's information monitor/analyzer "reads (i.e., snoops) control or command information" transmitted on the data bus between a mass storage device (hard drive) and a local processing unit (which processes received audio/video data and sends it to the mass storage device or to viewing devices) to "identify sets of digital information or digital information packets to be stored on or retrieved from the mass storage device." *Id.* [0027], [0024]. Conklin describes no means to identify the media content or its boundaries (when one program has ended and another started). Rather, it merely infers that data packets contain audio or video information by their size. *Id.* [0039]; Ex. 2001 ¶ 121.

The monitor/analyzer records source information (e.g., channel number) as well as storage location of the data packets on the hard drive when it recognizes that digital data is transferred on the bus. EX1010 [0029]-[0032]. This is what Conklin describes

as generating its “sequence of events record,” which “reflect[s] a series of data transfers between the local data processing unit” “and the mass storage device.” *Id.* [0031]. The sequence of events record also records current times (time stamps) for recording and playback/viewing times for each data packet, which are used to determine time shifting activities. *Id.* [0042]-[0045], [0070]-[0071]. The events record “enables tracking of read and write operations” on the hard drive, which can be “indexed by the storage location on the mass storage device.” *Id.* [0045]-[0046].⁵ Thus, Conklin relies on analyzing data packets sent to and from a mass storage device, with indexing based on their storage location. As mentioned above and as further detailed in Section V.B.4. below, Conklin does not disclose or suggest any knowledge or concern with the beginning (or any other boundaries) of any particular piece of media content contained in the indexed data packets, or any method of determining same. Rather, it only discloses determining the source and size of the data packets. Ex. 2001 ¶ 123.

Ramaswamy, on the other hand, identifies media sources by using signature information generated based on the media presented, including not only the source but

⁵ Conklin alternatively describes embodiments where the storage information is inserted into the data packet headers and not stored in the sequence of events record. EX1010 [0029]. There is no distinction between these alternatives relevant to the issues here; the point is that Conklin indexes based on storage location and not content.

also the content. EX1006 [0019], [0024], [0030]-[0031]; *see also* Section V.B.2 below. Ramaswamy identifies traditionally time shifted content based on matching of these signatures between recording times and playback times, represented by time stamps. *Id.* [0031]-[0034], [0036]-[0038], [0040]-[0041]. Matching signatures are assumed to represent the same media content. *Id.* [0060].

Other than generally asserting that Conklin's sequence of events record would be added to Ramaswamy, Petitioner does not explain how the record could be used in Ramaswamy's signature-based system to determine "more complex time shifting behaviors" (e.g., fast forwarding, rewinding, pausing, skipping). Conklin's sequence of events record relies on the read and write times to and from the hard drive and does not reflect whether a specific record denotes the beginning, middle, or end of any content. *See* Section V.B.4. Ramaswamy, relying on content signatures, cannot use any information in Conklin's sequence of events record to determine time shifting behaviors because that record is not indexed to content, but to storage location on the mass storage device. Ex. 2001 ¶ 125.

It is unsurprising that Petitioner fails to explain this aspect of the proposed combination given that Conklin itself specifically teaches that content signature based systems—such as Ramaswamy—cannot be used to identify the "more complex time shifting behaviors." Conklin teaches that:

The use of processing techniques **based on recognition of program signatures** has been employed in some cases where it is difficult to obtain accurate tuning information and in cases where it may otherwise be difficult to obtain viewing time information and channel information associated with the currently viewed program content. **Unfortunately, these techniques are not well suited for use in situations where a recording device enables rapid changes between live viewing of a program, delayed viewing of the program, viewing of another program while recording the program, etc.**

EX1010 [0007]. A POSA reviewing Conklin would thus be discouraged from following the path proposed by Petitioner in combining it with Ramaswamy to achieve the claimed invention, based on Conklin's explicit disparagement of systems like Ramaswamy that rely on recognizing content signatures as being "not well suited" to determine "more complex time shifting behaviors" such as those referenced in Conklin. Ex. 2001 ¶ 126. Accordingly, Conklin teaches away from Petitioner's proposed combination, negating any alleged motivation to combine.

Further, as noted in *KSR* when discussing the same rationale advanced by Petitioner here, when the actual application of the technique would have been beyond the skill of a POSA, then using the technique would not have been obvious. *KSR Int'l Co.*, 550 U.S. at 417. Petitioner asserts that there was a reasonable expectation of success based on the fact that both Ramaswamy and Conklin "relate to processing data,"

and “teach that data can be stored in well-known data structure formats and then analyzed” “using [v]arious [well known] database processing techniques.” Petition, 58. Petitioner thus focuses solely on whether a POSA would have known how to store, analyze, and process the data in Conklin’s sequence of events record for use in Ramaswamy. But that is not all that the proposed combination necessarily entails.

