

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

SPORTRADAR AG,
Petitioner

v.

SPORTSCASTR INC.,
(D/B/A PANDA INTERACTIVE),
Patent Owner.

IPR2025–00314
U.S. Patent No. 10,425,697

PETITION FOR *INTER PARTES* REVIEW UNDER 37 C.F.R. §42.101

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	MANDATORY NOTICES UNDER 37 C.F.R. §42.8(A)(1)	3
	A. Real Parties–in–Interest.....	3
	B. Related Matters.....	3
	C. Lead and Back–up Counsel and Service Information.....	4
III.	REQUIREMENTS FOR IPR UNDER 37 C.F.R. §42.104	5
	A. Grounds for Standing	5
	B. Identification of Challenge and Relief Requested	5
	1. Claims for Which IPR Is Requested	5
	2. Grounds for Challenge	5
	3. Claim Construction	6
	4. How the Claims Are Unpatentable	6
	5. Evidence Supporting This Challenge	7
IV.	TECHNOLOGY BACKGROUND: PROTOCOLS FOR REAL– TIME OR LOW–LATENCY DATA TRANSMISSION.....	7
	A. HTTP Polling	7
	B. WebSocket.....	8
V.	THE ‘697.....	8
	A. Alleged Problem.....	8
	B. Alleged Invention	9
	C. Prosecution History	12
VI.	LEVEL OF ORDINARY SKILL IN THE ART	13

VII. OVERVIEW OF THE PRIMARY REFERENCES	13
A. Ellis.....	13
B. Herzog	15
VIII. EACH OF THE CHALLENGED CLAIMS IS UNPATENTABLE	17
A. Ground 1: Claims 23 and 24 are Obvious Over Ellis in view of Spivey	17
1. Motivation to Combine	17
2. Claim 23	22
a. Element 23.0	22
b. Element 23.1	23
c. Element 23.2	25
d. Element 23.3	29
i. First event socket corresponds to the first event information germane to the first live sporting event	34
ii. Transmitting the first score information to at least the first VCD	37
iii. Transmitting via a first event information communication channel between at least one first event socket of the at least one socket server and the first VCD of the first plurality of VCDs	37
3. Claim 24	39
a. Elements 24.0–24.1	39
B. Ground 2: Claim 26 is Obvious over Ellis in View of Spivey, Kellicker and Phillips	41
1. Motivation to Combine	41

2.	Claim 26.....	45
a.	Elements 26.0–26.2	45
b.	Element 26.3	47
c.	Element 26.4–26.5	48
d.	Element 26.6	52
C.	Ground 3: Claims 23 and 24 are Obvious Over Herzog in view of Spivey and Ellis	53
1.	Motivation to Combine	53
a.	Socket Server (Spivey)	54
b.	Providing Users with Sports Score and Sports Commentary Information Using Separate Communication Channels (Ellis)	57
2.	Claim 23	58
a.	Element 23.0	58
b.	Element 23.1	60
c.	Element 23.2	61
d.	Element 23.3	63
i.	First event socket corresponds to the first event information germane to the first live sporting event	65
ii.	Transmitting via a first event information communication channel between at least one first event socket of the at least one socket server and the first VCD of the first plurality of VCDs.....	65
3.	Claim 24	67
a.	Elements 24.0–24.1	67

D.	Ground 4: Claim 26 is Obvious over Herzog in View of Spivey, Ellis, Kellicker and Phillips.....	68
1.	Motivation to Combine.....	68
a.	Elements 26.0–26.2	68
b.	Element 26.3	70
c.	Element 26.4–26.5	72
d.	Element 26.6	75
IX.	SECONDARY CONSIDERATIONS OF NON–OBVIOUSNESS	76
X.	THE BOARD SHOULD NOT EXERCISE DISCRETIONARY DENIAL	77
A.	The ‘697 Has Not Been Subject to a Prior Petition by the Current Petitioner	77
B.	<i>Fintiv</i>	77
XI.	CONCLUSION	78

*All emphases and coloring added, unless otherwise noted.

TABLE OF AUTHORITIES

Page(s)

Cases

<i>Geo. M. Martin v. Alliance Machine</i> , 618 F.3d 1294 (Fed. Cir. 2010)	76
<i>Phillips v. AWH Corp.</i> , 415 F.3d 1303 (Fed. Cir. 2005)	6
<i>SportsCastr Inc. v. Genius Sports Ltd.</i> , No. 2:23-cv-04771-JRG (E.D. Tex.)	3
<i>SportsCastr Inc. v. Sportradar Group AG</i> , No. 2:23-cv-04772-JRG (E.D. Tex.)	3
<i>Videndum Production Solutions, Inc. v. Rotolight Limited</i> , IPR2023-01218, Paper 12 (April 19, 2024).....	77

Rules

35 U.S.C. §101	12
----------------------	----

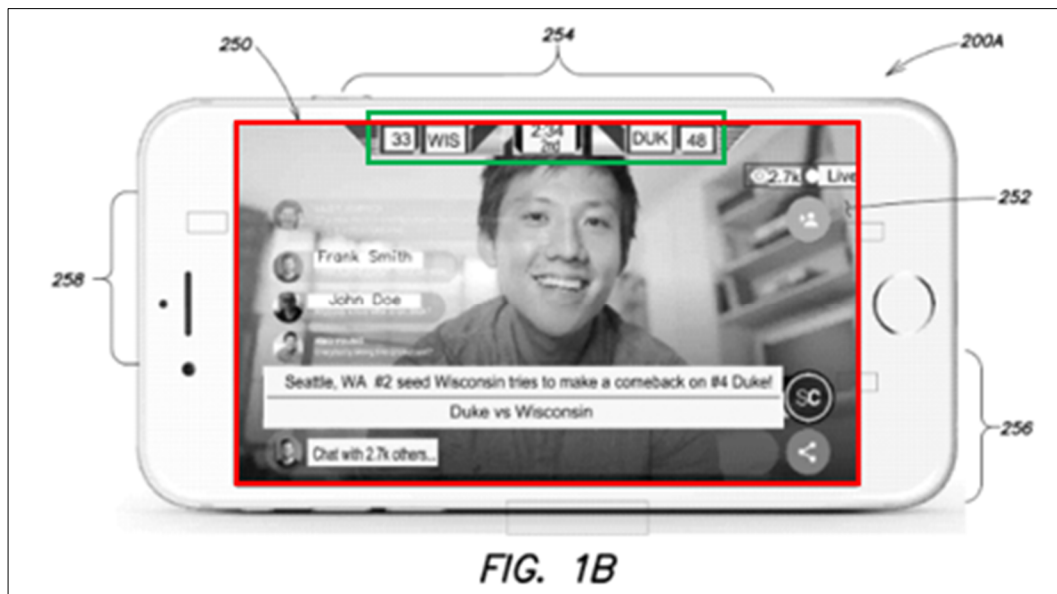
Other Authorities

37 C.F.R. §42.8(A)(1).....	3
37 C.F.R. §42.10(b)	4
37 C.F.R. §42.63(d)(2)(i)	81
37 C.F.R. §42.100(b)	6
37 C.F.R. §42.104	5

I. INTRODUCTION

U.S. Patent No. 10,425,697 (“‘697”) replays previously disclosed server components in a known way, to achieve a previously accomplished goal. Ex.1001 ¶¶63–71. The ‘697 does not claim to have invented any new component. Rather, ‘697 is directed to the known concept of providing two or more related types of content to user devices over the Internet using known real-time communication protocols. *Id.*

The ‘697 provides a video feed (i.e., a live stream of a sporting event) and at least one data feed (e.g., the sporting event’s score) to each device over separate “communication channels.” *Id.* ¶¶63–66; Ex.1006, 19:6–13, 19:64–20:11, 22:53–67. The ‘697’s final product is a **scoreboard feature** displayed on top of a **video feed**:



Ex.1006, Fig. 1B(annotated). But displaying content, such as the score or game clock, has existed since at least 1994:



Ex.1045.

The ‘697 acknowledges that adapting this feature to live streaming and internet-based content distribution systems existed well before the ‘697’s earliest possible priority date of August 5, 2016. Ex.1006, 1:62–2:18. For example, ‘697 admits that Twitch.tv, released in 2011, was a “live video streaming platform[]” that included “social networking components”—e.g., additional content such as comments and other social-media interactions. *Id.*

Although the ‘697 framed the purported invention as an “inventive server and memory storage architecture” the prior art discloses the ‘697’s server and memory storage architecture. *Id.* 2:65–3:11. As shown below, Ellis and Spivey, and Herzog, Spivey and Ellis, disclose every element of claims 23, 24 and 26 (“Challenged Claims”) and any differences in the arrangement of their disclosed architecture or data flow processes would have been obvious. Ex.1001 ¶71. Thus, for the reasons

detailed in the asserted grounds below, Petitioner respectfully requests *inter partes* review (“IPR”) of the Challenged Claims.

II. MANDATORY NOTICES UNDER 37 C.F.R. §42.8(A)(1)

A. Real Parties-in-Interest

Sportradar AG (“Petitioner”), Sportradar Group AG, Sportradar Solutions LLC and Sportradar US LLC are the real parties-in-interest.

B. Related Matters

Patent Owner (“PO”) has asserted the ‘697 against Petitioner in *SportsCastr Inc. v. Sportradar Group, AG.*, No. 2:23-cv-04772-JRG (E.D. Tex.), filed October 5, 2023. PO has also asserted ‘697 against Genius Sports Ltd. (“Genius”) in *SportsCastr Inc. v. Genius Sports Ltd.*, No. 2:23-cv-04771-JRG (E.D. Tex.). Both cases (“Texas Litigation”) were filed October 5, 2023. The ‘697 is one member of a patent family, other members of which are also asserted in the Texas Litigation, and each of which shares the same specification: U.S. Patent Nos. 10,805,687; 10,425,697; and 11,039,218. The ‘697 is the subject of IPR Petitions filed by Genius (“Genius IPRs”): IPR2024-01308 (claims 1–9) and IPR2024-01309 (claims 19–25, 27–30); the ‘687 is the subject of IPR2024-01305 (claims 1–9) and IPR2024-01307 (claims 19–25, 27–30); and the ‘218 is the subject of IPR2024-2024-01310 (claims 1–15) and IPR2024-01311 (claims 16–30). Petitioner also intends to seek joinder with the Genius IPRs. *See* Exs.1003– 1011.

The current petition is directed to claims that were previously unasserted by Patent Owner (“PO”) in the Texas Litigation, and thus were not the subject of the Genius IPRs. Exs.1014, 1048. This case may affect, or be affected by, the Texas Litigation and the Genius IPRs.

C. Lead and Back-up Counsel and Service Information

Lead Counsel	Backup Counsel
Scott W. Cummings Reg. No. 41,567 scott.cummings@bclplaw.com BRYAN CAVE LEIGHTON PAISNER LLP 1155 F St, N.W. Washington, D.C. 20004 Telephone: (202) 508–6000	Alexander Walden <i>Provisionally recognized PTAB attorney under 37 C.F.R. §42.10 (pro hac vice affidavit to follow)</i> alexander.walden@bclplaw.com Frank Fabiani Reg. No. 75,934 frank.fabiani@bclplaw.com BRYAN CAVE LEIGHTON PAISNER LLP 1290 Avenue of the Americas New York, NY 1104 Telephone: (212) 541–2000 Facsimile: (212) 541–4630

A Power of Attorney accompanies this Petition pursuant to 37 C.F.R. §42.10(b). Petitioner consents to electronic service by email at scott.cummings@bclplaw.com, alexander.walden@bclplaw.com, frank.fabiani@bclplaw.com, and SportradarIPR@bclplaw.com.

III. REQUIREMENTS FOR IPR UNDER 37 C.F.R. §42.104

A. Grounds for Standing

Petitioner certifies the ‘697 is available for IPR and Petitioner is not barred or estopped from requesting IPR on the grounds identified herein.

B. Identification of Challenge and Relief Requested

1. Claims for Which IPR Is Requested

Petitioner challenges claims 23, 24 and 26 of the ‘697¹.

2. Grounds for Challenge

Petitioner challenges the claims based on the following references:

1. U.S. Patent Publication No. 2014/0229992 to Ellis *et al.* (“Ellis”) published August 14, 2014;

2. U.S. Patent Publication No. 2016/0036910 to Spivey (“Spivey”) published February 4, 2016;

3. U.S. Patent Publication No. 2015/0163379 to Herzog (“Herzog”) published June 11, 2015;

4. U.S. Patent Publication No. 2017/0250882 to Kellicker (“Kellicker”), filed on February 29, 2016, and published in August 31, 2017; and

5. U.S. Patent Publication No. 2015/0127845 to Phillips (“Phillips”) published May 7, 2015.

¹ A claims listing for the ‘697 is in Ex.1007.

Petitioner requests IPR on the following grounds:

Ground	Basis	Claim(s)	References
1	§103	23, 24	Ellis in view of Spivey
2	§103	26	Ellis in view of Spivey, Kellicker and Phillips
3	§103	23, 24	Herzog in view of Spivey and Ellis
4	§103	26	Herzog in view of Spivey, Ellis, Kellicker and Phillips

3. Claim Construction

Claims are construed according to the principles set forth in *Phillips v. AWH Corp.*, 415 F.3d 1303 (Fed. Cir. 2005) (*en banc*). 37 C.F.R. §42.100(b). The claim terms of '697 should be given their plain and ordinary meaning as understood by a person of skill in the art ("POSA"). Ex.1001 ¶¶77–78.

The claim constructions proposed for purposes of this proceeding are not a waiver of any argument in any litigation that claim terms in the '697 are indefinite or otherwise invalid or unpatentable.

4. How the Claims Are Unpatentable

§VIII. *infra*, explains how the Challenged Claims are unpatentable.

5. Evidence Supporting This Challenge

A list of exhibits is provided at the end of the Petition. The relevance of this evidence and the specific portions supporting the challenge are provided in this Petition and in a supporting declaration of Dr. Kia Ph.D., Ex.1001. Petitioner relies on, and incorporates herein, the entirety of Ex.1001 in support of this Petition and reserves the right to rely on all of Ex.1001 in response to arguments raised by PO.

IV. TECHNOLOGY BACKGROUND: PROTOCOLS FOR REAL–TIME OR LOW–LATENCY DATA TRANSMISSION

Transmission control protocol (TCP) and internet protocol (IP), together known as “TCP/IP,” aim to ensure reliable, low–latency, end–to–end data transmission between network–connected devices. Latency, as it relates to live streaming, is the time delay between when an event occurs and when the distributed content related to that event, e.g., video, audio, or other data, is displayed on a user device. Before ‘697, developers and standards development organizations had built on top of TCP/IP or created additional protocols to facilitate real–time transport of video and other data over the Internet. Ex.1001 ¶¶31–50, 54–58.

A. HTTP Polling

HTTP is a protocol that is built on top of TCP/IP. And HTTPS is a secure version of HTTP that uses encryption and verification to protect data. HTTP polling consists of a sequence of request–response messages between a client and a server. Upon receiving a request, the server responds with a new message, if there is one,

or with an empty response if no new message is available for that client. After a short time Δ , called the polling interval, the client polls the server again to see if any new messages are available. Various real-time applications including chat, online games, and text messaging used HTTP polling in the 2016-time frame. Ex.1001 ¶51.