As mentioned above, Conklin’s sequence of events record only records source information, storage location, and time stamps for the individual data packets read from or written to its hard drive. It contains no teaching of how to determine the beginning or end of any specific media content. *See* Section V.B.4. Instead, it uses the delay time for specific packets of data (the difference between playback or viewing time and record time) without reference to their content. EX1010 [0071]. Any combination with Ramaswamy must explain how a POSA could use the information in the events record to detect “more complex time-shifting behavior” in Ramaswamy’s system, given that Ramaswamy instead relies on identifying content using signatures. Ex. 2001 ¶¶ 127-128. The Petition provides no such explanation or evidence in support. Accordingly, a POSA would not have considered there to be a reasonable expectation of success in detecting these behaviors in Ramaswamy’s system using Conklin’s blind reading of hard drive data packets. *Id.*

B. Ramaswamy-Conklin Does Not Disclose All Limitations of the Challenged Claims

1. Ramaswamy-Conklin does not teach or suggest *video data samples* or *submitted data samples* – Claims [1.1], [9.1]

Petitioner identifies Ramaswamy's output signal 210 as the claimed *video data samples*. Petition, 59-60. Ramaswamy explains that the output signal is received by the signature information generator directly from the media delivery device. EX1006 [0030]. This mirrors the description in the challenged patent for data (whether raw or processed) captured by a monitoring device and uploaded to the system. EX1001, 3:45-62. As discussed, in the challenged patent this data is divided into samples for subsequent analysis. *See* Section III.A; EX1001, 4:58-5:5. However, Petitioner identifies nothing in Ramaswamy disclosing or suggesting that its output signals are ever divided into samples, for subsequent content recognition analysis (or any other purpose), as described in the patent. The division of raw data into multiple samples for analysis is not necessary in ACR systems. Therefore, it cannot be considered as somehow necessarily present in Ramaswamy (which Petitioner does not even argue). Ex. 2001 ¶¶ 129-130.

Petitioner does not specifically identify the *submitted data samples* here, but merely states that Ramaswamy sequentially generates signature information from them (whatever they are). Petition, 60. As discussed, in the challenged patent *submitted data samples* are selected from the raw or processed data samples. *See* Section III.A. Just as

Ramaswamy does not teach or suggest dividing the data into data samples, it does not teach or suggest submitting any such samples for signature generation (and content identification). Ex. 2001 ¶ 132. Rather, Ramaswamy describes generating signatures based on analysis of the entire raw audio/video output signal. EX1006 [0030]-[0031]. Thus, Ramaswamy does not disclose the use of *a sequence of submitted data samples, the submitted data samples corresponding to video data samples*. Accordingly, Ground 2 cannot render any challenged claim obvious.

2. Ramaswamy-Conklin does not teach or suggest *utiliz[ing] sample sequence data and content offset data to deduce play-altering actions of the monitored audience member* – Claims [1.2], [9.2]

Petitioner does not rely on any disclosure in Conklin as disclosing or suggesting the *content offset data* required by elements [1.2] or [9.2] (or [1.4] or [9.5]). Petition, 62-65, 66-68. Petitioner asserts that Ramaswamy alone discloses or renders this element obvious when *play-altering actions* includes traditional time shifting activity, and that Ramaswamy-Conklin renders it obvious when *play-altering actions* must include more complex actions.⁶ *Id.*, 62-65. Under both theories, Petitioner asserts that Ramaswamy's use of time stamps, contained in its signature information to detect time shifted media

⁶ Precise construction of *play-altering actions* on this point is unnecessary because, as shown herein, none of the asserted grounds render any claim unpatentable even using the broader scope.

consumption, discloses the use of *content offset data* as required by the claims. *Id.*, 63-64. As shown below, Ramaswamy simply does not disclose or suggest the use of *content offset data* in any way.

The '896 Patent clearly teaches that a *content offset* “represents a time offset from the beginning of a particular piece of content.” EX1001, 11:41-43. Petitioner interprets this as meaning a determination “in relation to the beginning time of reference content.” Petition, 66-67. This is correct. A content offset, as suggested by its name and as specifically explained in the specification, is wholly relative—the offset or difference between the beginning time of a sample within a particular piece of content from the beginning time of that specific content. Ex. 2001 ¶ 134.

Despite this clear disclosure, Petitioner attempts to twist the term’s meaning to encompass absolute times by citing a portion of the specification having nothing to do with content offsets. Petitioner thus asserts that *content offset* “is characterized broadly in the '896 patent, such that it may be determined ... from the ‘absolute time’ for a broadcast channel.” Petition, 66-67 (quoting EX1001, 10:10-19). But the cited passage is not about content offsets. It references Figure 4, and specifically channel data 470, which “relates a content ID to a start time and end time,” which are relative to an “arbitrary start time value.” EX1001, 9:66-10:19. While the passage states that in some cases, the “absolute time might be both available and useful,” this is not referring to content offsets but to absolute start and end times. Indeed, the entire discussion of

Figure 4 begins with the proviso that “[t]he example in FIG. 4 ... **does not rely on content offset data** in scrubbing the raw play stream to obtain a clean play list. However, note that **in other examples, such as that of FIG. 5, sample sequence data ... includes content offset data.**” *Id.*, 9:25-45. Thus, this passage says nothing about content offsets (much less broadly characterize the term as Petitioner asserts), except in referring to the subsequent discussion of Figure 5.

Based on its flawed premise, Petitioner explains that Ramaswamy describes generating signature information for content received by a media source, which is stored in a local reference library. Petition, 67. Petitioner acknowledges that the time stamps that may be included in the signature information “would be the absolute time of the broadcast data”—that is, not *content offset data*, which is relative. *Id.* Despite this, Petitioner inexplicably asserts that a POSA would have “understood that the claimed *content offset* would encompass at least using absolute times, as opposed to arbitrary reference times, for broadcast channels.” *Id.*, 67-68. This makes no sense, as a *content offset* is a relative value internal to the content itself, and has nothing to do with absolute times. Ex. 2001 ¶ 136.

Petitioner next asserts that a POSA would have understood that the “the difference between the time stamp of an identified segment of the broadcast ... and the time stamp of the beginning of that broadcast corresponds to the claimed *content offset*.” *Id.*, 68. Petitioner’s analysis concludes with the proclamation that, by considering the

relative broadcast times of “that particular piece of reference signature information” (corresponding to the broadcast segment) and the “recorded program as a whole,” the “*time position* of that section ... within the content as a whole ... **can be determined.**” *Id.* Thus, Petitioner asserts that the *content offset* for each broadcast segment **can be calculated** using this information, and therefore Ramaswamy (somehow) renders **utilizing** content offsets obvious. *Id.*

As an initial matter, Petitioner does not explain where Ramaswamy discloses time stamping both “an identified segment of the broadcast” and the “time stamp of the beginning of that broadcast.” Petition, 67-68. Petitioner points only to the time stamp for signature information 220 as indicating the “time at which that particular section of the recorded program was broadcasted [sic].” *Id.*, 68. But Ramaswamy does not discuss time stamps for particular media segments, and instead only broadly states that signatures are generated for “each program, channel, or, more generally, any and all [presented] media content.” EX1006 [0033]. Petitioner provides no evidence explaining the validity of its apparent assumption that this encompasses generating signatures (and associated time stamps) for individual program segments, as Dr. Russ’s declaration merely parrots the Petition. EX1003 ¶ 263. On the contrary, Ramaswamy appears to indicate that broadcasts are stored in its reference database as a single signature, which would have a single time stamp. EX1006 [0019] (referencing “media (e.g., a television **program**, a radio **program**, etc.”)), [0024] (referencing “identifying media content”

with “reference signature information associated with media content broadcast on a specific channel at a specific time” and specifically indicating a (singular) “**time** at which that **media content** was received at the reference site”), [0032] (referencing storage of “signature information corresponding to **thirty minutes**” of content), [0036] (referencing “a signature” and “a time stamp” when media content is played back), [0039] (discussing usage of “fixed preamble sounds” to generate signatures to identify entire associated games, not segments of such games); Ex. 2001 ¶¶ 138-139. At the very least, Ramaswamy is not clear on this point, and Petitioner fails to provide any evidence showing that it teaches or suggests time stamps that correspond to specific broadcast segments.

For argument’s sake, even if a POSA could determine the content offsets based on Ramaswamy’s time stamps, this does not mean that Ramaswamy itself teaches or suggests **utilizing** content offsets to do anything. It does not matter that a POSA **could have** calculated content offsets, but only whether a POSA **would have** modified Ramaswamy to do so to arrive at the claimed invention. *See* Section IV.5 (citing *Belden Inc.*, 805 F.3d at 1073; *Virtek Vision Int’l ULC*, 97 F.4th at 888; *PersonalWeb Techs., LLC*, 848 F.3d at 993-94).

When Ramaswamy monitors for time shifted information, the library generator associated with the media source generates signature information for the media content when it is first recorded, which is then stored in the reference library. EX1006 [0036].

When that content is ultimately played back, the signature information generator generates a signature for the played content from the media delivery device. *Id.* [0038]. In both cases, time stamp information—the absolute time the recording or playback occurs—is included in the signature information. *Id.* [0036]-[0038], [0050]. Ramaswamy’s matcher compares the time stamps, and when they are the same or substantially similar the matcher compares the signature information associated with the matching time stamps. *Id.* [0038].

The matcher sends matcher collections to a “time shift analyzer” which converts each collection to a “time shift analysis data structure.” *Id.* [0040]. The analyzer flags the collection in the data structure to denote that it includes time shifted information. *Id.* [0040]-[0041]. The analyzer determines whether there is time shifted information by determining whether two records have the same signature values but different time stamp values. *Id.* [0046]. Thus, the time shift analysis renders a binary determination (contains or does not contain time shifted information). *Id.* [0040]-[0041]; *see also id.*, [0062]. The time shift analysis data structure thus captures whether a “tuning time” (the time the media is viewed) is “subsequent to the time at which the media content was downloaded, stored, recorded, etc.” *Id.* [0066].

Thus, Ramaswamy relies solely on comparing the absolute times shown in the time stamps; when the playback time is subsequent to the download/recording time, Ramaswamy determines that time shifting activity has occurred. Ramaswamy never

calculates a content offset, and therefore never uses content offsets in any part of its disclosed method or apparatus. Ex. 2001 ¶¶ 140-143.