B. WebSocket

WebSocket was a known alternative to HTTP polling in the 2016-time frame. It is a client-server communication protocol that is bi-directional, full-duplex, and continuous. Real-time web applications commonly feature WebSocket connections to facilitate a persistent stream of data from a backend server to the end client. Once a WebSocket connection is established between a server and one or more clients, the clients receive instant updates from the server without the need to poll. Ex.1001 ¶¶52–53.

V. THE ‘697

The ‘697 issued from Application No. 16/267,887 (“‘887 Application”), which was filed February 5, 2019. Ex.1006. The ‘887 Application claimed priority to a series of Provisional Applications. Priority application No. 62/371,558, has the earliest filing date of August 5, 2016. *Id.*

A. Alleged Problem

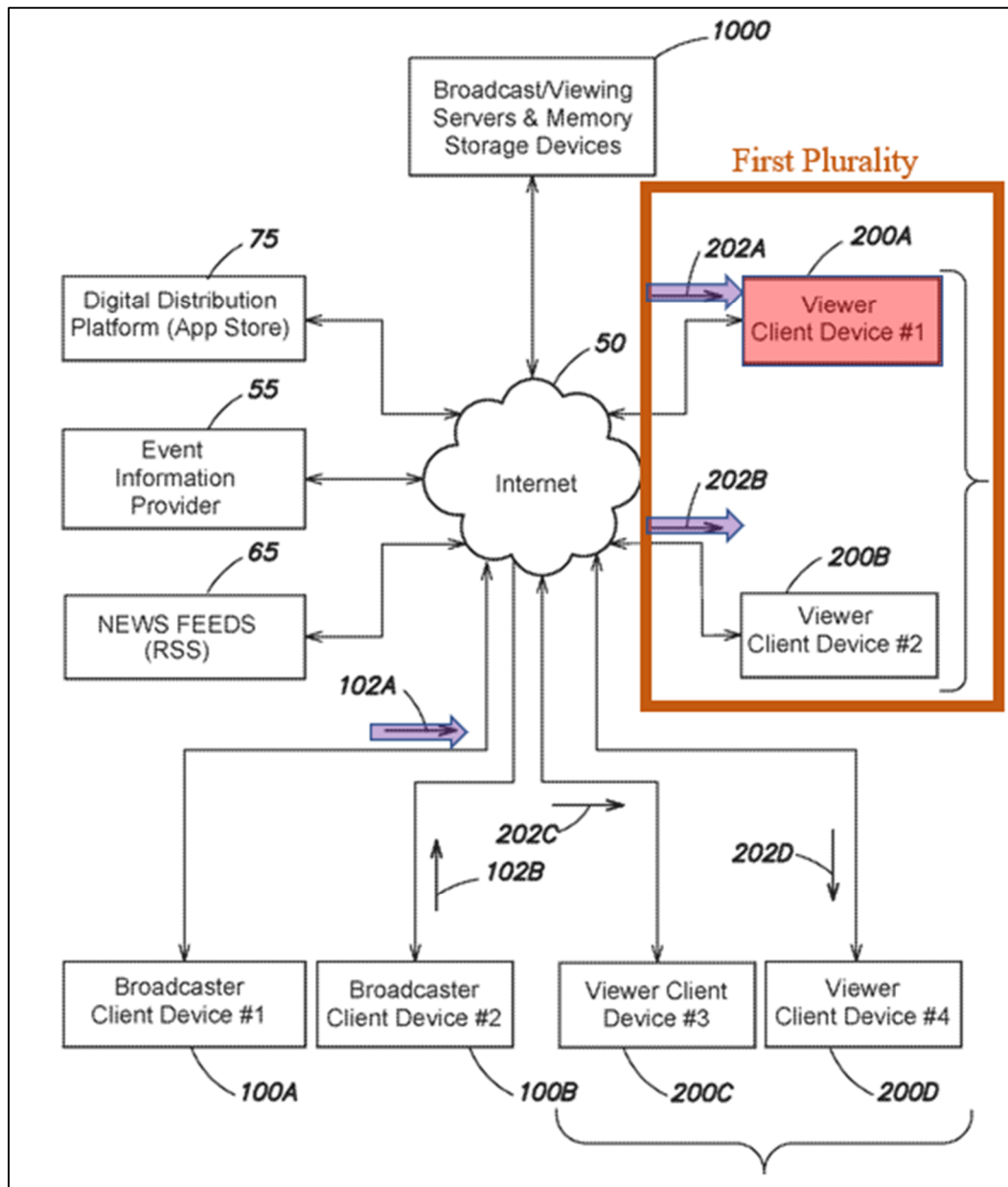
According to the ‘697, prior art techniques for streaming video to multiple users suffered from a latency problem in which there would be a delay between the

generation of a video and the time in which a user would receive the video stream.

Ex.1006, 2:22–47, 5:19–28; Ex.1001 ¶¶60–62.

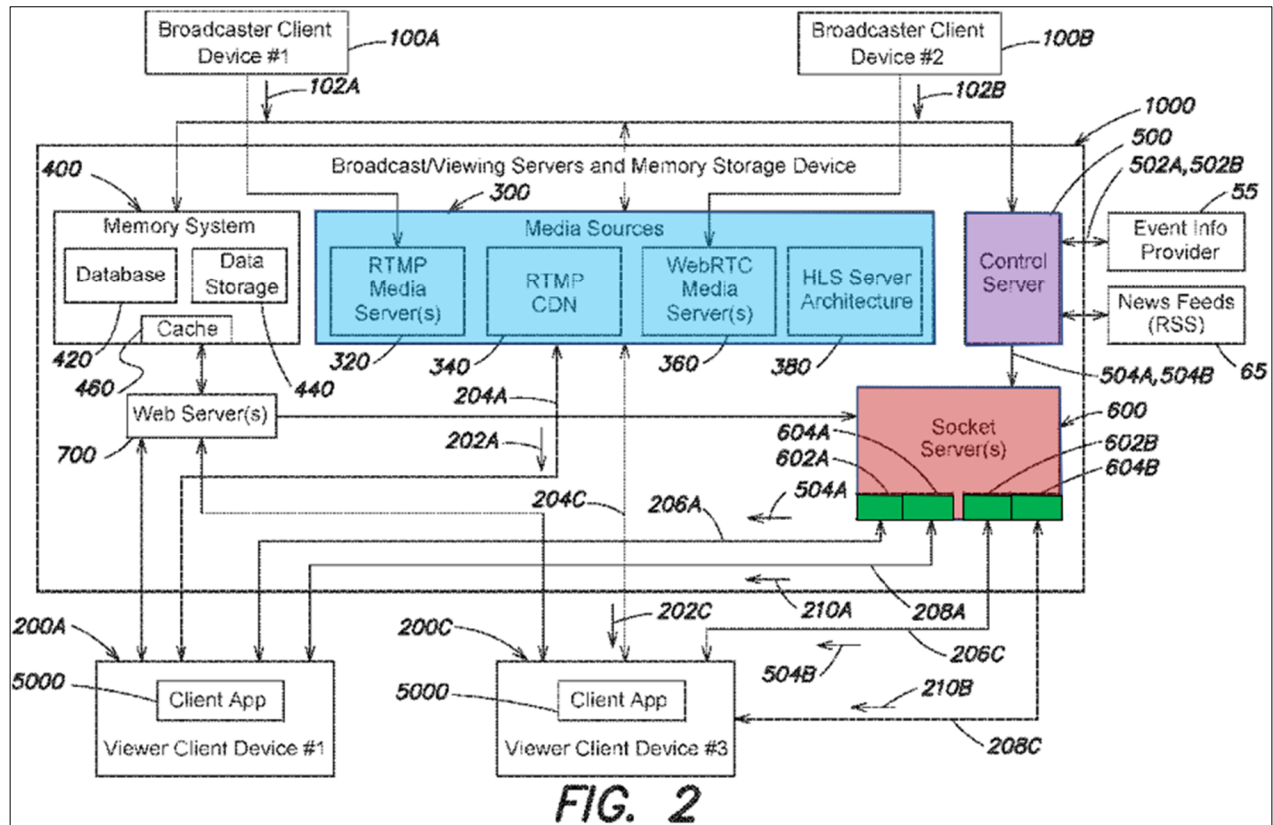
B. Alleged Invention

The ‘697 purports to solve this latency issue. Ex.1006, 5:19–28. The ‘697 is directed to a “server and memory storage architecture” for obtaining and providing to viewers, (1) an “audio and/or visual feed” including “commentary associated with [a] live event” and (2) a “separate event data feed including real time information associated with the ... live event.” *Id.* 2:65–3:11; *see also Id.* 7:7–28. A block diagram of the system is shown in Figure 1A:



Id. 11:9–19, Fig. 1A(annotated). Figure 1A depicts providing “a first viewer client device (200A)” of “a first plurality of viewer client devices (200A, 200B)” individual “copies (202A, 202B) of a first broadcaster’s live stream of digital content (102A).” *Id.* 9:16–41.

Figure 2 shows additional system details. For instance, Figure 2 depicts viewer client devices (“VCDs”), media sources, a control server, a socket server, sockets, an event information provider, and a memory system:



Id. Fig. 2(annotated).

The ‘697 describes media sources as generic “computer servers.” *Id.* 19:7–21, 55:60–6:3. The media sources’ purpose is to “receive a live stream of video–based commentary from a given broadcaster client device and provide copies of the live stream of video–based commentary to one or more [VCDs].” *Id.*

The ‘697 describes a control server as a generic server that retrieves “event information” from an “event information provider” to pass to the socket server,

discussed below. *Id.* 19:41–45, 21:37–48. The event information is “germane to” a particular “live sporting event.” *Id.* 19:27–45, 21:37–48.

The ‘697 describes socket server(s) as generic servers. *Id.* 55:60–6:3. The socket server’s purpose is to “facilitate communication” of “synchronized event information retrieved by the control server” between the control server and the VCDs. *Id.* 19:64–20:5. The socket server uses “one or more sockets ... dedicated to the particular event” allowing VCDs “to establish an event information channel with the socket server.” *Id.* 20:5–11. The socket server then shares event information in “event messages” via the event information channel “in a synchronized manner” to all viewers “following the particular event.” *Id.* The socket server may facilitate communication of “chat messages” in a similar manner. *Id.* 20:12–44.

Although the ‘697 claims that the foregoing server architecture and its functionality are “inventive,” as explained in more detail herein, the prior art discloses or renders obvious every element of the Challenged Claims. Ex.1001 ¶¶63–71.

C. Prosecution History

The ‘697 Examiner issued a single office action on the merits, rejecting the claims only based on 35 U.S.C. §101. Ex.1008 381–385. A notice of allowance issued on June 26, 2019. *Id.* 436–43. Although the Examiner provided a statement

of reasons for allowance, the rationale is conclusory. None of the prior art relied upon in this Petition was applied during prosecution. *Id.*

VI. LEVEL OF ORDINARY SKILL IN THE ART

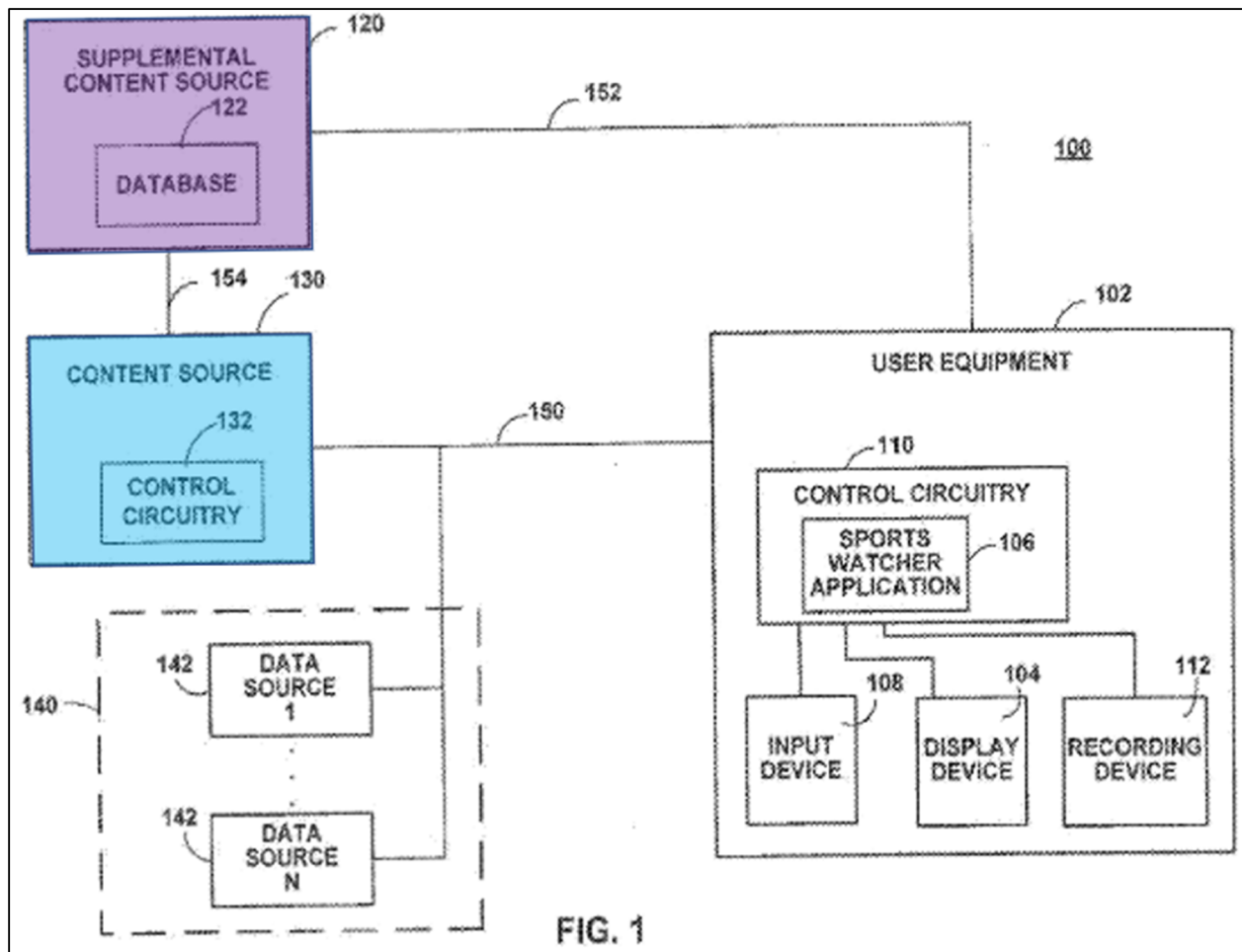
A POSA in the field of live streaming technology at the time of the alleged invention, August 5, 2016, would have had at least a bachelor's degree, or equivalent, in electrical engineering, computer engineering, computer science, or a related field, and at least two years of experience in video and content processing and communications systems and methods. Ex.1001 ¶¶72–76. Additional education or experience in related fields could compensate for deficits in the above qualifications. *Id.*

VII. OVERVIEW OF THE PRIMARY REFERENCES

A. Ellis

Ellis identified the need for current delivery systems to provide “a custom set of features and functionality for specific types of media content, such as sporting events” Ex.1016 ¶4. Ellis explained that the current systems “fail to take full advantage of today’s high–end user equipment devices.” *Id.* ¶5. To solve the above problems, Ellis discloses broadcasting systems and methods for “enhanced sports–related media content access and display.” *Id.* Abstract.