Although Petitioner asserts that Ramaswamy alone renders the **use** of *content offset data* obvious, nothing in the Petition (or Russ Declaration) contains any reasoning or rationale for **why** a POSA would have used them in Ramaswamy. Ramaswamy accomplishes its stated goal of measuring time-shifted media consumption by merely considering absolute time stamps—in the vernacular of the '896 Patent, log times—not content offsets. Petitioner provides no reasoning or explanation for why a POSA would choose to incorporate content offsets in Ramaswamy when it already achieves its goal by considering only the absolute time stamps of recording and playback. Thus, Petitioner has failed to show that Ramaswamy teaches or renders obvious “*to utilize ... content offset data to deduce play-altering actions.*”

As previously noted, Petitioner relies solely on Ramaswamy to disclose *content offset data*, citing Conklin only as a method of identifying play-altering activities other than traditional time shifting. Petition, 63-65. But, as discussed in Section V.B.4 below, Conklin does not disclose using content offsets either. Merely claiming that a POSA **could have** used content offsets in Ramaswamy (or Conklin) does not show that a POSA **would have** used them in the asserted combination. Ramaswamy-Conklin thus does not render any challenged claim unpatentable.

3. Ramaswamy-Conklin does not teach or suggest *content offsets ... indicating a time position of a content identification result within identified content* – Claims [1.4], [9.5]

Ramaswamy does not teach or suggest the use of content offsets at all. *See* Section V.B.2. Ramaswamy’s time stamps are absolute times, and therefore do not denote *a time position ... within identified content*. Even if a POSA could determine a content offset for a particular segment by referring to the time stamp for the recorded broadcast “as a whole,” (Petition, 68), Petitioner provides no reason for a POSA to have done so. *See* Section V.B.2. Nor does such a reason appear in the Russ declaration, which only parrots the Petition’s conclusory assertion of obviousness. EX1003 ¶¶ 263-265.

4. Ramaswamy-Conklin does not teach or suggest *comparing a progression of log times associated with capture timing of each of the particular plurality of submitted data samples and a progression of content offsets of the corresponding content identification results* – Claims [1.5], [9.6]

Petitioner maps Ramaswamy’s time stamps for consumed signature information⁷ to *log times associated with capture timing*. Petition, 69. Petitioner asserts that these

⁷ For this element, Petitioner asserts that “signature information 217 corresponds to the *submitted data samples*.” Petition, 69. However, for element [1.1], Petitioner stated that the signature information was generated **from** the submitted data samples (Petition, 60), which as explained in Section V.B.1. were not themselves identified.

are compared to a *progression of content offsets* “determined based on the time stamps in the reference signature information.” *Id.* Again, however, Petitioner identifies nothing in Ramaswamy that actually calculates or utilizes this alleged offset determination. *See id.*; *see* Section V.B.2.

Petitioner asserts that “it would have been obvious to a POSA that a *progression* of such comparisons will be made because a single comparison would not be useful in the context of Ramaswamy’s system.” Petition, 69-70. But while Ramaswamy may compare the time stamps of its consumed and reference signature information, this does not involve *content offsets* as required by the claim. *See* Section V.B.2. That is, Ramaswamy never makes a single comparison between its consumed signature time stamp (the asserted *log time*) and any calculated *content offset*. Ex. 2001 ¶ 148. And although Petitioner asserts that it was obvious to “compar[e] the collected data in a slightly different way” (that is, compare consumed signature time stamps with content offsets calculated from reference signature time stamps, something that Ramaswamy neither does nor suggests), the only support for this statement is a citation to the Russ Declaration. *Id.*, 70. But Dr. Russ testifies only that it was well known to **compare time stamps**—which Petitioner admits are log times, and which Ramaswamy already does. He never asserts that it was well known to compare log times **to content offsets** as required by the claim. EX1003 ¶ 274 (“To the extent not expressly disclosed by Ramaswamy, a POSA would know that **comparing time stamps** ... would be a simple

way to detect ‘trick play’ behavior such as fast-forward or rewind.”). This does not suggest comparing *a progression of log times* with *a progression of content offsets* as required by the claims. Ex. 2001 ¶ 149.

Moreover, Petitioner’s “progression of comparisons” (Petition, 69-70) does not meet the claim language, which requires a comparison of progressions. The former is a series of comparisons between (a) a *single* log time and (b) a *single* calculated content offset. Ex. 2001 ¶ 150. The latter is a comparison between (a) a *progression* of log times and (b) a *progression* of content offsets. *Id.*

Per Petitioner, and assuming that Ramaswamy actually used content offsets (it does not), Ramaswamy’s “stream of matcher collections” sequentially compares, for each matched content segment, the consumed signature time stamp (alleged *log time*) with the difference between the reference signature time stamp for the content segment and the reference signature time stamp for the beginning of the content as a whole (alleged *content offset*). Petition, 70. Each comparison thus requires three data points—the consumed signature time stamp and the two reference signature time stamps used to determine the content offset. Ex. 2001 ¶ 151.