Figure 1 is an illustrative media system:



Id. ¶15, Fig.1(annotated). As depicted, Ellis’s system contains a **content source**, a **supplemental content source** (“SCS”), data sources, user equipment, and a sports watcher application. *Id.* Fig. 1.

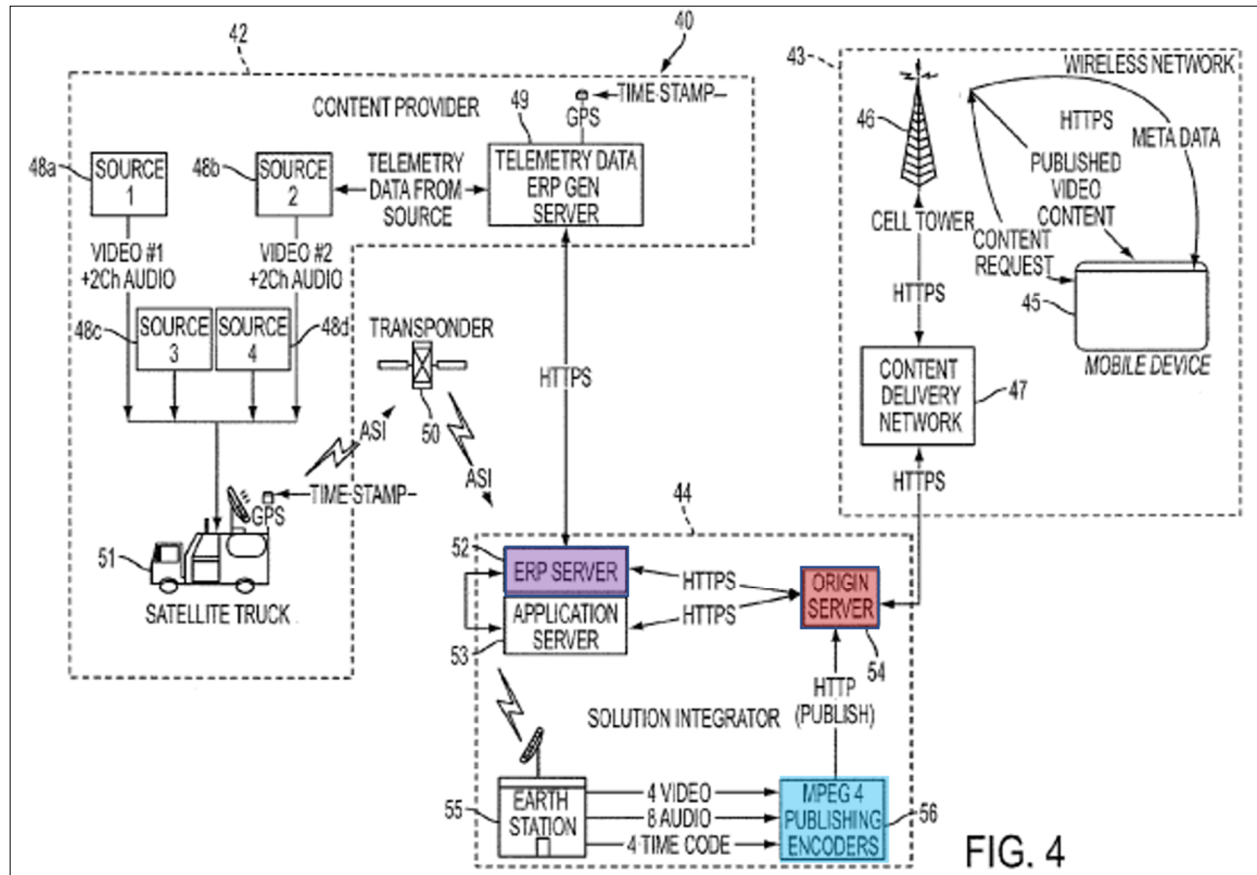
While the system provides the video feed over communication path 150, the SCS further provides real–time information over a separate communication path 152. *Id.* ¶¶39–42, Fig. 1. Importantly, “in some embodiments, this sports–related information is provided to user equipment 102 as a plurality of real–time data feeds” where each “data feed may be individually accessed or subscribed to” *Id.* ¶41.

The SCS “real-time” communication path 152 is protocol agnostic. Ex.1001 ¶¶87–92.

B. Herzog

Herzog explains that it “may be desirable to provide a mobile device, such as a tablet” with a “video that is streamed live from the Internet to a client player using” HLS while also providing “additional information related to the HLS video,” such as meta-data. Ex.1015 ¶3.

Herzog uses an “Indianapolis Car Race” to explain how its system transmits live “video, audio, and meta-data ... from a sports venue via different communication paths,” to a client device. *Id.* ¶16. Figure 4, shown below, depicts an illustrative streaming network:



Id. Fig. 4(annotated). Figure 4 shows MPEG 4 Publishing Encoders, an ERP server, and an origin server. *Id.* ¶¶43, 45. Content provider 42 provides “four video/audio streams” from “four different imaging sources” to MPEG 4 Publishing encoders who then stream the “video and audio data ... to the origin server 54 in an HTTP communications format.” *Id.* ¶¶46, 49.

At the same time, the ERP server “receives the meta-data from ERP generator server 49 in HTTPS.” *Id.* ¶50. The meta-data “provides additional information related to the HLS video” and in the context of a car race, such as the example shown in Figure 4, “may include varying parameters regarding the status of each car and driver in the race” including “lap number, car speed, ... etc.” *Id.* ¶¶3, 45. The user

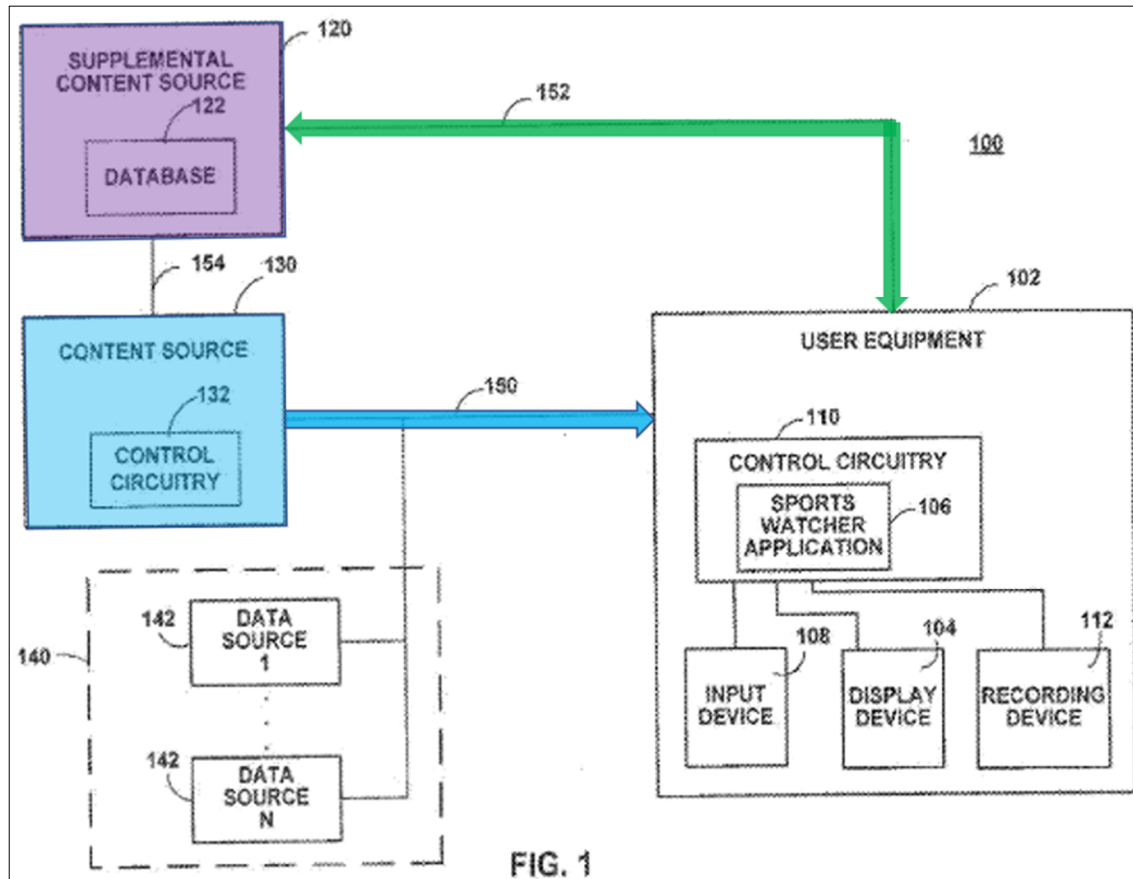
then receives a list of available meta-data and can “select specific meta-data associated with the audio and video data.” *Id.* ¶34. As shown above in wireless network 43, the meta-data is then provided through the origin server to the user device “through one or more networks” using HTTPS. *Id.* ¶¶3, 27, 51. At the user device, “all the data may be displayed ... in correct time sync” because the “video and audio data and the meta-data are both time stamped with consistent (e.g., GPS) timing protocols.” *Id.* ¶18; Ex.1001 ¶¶79–86.

VIII. EACH OF THE CHALLENGED CLAIMS IS UNPATENTABLE

A. Ground 1: Claims 23 and 24 are Obvious Over Ellis in view of Spivey

1. Motivation to Combine

Ellis discloses the ‘697’s broad functionality—providing (1) video and (2) data to a user over separate connections:

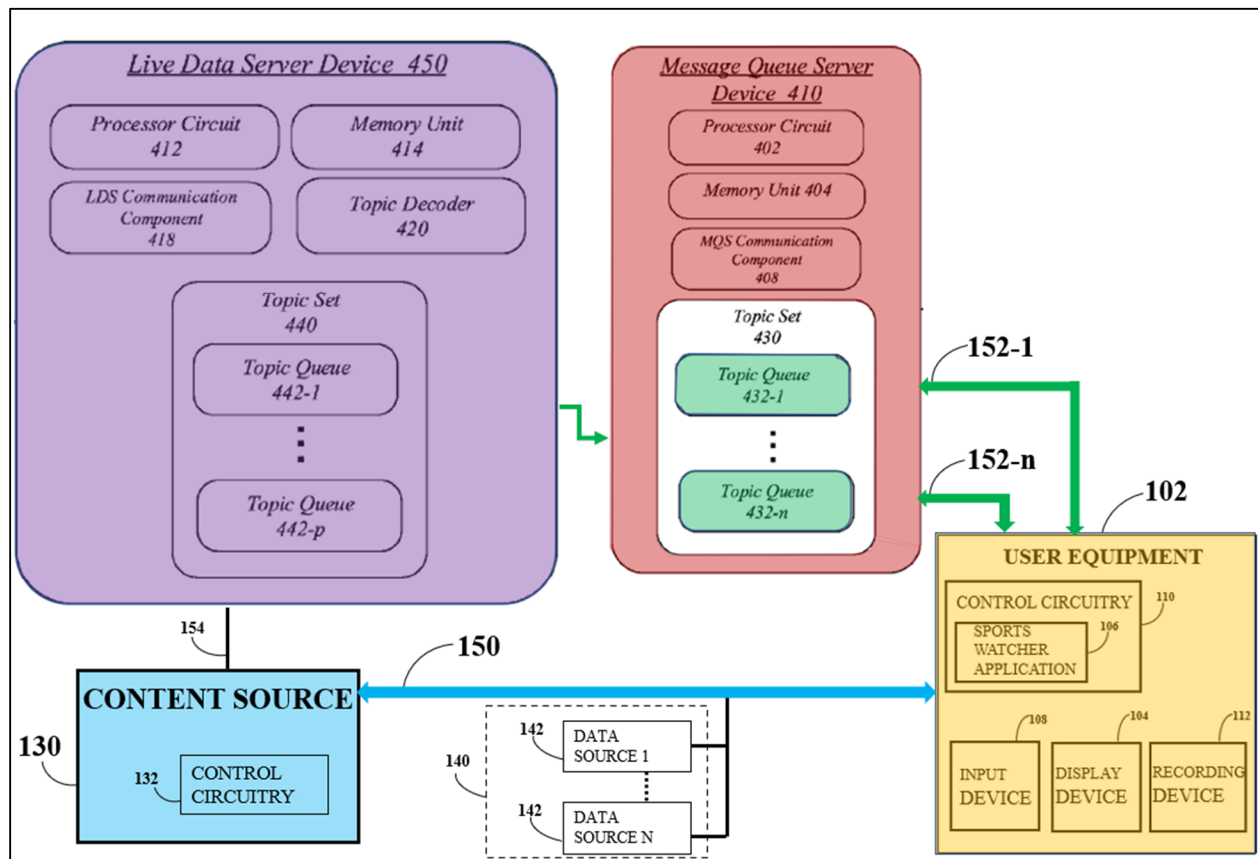


Ex.1016, Fig. 1(annotated); Ex.1001 ¶¶105–06. Spivey provides details relating to backend server architecture supporting the described functionality and confirms that it would have been obvious to use a websocket connection in lieu of HTTP to transmit Ellis’s SCS information. *Id.* ¶¶106.

Spivey explains that some live–event data delivery systems, such as Ellis’s, suffer from latency issues. *Id.* So, Spivey teaches a Live Data Server Device (“LDSD”) and Message Queue Server Device (“MQSD”) that could provide live–event data to any number of customers in milliseconds. Ex.1017 ¶¶27–28; Ex.1001 ¶106. A POSA, looking to improve Ellis’s latency issues, would have implemented

a system such as Spivey, and would have recognized that combining Spivey's teachings with Ellis would improve Ellis's latency issues. Ex.1001 ¶¶107–09.

A POSA would have implemented Spivey into Ellis's **data** delivery system by replacing Ellis's **SCS** with Spivey's **LDS** and adding the **MQSD** to achieve a predictable result of decreased latency data delivery using a websocket connection in lieu of HTTPS. See Ex.1016 ¶¶39, 41, Fig. 1; Ex.1017 ¶¶53–57, Fig. 4; Ex.1001 ¶110. An example of this combination is illustrated here in an annotated, combined diagram of Ellis Figure 1 and Spivey Figure 4:



Ex.1016, Fig. 1 (recreated); Ex.1017, Fig. 4 (recreated) (combined as explained above); Ex.1001 ¶110. The combined system would maintain Ellis's video content delivery system and upgrade the data delivery system to reduce latency. Ex.1001 ¶¶110–11.

A POSA would have been motivated to combine Ellis and Spivey. First, both relate to the same subject matter—content delivery for real-time events. *See* Ex.1016 Abstract; Ex.1017 Abstract. Both describe methods and/or systems for streaming video and/or event data to one or more users simultaneously. *See, e.g.,* Ex.1016 ¶156; Ex.1017 Abstract; Ex.1001 ¶112. Second, the references solve the same or similar problems, namely, quickly providing video and/or event data to multiple users over the Internet. *See, e.g.,* Ex.1016 ¶¶39–40, 176; Ex.1017 ¶27. Ex.1001 ¶112.

A POSA would have successfully combined Ellis and Spivey. Ellis and Spivey disclose complementary means of achieving the common goal of synchronizing multiple forms of data, including video/audio and event data and transmitting that data to one or more users. For example, Ellis discloses systems for transmitting live video and related event data to viewers over the Internet. *See, e.g.,* Ex.1016 ¶¶40–41, 45, 74; Ex.1001 ¶112. Ellis also teaches that multiple users may view one or more different live events and receive related event information and

discloses various systems for transmitting such information. *See, e.g.,* Ex.1016 ¶¶39–41, 45, 74.

A POSA would have been motivated to combine Ellis with Spivey because Spivey provides a complementary solution describing the details of the backend architecture for transmitting event data to multiple users. Ex.1017, Abstract; Ex.1001 ¶114. For example, to implement Ellis’s system and reduce its latency, a POSA would have had a reasonable expectation of success in using Spivey’s system of “topic queues and open duplex communication connections” employing websockets and a “message queue server device” to broadcast “data from a live event (e.g., news event, a gaming event, sporting event, etc.)” to multiple users “in real–time.” Ex.1017 ¶¶27, 58.

Implementing certain portions of Spivey’s system into Ellis’s system would have been obvious because doing so would have required only the simple substitution of known elements to obtain predictable results. Ex.1001 ¶113. For example, Ellis discloses compatible and customizable hardware and software components to act as a source for event information to be transmitted to users. *See, e.g.,* Ex.1016 ¶47(“Content source 130, supplemental content source 120, and data sources 140 are shown in Fig. 1 as separate elements” but “[i]n practice, their functionality may be combined into a single mechanism and provided from a single system at a single facility, or their functionality may be provided by multiple systems

at multiple facilities.”). Predictably, therefore, a POSA would have simply substituted portions of Ellis’s SCS with Spivey’s LDS and MQSD. For example, it would have been obvious to substitute any of Ellis’s elements with any combination of Spivey’s LDS, MQSD, and/or Live Data Source(s), which are similarly involved in transmitting live event data to users and may be configured in any manner. *See* Ex.1017 ¶34; Ex.1001 ¶¶113.

2. Claim 23

a. Element 23.0

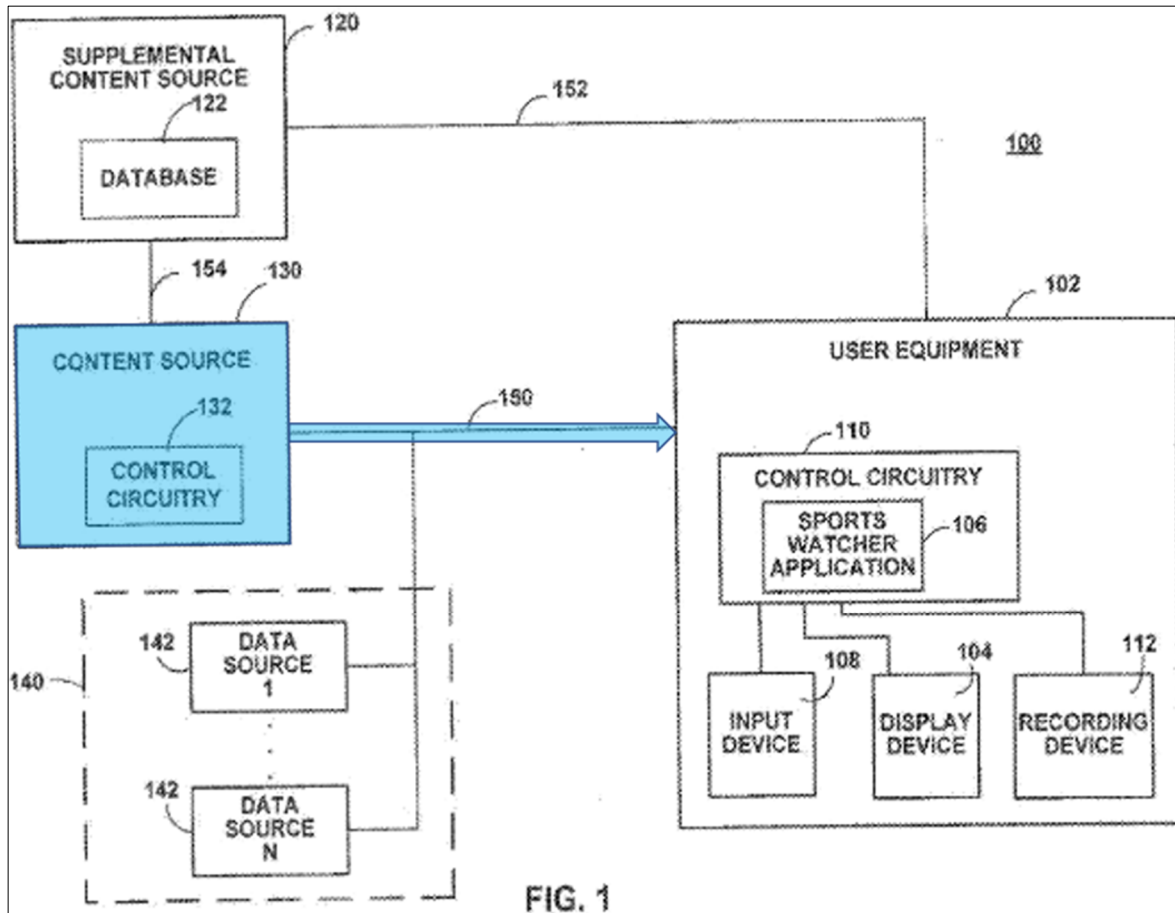
To the extent the preamble is limiting, Ellis discloses this element. Ex.1001 ¶¶317–18; Ex.1016 Abstract. Ellis discloses providing a first copy of a first broadcaster’s live stream of digital content including first live sporting event video–based commentary from a first broadcaster client device to a first viewer client device (VCD) by providing “enhanced information access and viewing functionality for sports–related programming,” which includes “a sporting event, sports news, or sports commentary.” *Id.* ¶¶8, 11, 38–40; Ex.1001 ¶318.

Ellis discloses providing the “media content” from those sporting events “to user equipment 102.” Ex.1016 ¶¶38–40; Fig.1. Ellis defines “sporting event” as encompassing “online sports commentary.” *Id.* ¶38. Further, media content includes broadcast programming “and any other content capable of being displayed by, presented to, ... or interacted with, using user equipment 102.” *Id.* Furthermore, a

POSA would have recognized that broadcasts of live events would obviously, if not implicitly, also include video-based commentary. Ex.1016 ¶38; Ex.1001 ¶319.

b. Element 23.1

Ellis discloses **content sources**, which are “any suitable source of media content” such as a “Internet or network media/web server, or any other suitable facility or system for originating or distributing passive or interactive content to user equipment.” Ex.1016 ¶¶39–40. Ellis’s content sources serve to receive the first/second broadcaster’s live stream of digital content and provide video content to users via communications paths. *See Id.* ¶¶39–41, 45, 47; Ex.1001 ¶320–21. Figure 1 shows a content source providing video content to users over communication path 150:



Ex.1016 ¶¶39–41, Fig. 1(annotated). Ellis also discloses multiple content sources, i.e., a plurality. *Id.* ¶39, 74(“[T]here may be numerous instances of content source 130 ...”). As shown in Fig. 1 above, Ellis also discloses that users can be connected with a plurality of data sources 140, 142. *Id.* Moreover, as explained below, Ellis’s content sources perform the same functions as the media sources as recited in 23.1. Ex.1001 ¶321.

As explained above, Ellis discloses providing individual copies of live streams to client devices. Ex.1001 ¶322. Ellis further discloses providing digital content or video content over a video Internet communication channel. As an example of an

“Internet communication channel,” the ‘697 describes a “video” channel between a media server and a client device to convey video-based commentary. Ex.1006, 5:36–49. Ellis discloses providing video from a content source to user equipment, over a communication path. Ex.1016 ¶¶39, 74. The content source can be an “internet or network media/web server” and the user equipment “may include computer equipment.” *Id.* ¶¶40, 48. The communication path can be “for example, a satellite path, a fiber-optic path, a cable path, or any other suitable wired or wireless path.” *Id.* ¶45. Ellis describes the communication path as a unique internet connection between the content source and the user equipment that provides a video live stream: “[f]or IPTV, tuned content may include content to be delivered on a particular network address (e.g., multicast address) on which an IP socket is currently joined.” *Id.* ¶74. Thus, Ellis discloses that each user is provided with a copy from the content source over a first video internet communication path that is unique to each user device. *Id.*; Ex.1001 ¶323.

c. Element 23.2

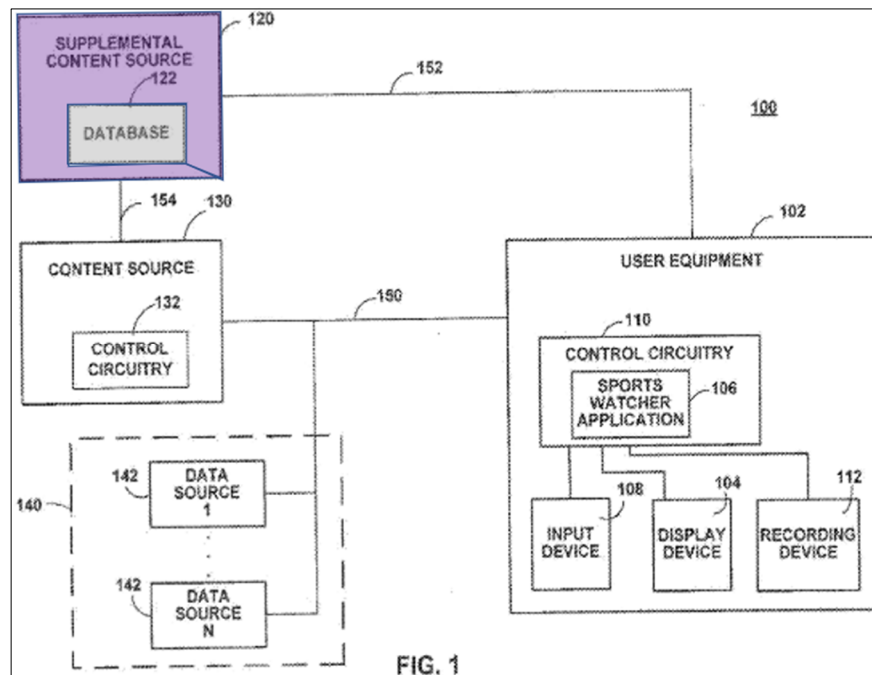
Ellis and Spivey both disclose this element. Ex.1001 ¶324. The ‘697 explains that the **control server** is a generic server. Ex.1006, 55:60–6:3. The control server retrieves “event information” from an “event information provider.” *Id.* 21:37–42. The control server then transmits the event information to the socket server. *Id.* 19:41–45, 21:37–48. The event information is “germane to” a particular “live

sporting event,” e.g., score information of one of the live streams of digital content. *Id.* 21:37–42, 19:27–37; Ex.1001 ¶325.

Ellis’s Supplemental Content Source (SCS) comprises a control server. Ex.1016 ¶45(“supplemental content source 120 may be configured to transmit signals to user’s equipment 102 over any suitable communication paths . . .”); Ex.1001 ¶326. The SCS can be “any suitable content source configured to provide real–time sports scores and game summary information” including a “content manager, content aggregator, or intermediate content provider.” *Id.*; Ex.1016 ¶¶42, 142. To the extent that the frequency with which the information is retrieved is considered as limiting the claimed system, the SCS periodically retrieves, via the internet and from an event information provider, first event information germane to the first live sporting event, including first score information for the first live sporting event because the SCS “provide[s] real–time sports scores and game summary information . . .” *Id.* ¶¶41, 72, 117; Ex.1001 ¶326. A POSA would have understood that this information is retrieved via the internet. *Id.* ¶¶37–38, 40, 142. Furthermore, a POSA would have known that is how real–time data was transferred to users at the time of Ellis and Spivey—including score updates and wagering information. Ex.1016 ¶¶42, 142. Additionally, it would have been obvious to a POSA to retrieve this information via the Internet as a well–known convenient and reliable means of acquiring the supplemental content. Ex.1001 ¶326.

While a “periodic” time interval is not specifically mentioned in Ellis, a POSA would have understood that any system that provides real-time sports statistics, such as Ellis, would update the statistics on a “periodic” basis. Ex.1001 ¶327. This is especially true given that no specific time interval is specified by the claim. Alternatively, it would have been obvious to a POSA to implement periodic retrieval of the event information to provide the user with up-to-date information thereby enhancing the overall viewing experience, whilst limiting queries to the system and load on the components. *Id.*

Further, as depicted in Figure 1, the SCS may store the received “sports-related information ... in database 122 and provided to content source 130 or directly to user equipment 120 over communication paths 154 and 152”:



Ex.1016 ¶42, Fig. 1(annotated); Ex.1001 ¶328.