The claimed *comparing a progression of log times* with the *progression of content offsets* requires, not this series of singular comparisons, but a comparison of the changes (progressions) **between** those singular comparisons. EX1001, 11:59-12:10; Ex. 2001 ¶ 152. Using Petitioner’s framework, it would require at least five data

points—*log times* for two consumed signatures and three reference signature time stamps (one for the beginning of the content as a whole, and one for each matched segment). *Id.* Nothing in Ramaswamy, or Petitioner’s analysis, suggests making such a comparison of progressions.

Finally, assuming with Petitioner that Ramaswamy were modified to compare time stamps of consumed signature information with content offsets calculated from the time stamps of reference signature information (“comparing the collected data in a slightly different way”) (Petition, 70), Ramaswamy would no longer be capable of determining even traditional time shifting. Ramaswamy determines time shifting by detecting differences between playback and recorded times. *See* Section V.B.2. If Ramaswamy were instead comparing playback times to calculated content offsets, it could no longer detect this difference, as the relative content offset values would always be different from the playback times—indeed, it becomes an apples to oranges comparison. Ex. 2001 ¶ 153. Thus, the proposed modification renders Ramaswamy inoperable for its intended purpose, negating obviousness. *In re Gordon*, 733 F.2d 900, 902 (Fed. Cir. 1984).

Recognizing that Ramaswamy does not disclose or suggest this limitation, Petitioner relies on Conklin to argue obviousness. Petition, 70. Petitioner first states that the “View Time” on the horizontal axis of Conklin’s Figure 13 is an absolute local time denoting the playback time—a time stamp just like those disclosed in Ramaswamy.

Petition, 71. Petitioner maps this to the *progression of log times*. *Id.* Petitioner further asserts that the vertical axis represents the delay time—the difference between the view time and the recording time. *Id.*, 72. Petitioner does not map this delay time to the claimed content offsets, but asserts that the claimed content offsets “are a function” of the record time, and that “[i]t would have been obvious, **knowing the start time of the full program** and the broadcast/record time of the identified segment of content, to determine the content offset therefor.” *Id.*, 72. Based on this, Petitioner asserts that Conklin’s record times render obvious the claimed progression of content offsets. *Id.*

First, Conklin’s view/playback times are not *associated with capture timing of each of the ... submitted data samples*. Petitioner acknowledges that the *log time* is the time the sample started to be captured. Petition, 71; *see* Section IV.2. But the view/playback times have nothing to do with capture timing. They instead reflect the time at which Conklin’s information monitor/analyzer determines that “a **previously stored** packet of digital information is to be retrieved” from the storage device for playback. EX1010 [0033]. The monitor/analyzer merely associates this time with the event record or adds it to the packet header. *Id.* But the determination that a stored packet is to be played back is not the time at which that data/sample was captured. That occurred earlier—when it was first stored on the storage device by the DVR’s local data processing unit. *Id.* [0024] (“The local data processing unit 22 processes received signals and stores the processed signals as digital information on the mass storage

device 24 ... sends the processed signals ... to one or more devices such as ...a television, for live viewing, and/or retrieves and outputs ... digital information previously stored on the mass storage device 24.”). Thus, Conkin’s view/playback times do not satisfy the claimed *progression of log times associated with capture timing of each of the ... submitted data samples*. Ex. 2001 ¶ 156.

Further, assuming that Conklin “know[s] the start time of the full program” and the “broadcast/record time of the identified segment,” (as discussed below, it does not), Petitioner does not assert that Conklin actually discloses content offsets, or the use of them. Rather, Petitioner asserts (as it did for Ramaswamy) that it was obvious **to determine** content offsets using this information. Again, even if true, Petitioner provides no explanation for **why** a POSA would have determined content offsets to deduce any *play-altering actions*, because Conklin already did so without them. Ex. 2001 ¶ 157. Petitioner’s reasoning here fails for the same reasons explained for Ramaswamy above. Only hindsight bias, using the invention of the ’896 Patent as a roadmap, supports a POSA even calculating (much less using) content offsets in Ramaswamy or Conklin—and only for use in mapping to the challenged claims, not to solve any problem presented in either reference (or identified by Petitioner).

Petitioner’s combination also fails because neither reference actually discloses or suggests using content offsets in any way. Petitioner does not identify where either reference teaches the calculation or use of content offsets, but instead asserts only that

it would have been obvious in both. While it may be a simple matter to **calculate** a *content offset* (given the correct inputs), Petitioner provides no rationale or reasoning to support a finding that the **use** of content offsets was obvious, but offers only conclusory assertions as discussed. Ex. 2001 ¶ 158. Nor does Petitioner provide any evidence showing that the **use** of content offsets in this manner was within the general knowledge of a POSA or would have been common sense. Petitioner thus supplies this missing limitation out of thin air.

Finally, it is in fact **impossible** to determine content offsets in Conklin, even under Petitioner's theory. Ex. 2001 ¶¶ 159-165. This is because a content offset is not merely any relative (offset) time—it is an offset from the beginning of a particular piece of media **content**, such as the beginning of a song or television program. EX1001, 11:29-46. And Conklin does not disclose any way to determine the beginning of any piece of content or any particular content segment, or that Conklin's system already knows (or cares) about this information.

Conklin discloses using an information monitor/analyzer which reads the digital packets transmitted on a data bus between a mass storage device (hard drive) and a local processing unit (which processes audio/video data and sends it to the mass storage device or to viewing devices). EX1010 [0024], [0027]-[0028]. This information contains "storage location information" (the physical memory location on the hard drive, or a logical block address) for the data packets. *Id.* [0028]. For the data packets,

the monitor/analyzer can also store time information—a time stamp containing a current time. *Id.* [0029], [0031]. Alternatively, the time and storage location are stored as a header in the data packets. *Id.* [0029], [0031]. The monitor/analyzer also includes source information (e.g., channel number) with the storage location and time stamp. *Id.* [0032].

The monitor/analyzer thus monitors the data bus to recognize when digital data is transferred—the individual read/write events where information is recorded or retrieved. *Id.* [0030]. This is described as generating “a sequence of events record reflecting a series of data transfers between the local data processing unit” “and the mass storage device.” *Id.* [0031]. When an event record having the same storage location information is found in the sequence of events record, “a playback time, which is preferably a current local time, is associated” with the event record. *Id.* [0033].

The record generator thus records source, time and storage location information uniquely corresponding to each packet written to the mass storage device. *Id.* [0042]. It may store this information in memory in a lookup table, which may be used to modify the event record to include a read or playback (viewing) time. *Id.* [0044]. This is the “sequence of events record,” “which enables tracking of read and write operations.” *Id.* [0045].

The sequence of events record is thus “used to maintain a record of write or record events **that is indexed by the storage location on the mass storage device ...**

In other words, for each storage location (e.g., each logical block address) on the mass storage device” “there is a corresponding event record containing the time at which the [data] associated with the write operation was written or recorded on the mass storage device” “as well as the source or channel from which the audio, video and/or image information within the data packet was originally derived.” *Id.* [0046]. When played back, “a current local time is added to the event record” to record the event. *Id.*

Thus, when media content is to be recorded on (written to) the mass storage device, the monitor/analyzer’s access detector determines the source and current time and generates an event record containing that information. *Id.* [0055]. The event record is stored in the sequence of events record. *Id.* When that stored event is to be played back (read from) the mass storage device, the access detector determines the locations on the storage device from which the data packets associated with the event are to be read. *Id.* [0057]. The event record generator uses the location information to look up the previously stored event record corresponding to that location in the sequence of events record/look-up table. *Id.* The generator adds the current local time to the event record. *Id.*

Thus, Conklin tracks the precise storage location for each individual data packet written on its system’s hard drive. But there is no identification in Conklin’s sequence of events record that a particular data packet marks the beginning of a particular piece of content. Nor does Petitioner identify anything in Conklin that it could use to make

such a determination—or any reason it would need to do so. Thus, there is no mechanism by which Conklin can even calculate a *content offset* using Petitioner’s methodology. Ex. 2001 ¶¶ 164-165.

Petitioner’s assertion that it was obvious to determine the content offset in Conklin explicitly depends on “knowing the start time of the full program and the broadcast/record time of the identified segment of content.” Petition, 72. Not only does Petitioner not show this was done in Conklin, as shown it could not be done based on Conklin’s disclosure. Thus, Conklin’s record times cannot render obvious the claimed *progression of content offsets of the corresponding content identification results*.

For the foregoing reasons, Ramaswamy-Conklin cannot render any claim unpatentable.

5. Ramaswamy-Conklin does not teach or suggest *generating a play list including content offset data as part of deducing media play-altering actions of a monitored audience member* [Claim 2]

Petitioner relies solely on Conklin’s disclosure for Claim 2. Petition, 73-74. Petitioner asserts that it was obvious to insert content offsets in Conklin’s sequence of events analysis, and therefore obvious to include them in Conklin’s sequence of events record (which Petitioner characterizes as a play list) “in place of or in addition to the record time information.” *Id.*, 74. Petitioner does not provide any rationale for doing so. *Id.*, 73-74. The only apparent evidence for Petitioner’s conclusory assertion of obviousness is an identical passage in the Russ Declaration. EX1003 ¶ 286. Petitioner

thus provides no supporting rationale necessary to support a conclusion of obviousness for claim 2. Nor can it, because Conklin does not teach or suggest using content offsets in any way, and therefore cannot contain any teaching or suggestion to incorporate them into its sequence of events record. *See* Section V.B.4.

Further, Conklin's sequence of events record is not the claimed *play list*. Unlike a play stream, a play list contains entries for each item of media content (which is useful for content identification in channel matching), not for each individual content sample. *See* Section III.B. Conklin's sequence of events record may (at best) be a play stream (one which does not contain any content identification results), but it is not a play list. The sequence of events record contains the absolute time stamps for specific data packets based on their storage location, not their content. *See* Section V.B.4. It is therefore not a play list. Ex. 2001 ¶ 170.

Accordingly, Petitioner has not shown that claim 2 is unpatentable over Ramaswamy-Conklin.