To the extent Ellis does not expressly disclose or render obvious this element, Spivey’s LDS, as shown in Figure 4, also contains the same basic components as the ‘697’s control server:

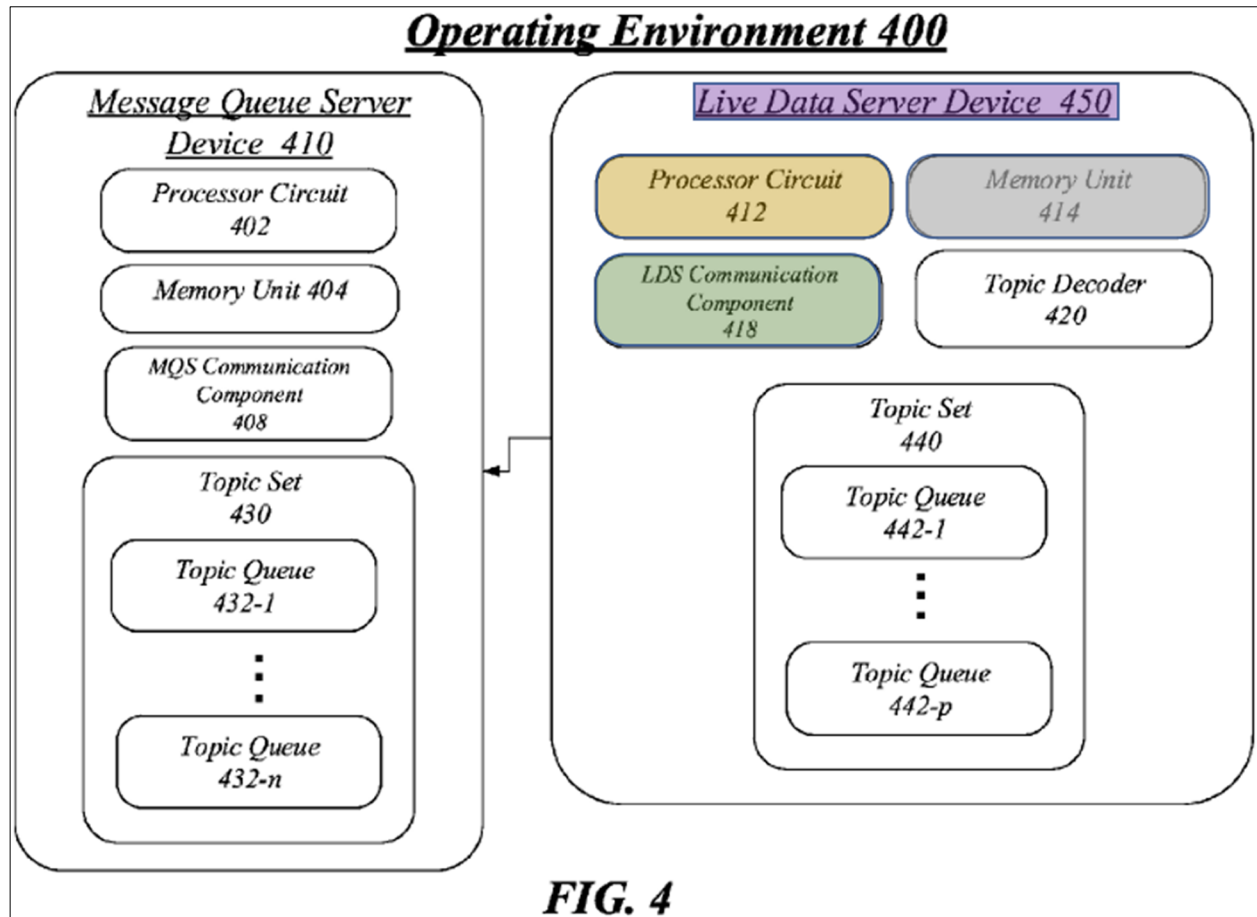


FIG. 4

Ex.1017, Fig. 4(annotated). The LDS is a generic server comprising one or more processors, memory devices, and communication interfaces. *Id.*; Ex.1001 ¶¶329–30. Moreover, the LDS performs the same functions as the ‘697’s control server. *Id.* ¶330. The LDS contains an LDS communication component that allows LDS “to receive live data and transmit the data payloads to MQS device 410.” Ex.1017 ¶55. The live data received by the LDS includes “sports scores and breaking news” and

when related to baseball can include “a score, a play, an error, ... and so forth.” *See Id.* ¶¶27, 48, 66 (“A data payload may include ... information about the live event in progress.”); Ex.1001 ¶330. The LDSD retrieves the first event information germane to a first live sporting event including the first score information for the first live sporting event via the internet and from an event information provider because the live data sources are associated with the live event, and servers in Spivey communicate over the internet. Ex.1017 ¶¶35, 37, 178; Ex.1001 ¶330. Thus, a POSA would have understood that Ellis and Spivey both disclose this element. Ex.1001 ¶330. To the extent the SCS does not meet this element, a POSA would have replaced it with Spivey’s LDSD and MQSD to decrease latency. *Id.*; §VIII.A.1, §VIII.A.2.b.

d. Element 23.3

Ellis in view of Spivey discloses a **socket server** communicatively coupled to the **control server** (which, as discussed above, is disclosed by Ellis’s **SCS** and Spivey’s **LDS**). The ‘697 describes the socket server as a generic server. *See* Ex.1006 55:60–6:3; Ex.1001 ¶331–33. The purpose of the socket server is to “facilitate communication” of “synchronized event information retrieved by the control server” associated with an event between the control server and the VCDs. Ex.1006 19:64–20:5. The socket server does so by using “one or more **sockets** . . . dedicated to the particular event” allowing VCDs “to establish an event information

channel with the socket server.” *Id.* 20:5–11. The socket server then receives event information transferred from the control server and shares the event information in “event messages” via the event information channel “in a synchronized manner” to all viewers “following the particular event.” *Id.*

Ellis discloses providing event information over a separate connection to user equipment. Ex.1016 ¶¶39–41, Fig. 1. A POSA would have looked to Spivey to add additional backend architecture to reduce latency in Ellis’s real-time data delivery system. Like the ‘697’s socket server, Spivey’s MQSD is a generic server that contains processors, memory, and communication components. Ex.1017 ¶¶35, 53, Figs. 3, 4. Moreover, Spivey’s MQSD carries out the same purpose as the ‘697’s socket server. Ex.1001 ¶334–35. The MQSD “receive[s]” and “process[es] live data to provide live data” to “the subscribed connected client devices substantially simultaneously.” Ex.1017 ¶¶96, 45(“[MQS] device 310 may be a representative example of message queue server 120”), 54(MQSD is “substantially similar” to “MQS device 310”). To accomplish this, the MQSD receives live event data from the LDSD, which just “receive[d] live data”:

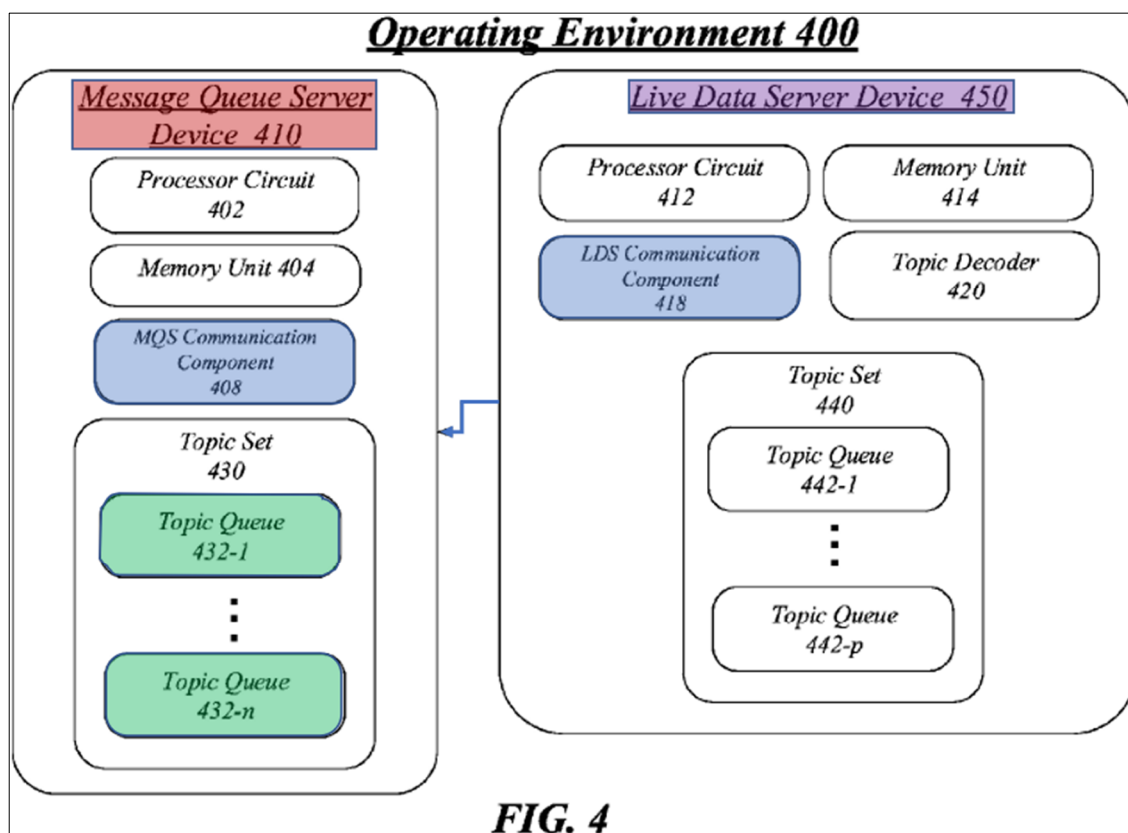


FIG. 4

Id. ¶54, 55, Fig. 4(annotated); Ex.1001 ¶335. The MQSD is communicatively coupled to the LDS through the MQS communication component and the LDS communication component. Ex.1017 ¶46(communicaion component 308 transmits and receives data), 54(communicaion component 408 is analogous to 308), ¶55(“LDS communication component 418 ... allows [LDS] to ... transmit the data payloads to [MQSD].”).²

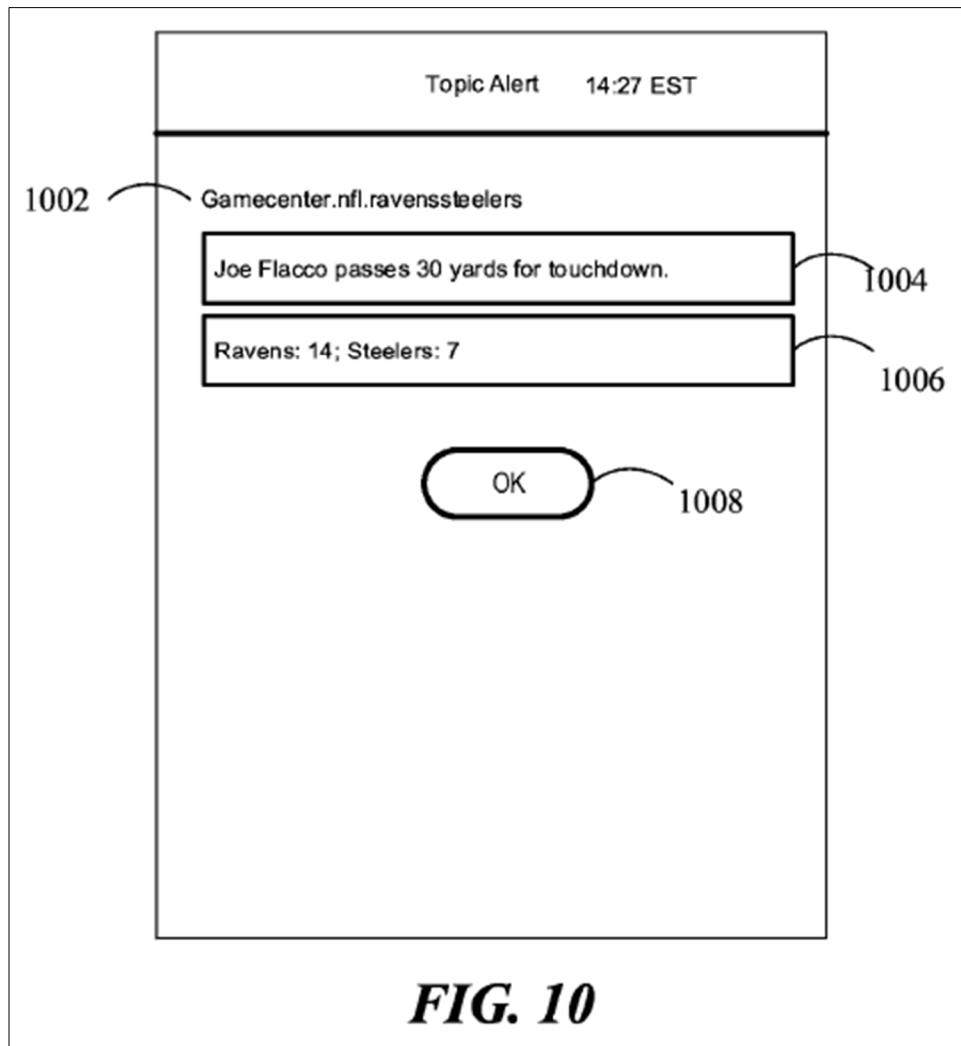
² Ellis also discloses a system where multiple servers provide the event information. Ex.1016 ¶47 (SCS’s functionality “may be provided by multiple systems at multiple facilities”).

The MQSD organizes the received live data “into one or more of a plurality of **topic queues**” dedicated to storing information about “a specific game.” *Id.* ¶¶47, 52. After sorting the data into topic queues, the MQSD shares the information “substantially simultaneously” to “the subscribed connected client devices.” *See Id.* ¶96. Thus, a POSA would have understood Spivey’s MQSD as disclosing a socket server communicatively coupled to a control server. Ex.1001 ¶335.

Alternatively, Spivey teaches that latency associated with the broadcast of event data, such as sports scores, to clients can be reduced by “using topic queues and open duplex communication connections.” Ex.1017 ¶¶27–28. Here, “topics” include or relate to specific sporting events. *Id.* ¶85 (“topic 910–1 corresponding to a National Football League (NFL) game between the Baltimore Ravens and the Pittsburgh Steelers.”). As an example of such “open duplex communication connections,” Spivey discloses that communications between a server and a client can be a websocket connection. *Id.* claim 3 (“the communication component comprises **a websocket server** component and listens for **websocket connection** from the client device”); *see Id.* ¶¶43, 61, 74, 92, claim 8. Thus, it would also have been obvious to a POSA in view of Spivey’s teachings to have established communications between the MQSD server for data transmissions around a specific topic or sporting event by utilizing a websocket server component (the MQSD) and a websocket connection with the client device. Ex.1001 ¶336. As explained above,

a POSA would have recognized that doing so would further the express goals of Spivey with respect to reduction of latency in such transmissions. Ex.1001 ¶336

Spivey's LDSD (control server) "receive[s] live data and transmit[s] the data payloads to [MQSD (socket server)]." Ex.1017 ¶55. Data payloads may include "a score, a player name, a play, and so forth."



Id. ¶¶85, 117, Figs. 9–10. Thus, a POSA would have understood that Spivey discloses a socket server receiving at least the first score information transmitted from the control server. Ex.1001 ¶337.

- i. First event socket corresponds to the first event information germane to the first live sporting event

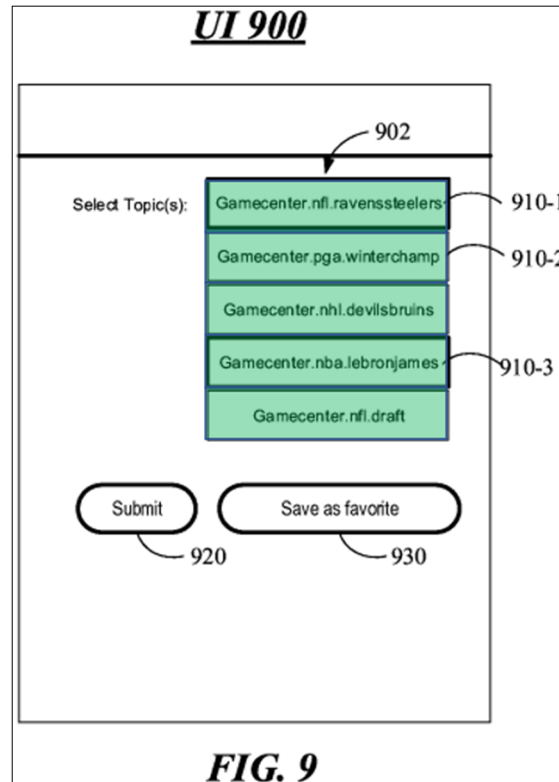
It is noted that, while Spivey discusses that a connection between the client device and the system “*may include*” a live data access server (“LDAS”) according to certain embodiments, Spivey also discloses and teaches that the message queue server device may be “accessible to clients over the network directly.” Ex.1017 ¶34(client and MQS may be communicatively coupled and “the system 100 may include...fewer elements in alternate topologies as desired for a given implementation”), 38, 84. For example, “MQS communication component 308 ... may push data from topic queues 332–1, 332–n to the subscribers of the topics corresponding to those topic queues.” *Id.* ¶46. Thus, a POSA would have understood Spivey discloses and teaches forming a direct connection between the client devices and MQSD. A POSA would have used this implementation because reducing the amount of server transfers would further reduce latency, which is Spivey’s main purpose. Furthermore, it was a well-known alternative to a POSA to implement the functionality of multiple servers as a single server. Ex.1001 ¶338.