6. Ramaswamy-Conklin does not teach or suggest *utilizing sample sequence data together with corresponding channel data to deduce play-altering actions of a monitored audience member* [Claim 3]

Petitioner asserts that Ramaswamy alone teaches claim 3, which requires that *sample sequence data* and *corresponding channel data* both be utilized (*together with*) to deduce *play-altering actions*. Petition, 74-75. While Ramasamy does teach using channel data to identify specific television programming contained in media content,

Ramaswamy does not use that channel data to determine whether that content has been time-shifted.

Ramaswamy explains that

collecting and analyzing ... matching signatures also enables the **identification of the delivery paths** of the media content that may be used to develop information about the manner in which that media content was received at a consumption site (e.g., a home), stored for later consumption, and consumed at a later time(s). In an example download, storage and retrieval process, **a set of matching signatures ...** each providing various media source information and time stamp information **may be used to determine that the media content was delivered on a particular channel at a particular time**, then stored on a particular device at a particular time, ... and subsequently displayed one or more times at the direction of the user.

EX1006 [0060]. Thus, Ramaswamy determines whether content was time shifted based solely on its signature matching process. Only after this determination is made does Ramaswamy mention using channel data—not to deduce time shifting as Petitioner asserts, but to efficiently identify the particular programming displayed so that it can be “credited” as having been consumed:

If the initial delivery of the media content corresponds to a broadcast channel having a known lineup, then the signature information associated with the download may be compared to the lineup **to**

identify the television programming contained in the media content in an efficient manner. The identified programming can then be **credited with having been consumed** at the times indicated by the signatures captured in connection with the playback of the programming, i.e., the signatures associated with the subsequent display of the media content.

Id. Ramaswamy thus teaches that “behaviors and characteristics associated with time-shifted media consumption ... can be accurately and efficiently captured and analyzed.”

Id. Therefore, while Ramaswamy uses channel data to capture and analyze “behaviors and characteristics associated with” time shifting, it is not used to deduce that time shifting occurred in the first instance as required by the claim (assuming that time shifting is a *play-altering action*). Ex. 2001 ¶¶ 172-173. Accordingly, Ramaswamy-Conklin does not render claim 3 unpatentable.

7. Ramaswamy-Conklin does not teach or suggest *wherein a clean play list is generated including deduced play-altering actions* [Claim 4]

Petitioner relies on Ramaswamy alone in its analysis of Claim 4, identifying Ramaswamy’s “collection of consumption records,” specifically “consumption reports” that detail viewed content, as including a *clean play list*. Petition, 75. Petitioner specifically asserts that these reports are created by the time shift analyzer and included in Ramaswamy’s time shift analysis data structure. *Id.*, 75-76. While Ramaswamy’s time shift analysis data structure includes the source of time shifted content and a time stamp for both a tuning (recording) time and playback time (EX1006, [0065]), there is

nothing in Ramaswamy indicating that this data structure is a play list, much less a *clean play list*.

Rather, the data structure is at best a raw play stream, providing sample content and time stamps. Ramaswamy teaches that its signature information generator “sequentially generates reference signatures for each program, channel or, more generally, any and all media content,” with the signatures representing the media content for media played on a delivery device. EX1006 [0030]-[0031], [0033]. That information is used by Ramaswamy’s matcher for comparison with signature information in the local reference libraries. *Id.* [0032]-[0034]. Resulting signature matches are included in a matcher collection. *Id.* [0033]-[0034]. This matcher collection may be sent to the time shift analyzer, resulting in a time shift analysis data structure which denotes whether or not the matches contain time shifted information. *Id.* [0040], [0046], [0062]. This data structure may be stored in a consumption record, which may include signatures associated with the time shift analysis data structure, with collections thereof sent to a central processing facility. *Id.* [0041]. This association enables “the identity of media content” to be determined at the facility. *Id.*

Assuming that Ramaswamy discloses generating signatures for content segments—which is not apparent as discussed in Section V.B.2 above—its description signifies, at most, that its matcher collections and associated time shift analysis data structures are a raw play stream containing sequentially generated signatures for the

consumed media content—a series of signatures for content as it is consumed and a notation of whether there was time shifting. Ex. 2001 ¶¶ 176-177. This is the same type of data set described in the '896 Patent as a raw play stream, which contains a sequential listing of selected media data samples, as they are played, with first-pass content identification results (or denoting that no initial content identification was determined).⁸ EX1001, 6:24-37, 7:26-48, 8:13-34. Indeed, this is exactly how Petitioner maps Ramaswamy's matcher collections for element [1.1]. Petition, 61-62. Even under Petitioner's unsupported interpretation of Ramaswamy (that it generates signatures for individual content segments), it contains no disclosure that can be reasonably construed as analyzing or condensing the sequentially generated signatures or signature matches into a structure identifying entire programs, songs, or ads, versus a stream of data containing only portions of such items. *See* Section III.B. Thus, Ramaswamy does not teach or suggest a *play list*.

Even if any of these data structures are play lists, nothing in Ramaswamy indicates that they are ever scrubbed to convert them into *clean* play lists. Although

⁸ Alternatively, if Ramaswamy generates signatures for entire programs versus mere portions (as discussed in Section V.B.2), then its matcher collections are not play streams, but are play lists. Ex. 2001 ¶ 177 n.10. In that case, Ramaswamy does not teach *generat[ing] a play stream* under claims 1 or 9 (elements [1.1], [9.1]).