A first event socket corresponding to the first event information germane to the first live sporting event is simply a socket where information about one sporting event is provided to VCDs. Ex.1001 ¶339. A POSA would have understood a “socket” to be a uniquely addressable communication endpoint on a server where

first event information is provided to a VCD. Ex.1006 20:45–61, 21:53–22:6; Ex.1029 14–20; Ex1030; Ex.1001 ¶¶52–53.

Spivey’s **topic queues** are “where data payloads are placed” that, for example, “relate to a specific game between two NFL teams.” Ex.1001 ¶340; Ex.1017 ¶¶51–52. The data payloads, “[w]hen the live event is a sporting event,” include, “for example, a score, a play, ..., and so forth.” *Id.* ¶48. Topic queues have a corresponding “topic label” which is used “to identify a topic queue.” *Id.* ¶49. “A topic label may be, for example” a “text string such as ‘gamecenter.nfl.ravenssteelers.’” *Id.* ¶66.

Client devices use these topic label text strings to connect and subscribe to MQSD’s topic queues. First, client devices “request a live data connection” with the MQSD by “select[ing] or enter[ing] the address or connection information to a particular” MQSD. *Id.* ¶¶81–82. This “address may resemble a uniform resource locator (URL).” *Id.* ¶82. Second, as shown in Figure 9, client devices can use the topic labels to select “the topics available from [MQSD] in a topic selection list” and “subscribe to” those topic queues:



Id. ¶¶83–84, Fig. 9(annotated). Third, “the data payload” may be pushed to the subscribed connected devices that requested that topic. *Id.* ¶96. Thus, a POSA would have understood that Spivey’s topic queues disclose first event sockets corresponding to the first event information germane to the first live sporting event. Ex.1001 ¶341.

Alternatively, it would also have been obvious to a POSA in view of these teachings of Spivey to have established communications between the MQSD server for data transmissions around a specific topic or sporting event by utilizing a websocket server component (the MQSD) and a websocket connection with the client device for the same reasons explained above in §VIII.A.1. Ex.1001 ¶342.

- ii. Transmitting the first score information to at least the first VCD

Ellis discloses at least a first VCD. *See* §§VIII.A.2.a–b. Both Ellis and Spivey further disclose transmitting the first score information to the first/second VCD of the first plurality of VCDs. For example, in Spivey, the MQSD “may push data from topic queues ... to the subscribers of the topics corresponding to those topic queues.” Ex.1017 ¶46; §VIII.A.2.d, §VIII.A.2.d.i. Similarly, in Ellis, SCS may be configured “to provide real–time sports scores and game summary information, ... to user equipment 102.” Ex.1016 ¶41; Ex.1001 ¶343.

- iii. Transmitting via a first event information communication channel between at least one first event socket of the at least one socket server and the first VCD of the first plurality of VCDs³

Ellis discloses at least a first VCD. *See* §§VIII.A.2.a–b. To the extent the claim could also be interpreted as reciting “*a first plurality of viewer client devices*,” Ellis discloses multiple broadcasters providing copies of each broadcaster’s live stream of digital content including live sporting event video–based commentary to a plurality of user devices by providing a broadcast system where any “number of users” may receive signals from a content source. Ex.1016 ¶39. The signals are

³ It is noted that there is no antecedent basis for the recited “*the first plurality of viewer devices*.” The claim is clearly directed to “a first viewer client device.”

individual copies of the same live stream of digital content. *Id.* ¶¶40, 62, Fig. 2B; Ex.1001 ¶344. For example, Ellis’s system broadcasts digital streams from the content source, which can be an “[i]nternet or network media/web server.” Ex.1016 ¶40. Users can select which content to receive, as shown in Figure 2. *Id.* ¶62, 74. Ellis then sends an individual copy of the live event to the users over a communication path. *Id.* ¶45, 74. The communication path can be any “suitable wired or wireless path.” *Id.* “For IPTV,” content may be “delivered on a particular network address (e.g., multicast address) on which an IP socket is currently joined.” *Id.* ¶74. A POSA would have understood the distribution system of Ellis provides a plurality of copies of multiple broadcaster’s live streams of digital content to a plurality of viewers from a plurality of broadcaster devices. Ex.1001 ¶344.

Ellis in view of Spivey discloses providing first event information over a separate communication channel. Spivey’s entire system is based on providing “live event data in real time.” Ex.1017, Abstract. The MQSD can be an “internet server.” *Id.* ¶35. Spivey communicates with client devices over “any ... combination of wired and/or wireless networks including ... the [i]nternet.” *Id.* ¶179. For example, client devices “may request and participate in a websocket connection,” which “typically comprises a two–way connection between two entities.” *Id.* ¶¶43, 61. Moreover, consistent with Spivey, Ellis discloses providing event information over a separate communication path 152 than the video signal 150. Ex.1017 ¶¶39–41, Fig. 1. Thus,

a POSA would have understood Ellis in view of Spivey discloses transmitting first event information via a separate connection. Ex.1001 ¶345.

Further, this information is provided via a connection between a topic queue and the first VCD. Users can access the MQSD using a URL. *See* Ex.1017 ¶¶81–82. Then, the user can access the specific topic queue they wish to receive information from by using a text string, e.g., a topic label, associated with a topic queue. *Id.* ¶179, Fig. 9 (“Gamecenter.nfl.ravenssteelers.”). Thus, a POSA would have understood that Ellis in view of Spivey discloses distributing event information from a socket to a user device over a connection, such as an Internet connection. Ex.1001 ¶346. Further, a POSA would have understood that the internet connection is established using a URL and text string identifying a uniquely addressable communication endpoint on the MQSD, where first event information is transmitted to a VCD. *Id.*

3. Claim 24

a. Elements 24.0–24.1

To the extent the nature or content of the information (e.g., chat) is considered as limiting the claimed system, Ellis discloses sending and receiving chat information because users “may also leave feedback regarding the commentary that other users in the media system may view when they access the same commentary.” Ex.1016 ¶132. Such commentary includes “text.” *Id.* A POSA would have understood that Ellis discloses a system that transmits and receives first chat

information regarding the first live sporting event video-based commentary included in the first broadcaster's live stream of digital content. Ex.1001 ¶¶347–49; Ex.1016 ¶126.

Ellis also discloses providing this information on a separate channel to users. Ellis discusses transmitting the video “broadcasts” over communication path 150 and “game summary information, . . . and any other sports-related information” over communication path 152. *Id.* ¶41 (“sports scores and game status information may be included in one real-time data feed, while sports commentary may be included in another data feed.”). Such commentary includes “text.” *Id.* ¶132. Thus, a POSA would have understood that Ellis discloses transmitting via at least one first chat/system event Internet communication channel between at least one first chat/system event socket of the at least one socket server and the first VCD of the first viewer. Ex.1001 ¶350. To the extent that the claim could also be interpreted as reciting “*a first plurality of viewer client devices*,” Ellis discloses this aspect as explained at §VIII.A.2.d.iii.

Ellis in view of Spivey discloses a socket specifically designated transmitting/receiving chat/system information. Spivey utilizes topic queues to organize “data payloads” that “relate to a specific game between two NFL teams.” Ex.1017 ¶¶51–52. The data payloads, “[w]hen the live event is a sporting event,” include, “for example, a score, a play, . . . and so forth.” *Id.* ¶48. Topic queues may

be “hierarchically related.” *Id.* ¶52. This means that when the system looks at the text string associated with the data payload, the system “may place the data payload only in the topic queue exactly matching the topic label.” *Id.* ¶104; Ex.1001 ¶351. For example, if there were two topics “at the same hierarchal level” the data payloads placed in “gamecenter.nfl.ravenssteelers.statistics” would not be placed in” another topic queue “gamecenter.nfl.ravenssteelers.plays.” In sum, the statistics would have been separate from the chat information. *Id.* Thus, a POSA would have understood Spivey discloses a first chat/system event socket (“gamecenter.nfl.ravenssteelers.chat”) corresponding to the first chat information regarding the first live sporting event video–based commentary. *Id.*

B. Ground 2: Claim 26 is Obvious over Ellis in View of Spivey, Kellicker and Phillips

1. Motivation to Combine

Kellicker teaches that one of the objectives of the arrangement described therein is to “[coordinate] data analytics to improve performance of streaming servers, client devices, and analytics servers.” Ex.1049 ¶4. The arrangement disclosed by Kellicker is illustrated below.

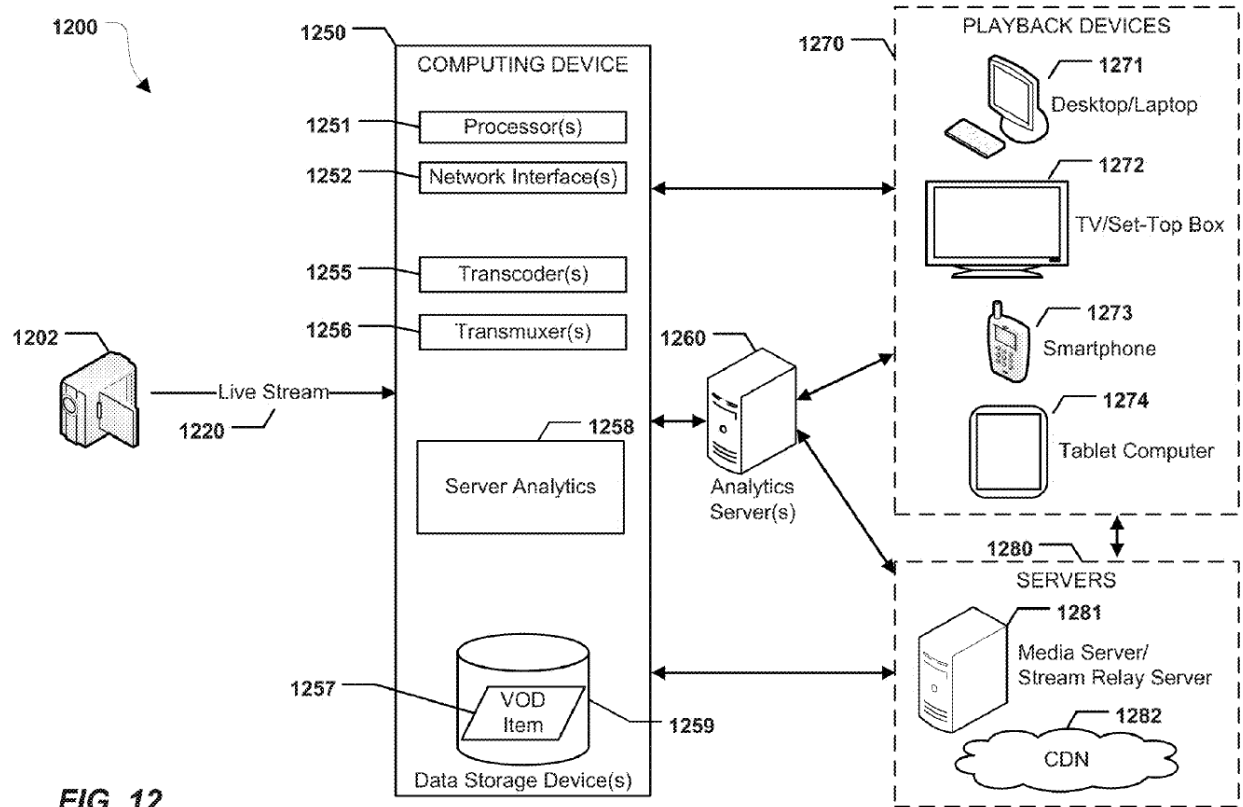


FIG. 12

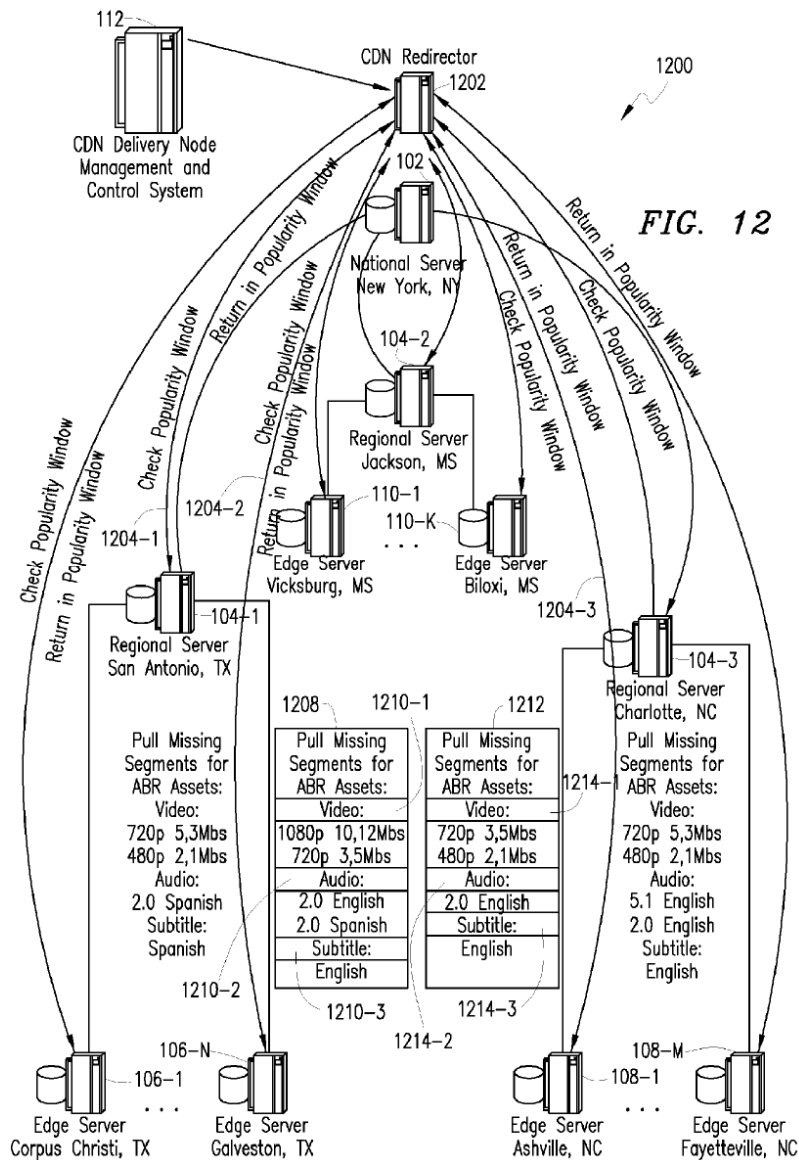
Id. Fig. 12.

A POSA would have understood that the “computing device” 1250 of Kellicker is analogous to the “media sources” of the ‘697, as well as the content sources 130 of Ellis. *See* §VIII.A.2.b; Ex.1001 ¶¶115–17. The computing device 1250 of Kellicker is a server that receives incoming live media stream 1220 from a capture source (e.g., broadcasting client) 1202, processes the incoming media and serves it to playback devices 1270 either directly, or through other servers 1260, 1280. Server 1250 includes a processor, a memory and a network interface. Ex.1049 ¶¶20, 99, Fig. 12. Kellicker teaches that server 1250 can receive a live media stream 1220, and stream that media to user devices. *Id.* ¶102; Fig.12. Kellicker teaches

server 1250 can support a number of real-time data communication protocols, including HLS, RTMP and/or WebRTC. Kellicker discloses that server 1250 can be implemented as a single device/server or as a plurality of devices/servers. *Id.* ¶¶98 (“the computing device 1250 may correspond to the streaming server 120”); and ¶19 (“the streaming server 120 . . . may be distributed across a plurality of devices”). Ex.1001 ¶¶118–19.

As discussed above, Kellicker teaches an HLS caching server architecture (e.g., 1280) communicatively coupled to at least one RTMP media server and/or at least one WebRTC media server (e.g., 1250). Kellicker teaches that this server (e.g., 1280) may be a “server of a content distribution network (CDN) 1282.” Ex.1049 ¶102. While Kellicker does not expressly describe the CDN architecture, conventional prior art CDN architectures routinely included both “parent” and “children” (regional, edge or point of presence (PoP)) servers. An exemplary CDN architecture that may be employed to distribute HLS media streams is described by Phillips. Ex.1050 ¶31 (“content may be delivered via CDN 100 using . . . HTTP Live Streaming or *HLS*”), FIG. 1. Phillips teaches that one of the objectives of its CDN architecture is to “[optimize] defragmentation of content in a content delivery network (CDN).” *Id.* Abstract. Phillips does this, at least in part, by providing a hierarchal server structure wherein “a central or national distribution node may in turn operate as a parent node to one or more child regional distribution nodes.” *Id.*

¶31. Phillips teaches that this architecture provides “for high-performance streaming . . . to subscribers using one or more Internet-based infrastructures.” *Id.* ¶30; Ex.1001 ¶121–25. The CDN architecture disclosed and taught by Phillips is depicted, for example, in Fig. 12 therein, reproduced below.



Ellis, Spivey, Kellicker and Phillips all involve media content delivery systems including real-time streaming of live or real-time media. Each describe

methods and/or systems for streaming video and/or event data to one or more users. The references solve the same or similar problems, namely, providing video and/or event data to multiple users, and provide the ability to support the processing and transmission of the broadcast media utilizing several different well-known transmission protocols and architectures. Ex.1001 ¶¶126, 120. A POSA would have successfully combined Ellis, Spivey, Kellicker and Phillips because such modifications involve only the substitution or addition of known elements in a predictable manner to achieve a known or desired result. Ex.1001 ¶¶126, 120.

2. Claim 26

a. Elements 26.0–26.2

As explained in §VIII.B.1, Kellicker discloses a server (120/1250) that receives incoming live media streams, processes the incoming media and serves it to playback devices, including receipt of a live incoming media stream (1220), and real-time streaming of the media. Ex.1049 ¶¶101, 106. Kellicker discloses that this server can be implemented through a plurality of such servers. *Id.* ¶98(“the computing device 1250 may correspond to the streaming server 120”); and ¶19(“the streaming server 120 . . . may be distributed across a plurality of devices”). Thus, Kellicker discloses a plurality of media sources in the form of real-time servers. Ex.1001 ¶¶352–54.

Further, Kellicker teaches that the server(s) may support various streaming protocols including, RTMP, and WebRTC. Ex.1049 ¶106(“computing device 1250 may also support real time messaging protocol (**RTMP**) (and variants thereof), real–time streaming protocol (RTSP), real–time transport protocol (RTP), web real–time communication (**WebRTC**), and MPEG–2 transport stream (MPEG–TS)”). Thus, Kellicker discloses teaches media sources in the form of a RTMP media server and/or a real–time WebRTC server. Ex.1001 ¶355.

It would have been obvious in view of Kellicker for a POSA to have implemented the content source(s) 130 of Ellis in the form of a real–time a RTMP media server or a real–time WebRTC server. A POSA would have been motivated to do so in order to provide the content source of Ellis with the processing and transmission ability to conform with these well–known and widely utilized protocols. Ex.1049 ¶¶103–106; Ex.1001 ¶356. Indeed, Kellicker teaches providing its server(s) with the capability to “enable the computing device 1250 to process data in accordance with multiple coding technologies and protocols.” Ex.1049 ¶104.

It is noted that claim 26 only requires *either* a RTMP server *or* a WebRTC server. Nevertheless, it would have also been obvious to a POSA in view of Kellicker to have modified Ellis to implement the content source(s) 130 therein in the form of multiple servers, including a RTMP server *and* a WebRTC server. A POSA would have recognized that doing so provides the system with even more flexibility and

adaptability to handle different types of incoming media and comply with multiple types of protocols. Ex.1001 ¶357.

b. Element 26.3

Kellicker teaches at least one RTMP media server and at least one WebRTC media server for receiving a livestream in a corresponding protocol and transforming that livestream into another protocol for transmission to user devices. Ex.1049 ¶¶103–106; Ex.1001 ¶358–59.

Kellicker teaches that the “network interface(s) 1252 [of computing device 1250] may further be configured to transmit data to the one or more other servers 1280... a media/relay server (1281), a server of a content distribution network (1282)” (e.g., a regional or edge server), etc. Ex.1049 ¶102, FIG. 12. Kellicker explains that the data transmitted to the playback to the servers 1280 includes “media streams (e.g., streams including media content, such as audio, video, text, still images, or a combination thereof).” *Id.* Accordingly, Kellicker teaches that the RTMP media server and/or the WebRTC media server are communicatively coupled to the one or more other servers 1280. *Id.*; Ex.1001 ¶359.

Kellicker further discloses that the media streams received by the one or more other servers 1280 “may be associated with the same encoding format and transmission protocol or may be associated with different encoding formats and transmission protocols.” Ex.1049 ¶102, FIG. 12. Kellicker also specifically teaches

providing the real-time streaming server(s) with the HLS streaming protocol ability. *Id.* at ¶106. It would have been obvious to a POSA to implement this capability in the content source(s) of Ellis in the form of the one or more other servers 1280. A POSA would have been motivated to provide HTTP live streaming (HLS) media streams because, as was well-known to a POSA, HLS provides a reliable, cost-effective means of delivering continuous and long-form video over the Internet. Ex.1001 ¶360.

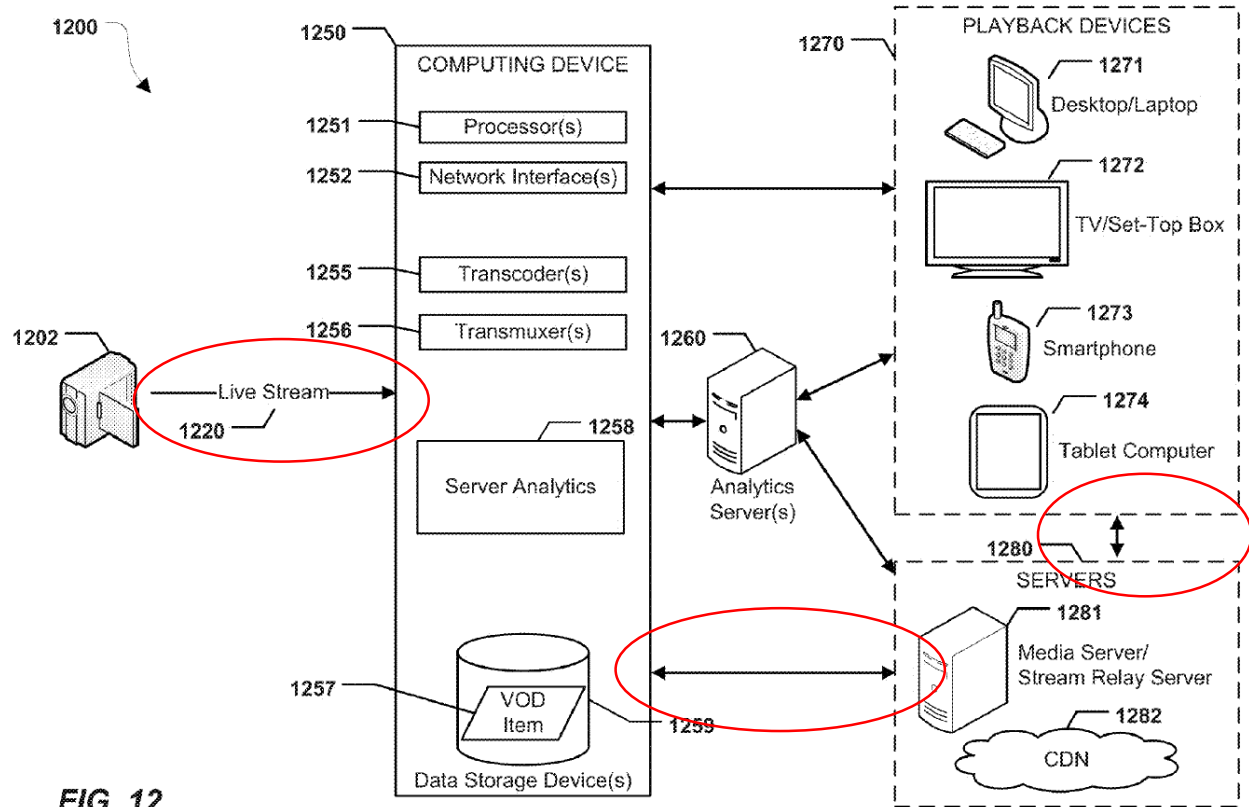
As explained in Kellicker “the other servers 1280 include a media server/stream relay server 1281 and a server of a content distribution network (CDN) 1282.” Ex.1049 ¶102. As would have been readily understood by a POSA, a CDN is a caching architecture. Ex.1001 ¶361. Furthermore, it would have been obvious to a POSA to have provided the other HLS-compatible server(s) 1280 architecture with caching capability in order to improve responsiveness to requests from client devices to retrieve the streamed media from the other server(s) 1280, which is one of the fundamental objectives of a CDN server architecture. Ex.1001 ¶361.

c. Element 26.4–26.5

Kellicker teaches an HTTP live streaming (HLS) caching server CDN architecture communicatively coupled to at least one RTMP media server and/or at least one WebRTC media server. *See* §VIII.B.2.b. It would have been obvious to a POSA to have provided this HLS caching server CDN in the form of a hierarchal

server structure wherein “a central or national distribution node may in turn operate as a parent node to one or more child regional distribution nodes.” Ex.1050 ¶31. Consistent with the teachings of Phillips, a POSA would have been motivated to provide this hierarchal server architecture in order to provide “for high–performance streaming . . . to subscribers using one or more Internet–based infrastructures.” *Id.* ¶30. For example, Phillips teaches that this hierarchal server structure provides the ability to optimize defragmentation of content in the CDN. *Id.* ¶57. For instance, “[w]hen edge servers do not hold a copy of a requested segment file, they may be required to retrieve or pull the requested segment file from a parent node.” *Id.* ¶58, 59. A POSA would have been further motivated to implement such a parent/child HLS caching server CDN architecture in order to achieve greater CDN efficiency in the delivery of content. *Id.* ¶67(“the embodiments of the present disclosure can be advantageously implemented that allow CDNs to function more efficiently by making available what is needed where it is needed”). Ex.1001 ¶¶362–63.

Kellicker, as modified by Phillips, establishes a communication path from a first broadcaster’s livestream (1220) to a first server architecture (1250) which may be implemented as a plurality of media sources including at least one RTMP media server and/or at least one WebRTC media server, to a CDN in the form of parent/child HLS caching server CDN architecture (1280/1282), and from the CDN to a client/user device (1270):



Ex.1049, Fig.12(annotated).

Specifically, Kellicker discloses that the media server(s) 1250 is configured to transmit data to “a server of a content distribution network (CDN) 1282.” *Id.* ¶102. Further, Kellicker discloses “[i]n an embodiment where the analytics server(s) 1260 is collocated with a streaming server (e.g., the computing device 1250), first HTTP port(s) may be used to communicate a master rendition list and media chunks”. *Id.* ¶110; Ex.1001 ¶364–66.

Similarly, Phillips teaches that content that is requested for delivery by the CDN is received in the form of a “manifest” (i.e., an HLS file suite) that is made available to a “content delivery node” of the CDN server architecture. In the event

that the content delivery node does not possess the requested files, they can be retrieved from another node in the CDN architecture, such as a regional or edge server node (i.e., “child” server). Ex.1050, Abstract ¶31, Fig. 1. To the extent the nature/contents of the files being handled by the system is considered as limiting the claimed system, a POSA would have understood that a “manifest” or a “master rendition list” constitutes an “HLS file suite.” Ex.1001 ¶367. Phillips’s CDN architecture discloses mother servers (e.g., national server, regional servers), which are mother servers to children servers (e.g., regional servers, edge servers). A POSA would have readily appreciated that the mother–child relationship is defined by the hierarchy itself. The regional child servers are in–turn coupled to their children (e.g., edge servers). These edge servers form an HLS content delivery network (HLS CDN), such that, as discussed below, “the system establishes the first video Internet communication channel... between the first viewer client device and the HLS CDN.” Ex.1050 ¶31, FIGS. 1, 12; Ex.1001 ¶367.

Furthermore, with respect to the “HLS file suite” this represents nothing more than conventional HLS file transfer protocol, as admitted by PO. Ex.1006, 38:47–55. A POSA would have understood that the transfer of media according to the HLS standard as taught by both Phillips and Kellicker includes the use of exchanged HLS file suites, or manifests/master rendition lists. Ex.1001 ¶368.

Thus, the arrangement of Kellicker, as modified by Phillips, includes a mother server of a CDN that is “communicatively coupled . . . to receive . . . a first HLS file suite representing the first broadcaster’s live stream of digital content.” The node which first received the output from the RTMP and WebRTC media sources would operate as an HLS mother server. Ex.1001 ¶369.

Alternatively, this aspect of the claimed invention would have been obvious to a POSA in order to render the server HLS compatible. Ex.1001 ¶370–71.

d. Element 26.6

As explained above in §VIII.A.2.b, Ellis, as modified by Spivey, establishes the first video communication channel to provide the first copy of digital content to the first viewer client device. As further explained in §§VIII.B.2.b–c, Ellis, as modified by Spivey, Kellicker and Phillips, establishes that the first video communication channels are provided between the first viewer client device and the HLS CDN. Ex.1001 ¶372.