Ramaswamy states that its data structures may be stored in consumption records (which may be called a “consumption report,” “viewing records,” or something else) sent to a central facility, (EX1006 [0041]), Ramaswamy never suggests that these records are ever scrubbed to remove or correct misidentified or unidentified media content. Ex. 2001 ¶ 178. Petitioner’s conclusion that Ramaswamy’s consumption records include a clean play list thus lacks support. Therefore, Petitioner fails to show that Ramaswamy-Conklin renders Claim 4 unpatentable.

8. Ramaswamy-Conklin does not teach or suggest *determining if one or more discontinuities exist in the progression of content offsets* - Claim 7

Petitioner identifies the claimed *discontinuities* as “identify[ng] non-continuous values in the **progression of content offsets**.” Petition, 77. As for element [1.5], Petitioner asserts that “the claimed *content offsets* are a function of Conklin’s record time.” *Id.*, 78. As discussed, not only does Conklin not teach using content offsets, it is impossible to determine content offsets based on Conklin’s record time. *See* Section V.B.4. Thus, even if the transition points in Conklin’s Figure 13 (Petition, 78) correspond to discontinuities in media consumption, Conklin does not disclose determining them in relation to the claimed *progression of content offsets*.

Ramaswamy-Conklin thus fails to render claim 7 obvious.⁹

9. Ramaswamy-Conklin does not teach or suggest Claims 12 or 13 because it does not disclose *including content offset data in one or more of a play list or play stream.*

As discussed, Ramaswamy-Conklin does not disclose or suggest the use of *content offset data*. Section V.B.2, V.B.4. Thus, even (incorrectly) assuming that there is “no real distinction” between a play stream and a play list (Petition, 82-83), the combination does render either claims 12 or 13 unpatentable. Further, it is unreasonable to construe these plainly different terms as synonyms. Section III.B.

VI. Ground 3 Does Not Render Any Challenged Claim Unpatentable

A. Petitioner fails to identify any reason to “improve” Feininger with Conklin.

As for Ground 2, Petitioner first asserts a motivation to combine based on the fact that Feininger and Conklin are analogous. Petition, 83-84. For the same reasons discussed there, this is insufficient. *See* Section V.A.

Petitioner further asserts a motivation to add Conklin’s sequence of events analysis to Feininger to “improve a similar method” in the same way. Petition, 84. But

⁹ The Petition does not even assert that the combination does so, only that a POSA would understand that Conklin’s transition points correspond to discontinuities. Nor does the Petition ever cite Dr. Russ’s unsupported conclusion (EX1003 ¶ 296) that Ramaswamy-Conklin discloses or renders obvious this claim element.

Petitioner never explains how Conklin’s analysis would “improve” Feininger. Petitioner asserts that Feininger already possessed the ability to detect whether recorded content was paused, skipped, fast forwarded, or rewound. *See* Petition, 48-49, 51-53. Indeed, if Feininger was already able to perform the functions allegedly enabled by Conklin’s sequence of events analysis, there can be no reason to add Conklin’s functionality to Feininger. Ex. 2001 ¶¶ 188-189. Except, of course, Petitioner’s hindsight-driven motivation to enable mapping the combination to the challenged claims.

B. Ground 3 does not disclose the limitations of Claims 2, 7, 12, or 13.

Petitioner relies on Conklin as disclosing or suggesting the elements of claims 2, 7, 12, and 13. Petition, 85. Conklin cannot do so because it is impossible to even calculate the content offsets required by these claims, as further explained for each of these claims in Ground 2. *See* Sections V.B.4, V.B.5, V.B.8, V.B.9. Accordingly, Ground 3 does not render any of these claims unpatentable.

VII. Conclusion

Because the Petition fails to show a reasonable likelihood of proving any challenged claim unpatentable, AMR respectfully requests the Board deny institution.

Dated: November 14, 2024

Respectfully submitted,

/James H. Hall/
Counsel for Patent Owner
Anonymous Media Research Holdings,
LLC

CERTIFICATE OF COMPLIANCE

The undersigned certifies that this document complies with the type-volume limitations set forth in 37 C.F.R. § 42.24 *et seq.* This document, excluding those portions excluded under § 42.24(a)(1), contains 13,981 words as calculated by the program used in creating it.

/s/ James H. Hall
Counsel for Patent Owner
Anonymous Media Research Holdings,
LLC

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. §§ 42.6(e)(4), the undersigned certifies that on November 14, 2024, a copy of the foregoing document (*Patent Owner's Preliminary Response*, including all Exhibits), was served via electronic mail on all attorneys of record at the email addresses provided for electronic service listed in Petitioner's mandatory notices, as set forth below:

lkenton-PTAB@sternekessler.com
cobrien-PTAB@sternekessler.com
jchagnon-PTAB@sternekessler.com
jharrison-PTAB@sternekessler.com
syedahmed-PTAB@sternekessler.com
PTAB@sternekessler.com

/s/ James H. Hall