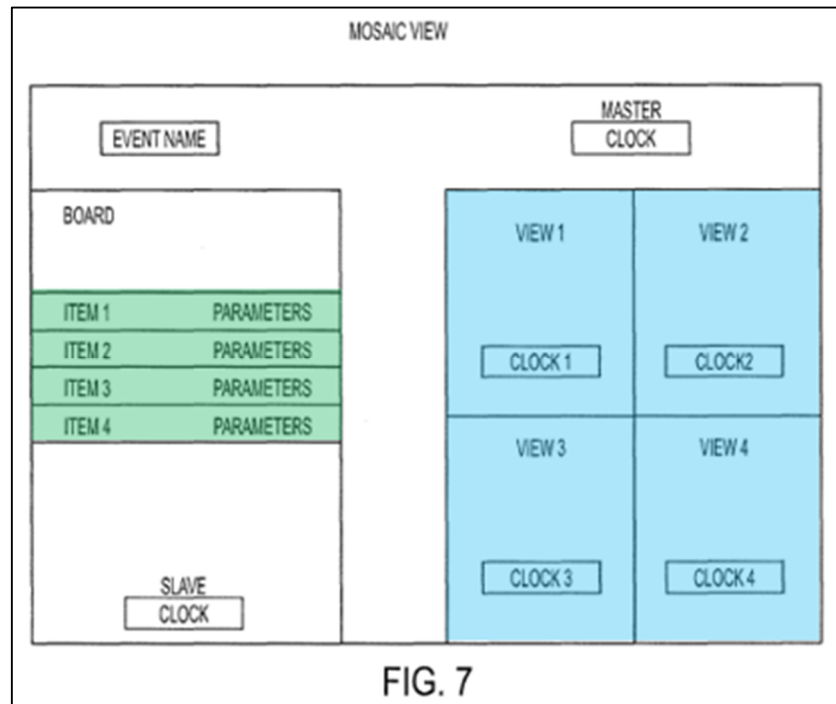
As explained in §VIII.B.2.c, Ellis, as modified by Spivey, Kellicker and Phillips, establishes that the first copy of the first broadcaster’s live stream of digital content is based on the (first) HLS file suite. Ex.1001 ¶373.

Thus, Ellis as modified by Spivey, Kellicker and Phillips satisfies this element.

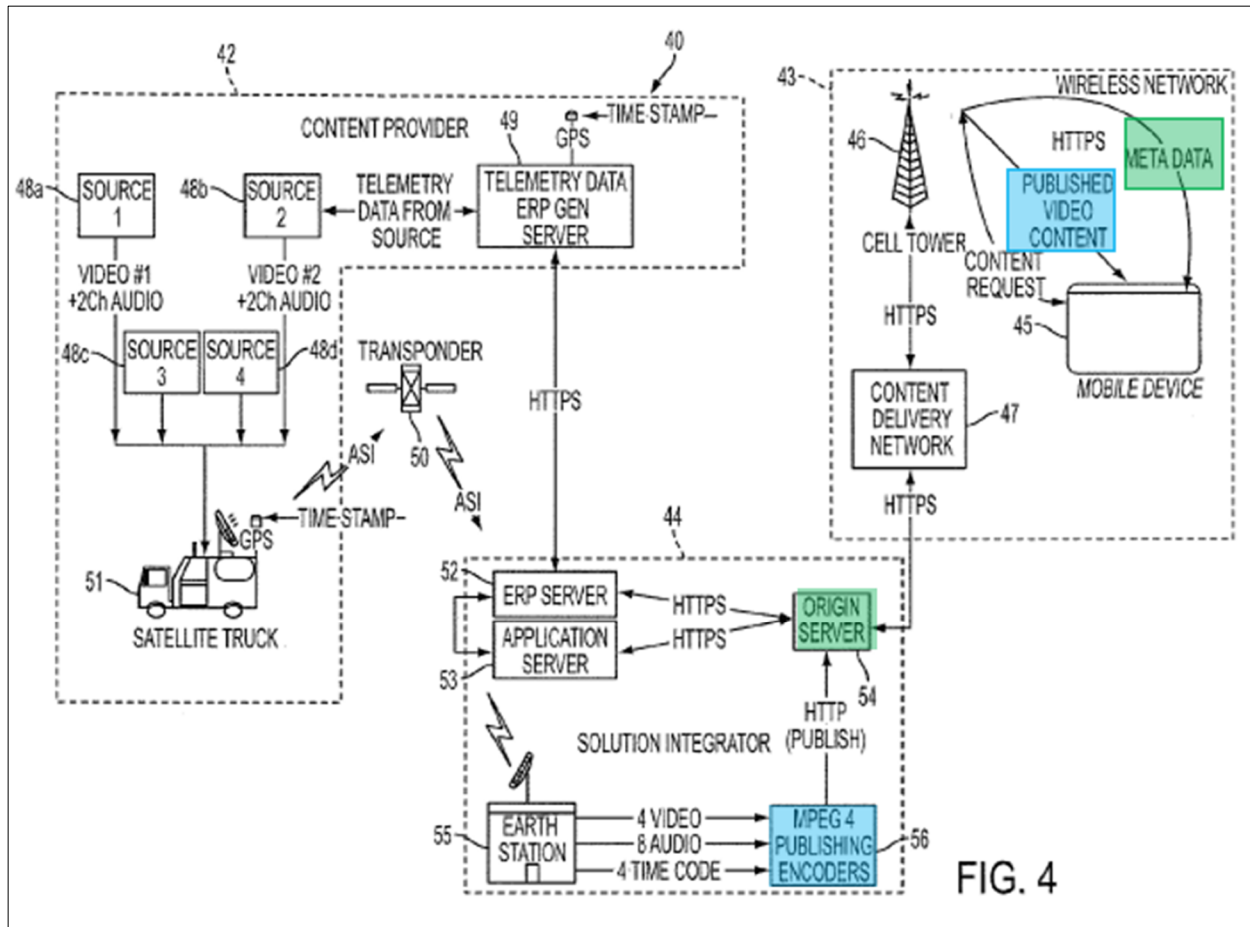
C. Ground 3: Claims 23 and 24 are Obvious Over Herzog in view of Spivey and Ellis

1. Motivation to Combine

Herzog discloses a system for viewing multiple videos related to the same sporting event in addition to live information about the event (meta-data). Ex.1001 ¶138. For example, Figure 7 shows “four different videos of a live event and meta-data of multiple parameters” from which a user can select:



Ex.1015 ¶¶12, 80–81, Fig. 7(annotated). Herzog provides these video and data streams over separate connections:



Ex.1015 ¶¶48–52, 67, Figs. 4(annotated), 5 and 6; Ex.1001 ¶138.

a. Socket Server (Spivey)

Herzog does not specifically disclose the backend server architecture supporting the data distribution—the claimed socket server. Ex.1001 ¶139. However, a POSA would have known that the websocket client–server communication protocol employed by Spivey was an alternative to the HTTP polling used by Herzog. Ex.1001 ¶139; §VIII.A.1. A POSA would have found it obvious to implement Herzog’s Origin Server as a websocket server, as taught by Spivey, to

achieve full-duplex communications between a client and a remote host with low latency. Ex.1001 ¶139.

As explained above in §VIII.A.1, a POSA would have recognized that socket servers were well-known components for implementing real-time data delivery, such as in Herzog, with reduced latency. Ex.1001 ¶¶140; Ex.1017 ¶¶35, 91–92, 96, Fig. 11. A POSA also would have appreciated the benefits associated with Spivey’s MQSD (i.e., “socket server”) which establishes websocket connections with client devices, including convenient and efficient full-duplex communications between a client and a remote host with low latency. Ex.1017 ¶¶27–28, 43, 61, 74, 91–92, 96, Fig. 11; Ex.1001 ¶140. A POSA, looking to improve Herzog’s latency issues, would have looked to implement a system such as Spivey. Ex.1001 ¶140.

A POSA would have done so by incorporating Spivey into Herzog’s data delivery system by implementing Spivey’s MQSD at Herzog’s Origin Server to achieve a predictable result of decreased latency data delivery. *See* Ex.1015 ¶¶50–51, Fig. 4; Ex.1017, Fig. 4; Ex.1001 ¶141. The combined system would maintain Herzog’s video content delivery system and upgrade the data delivery system to reduce latency. Ex.1001 ¶141.

A POSA would have been motivated to combine Herzog with Spivey for many of the same reasons that a POSA would have been motivated to combine Ellis with Spivey. First, Herzog and Spivey relate to the same subject matter—content

delivery for real-time events. *See* Ex.1017 Abstract; Ex.1015. Both references describe methods and/or systems for streaming video and/or event data to one or more users, such that video and/or event data are transmitted simultaneously. *See*, e.g., Ex.1017 Abstract; Ex.1015 Abstract; Ex.1001 ¶142. Second, the references solve the same or similar problems, namely, providing video and/or event data to multiple users over the Internet quickly. *See*, e.g., Ex.1017 ¶27; Ex.1015 ¶¶3, 19, 51; Ex.1001 ¶142. A POSA would have been further motivated to combine Herzog with Spivey because Spivey provides a complementary solution describing the details of the backend architecture for organizing and transmitting event data to multiple users. Ex.1017, Abstract; Ex.1001 ¶143.

A POSA would have had a reasonable expectation of success in combining Herzog with Spivey because Spivey contemplates being implemented in a digital content delivery system such as Herzog, stating that it can be used with “any well-known communication techniques and protocols, such as techniques suitable for use with packet-switched networks (e.g., public networks such as the Internet, private networks such as an enterprise intranet, and so forth).” Ex.1017 ¶155; Ex.1001 ¶143.

Combining the teachings of Herzog and Spivey would have been obvious because doing so would require only the simple substitution of known elements to obtain predictable results. Ex.1001 ¶143. Herzog encourages this flexibility, suggesting “modifications ... in various forms and examples, and [stating] that the

teachings may be applied in numerous applications.” Ex.1015 ¶90. To a POSA, it would have been obvious to substitute any of Herzog’s meta-data distribution elements with any combination of Spivey’s LDSD, MQSD, and/or Live Data Source(s), which are similarly involved in transmitting live event data to users and may be configured in any manner. *See* Ex.1017 ¶34; Ex.1001 ¶143.

b. Providing Users with Sports Score and Sports
Commentary Information Using Separate
Communication Channels (Ellis)

Ellis is directed to systems and methods “for providing enhanced information access and viewing functionality for sports-related programming.” Ex.1016, at ¶8. To this end, Ellis teaches that “sports score . . . information may be included in one real-time data feed, while sports commentary may be included and another feed.” *Id.* ¶41. A POSA would have understood that Herzog was already equipped to provide multiple, different live video and meta-data streams, so configuring the system to provide sports score and sports commentary information/data in separate data feeds would have been obvious and well within the skill of a POSA. Ex.1001 ¶¶144, 147. A POSA would have been motivated to configure Herzog with Ellis’s functionality because, as taught by Ellis, doing so serves to enhance information access and viewing functionality for the user. A POSA would have understood that users would enjoy having more content options, e.g., sports commentary, score

information, chat information, etc. Ex.1016 Abstract, ¶¶46, 68, Figs. 1, 2B, 3; Ex.1001 ¶¶149–51, 153.

A POSA would have had a reasonable expectation of success in providing separate feeds of sports score and sports commentary information because Ellis specifically contemplates being implemented in a content delivery system such as Herzog, stating that it can be used with “any suitable source of media content,” which includes IPTV content “delivered on a particular network address (e.g., multicast address) on which an IP socket is currently joined.” Ex.1016 ¶¶39–40, 74; Ex.1001 ¶¶148, 152.

2. Claim 23

a. Element 23.0

Herzog provides “live video and audio data” to user devices over a “communication network.” Ex.1001 ¶¶375–76; Ex.1015 Abstract. Herzog improves on traditional “sports application systems” where the meta-data is included “as part of the video broadcast.” Ex.1015 ¶83, Fig. 4; §VIII.C.1 and §VII.B.

To the extent the nature of the content is considered as limiting the claimed method, a POSA would have readily appreciated that broadcasts of live events would obviously, if not inherently, also include video-based commentary. Ex.1016 ¶38; Ex.1001 ¶376.

Alternatively, as explained in §VIII.C.1.b and §VIII.A.2, Ellis teaches providing sports commentary in a data feed or over communications channel to a user.” *Id.* ¶41. Ellis teaches that this enhances information access and viewing functionality for the user. A POSA would have understood that users would enjoy having more content options, e.g., sports commentary, score information, chat information, etc. Ex.1016 Abstract, ¶¶46, 68, Figs. 1, 2B, 3; Ex.1001 ¶377. Thus, its inclusion in the video and/or audio data being served to users according to the system of Herzog would have been obvious to a POSA. Ex.1001 ¶377.

Herzog discloses a content producer (42) that provides a live stream of content, including video, to a “solutions integrator” (44), which then provides the content to a mobile device:

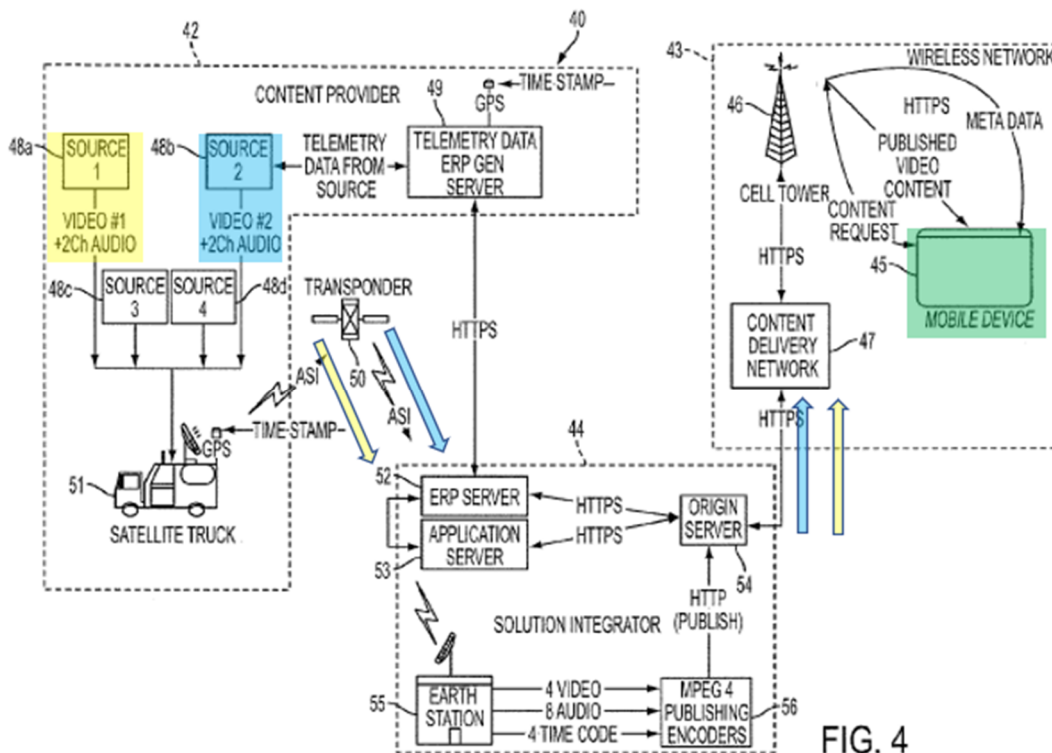


FIG. 4

Ex.1015 ¶¶44–51, Fig. 4(annotated).

The above system focuses on one live event, a car race, but can be “used for any sporting event.” *Id.* ¶¶45, 83. The system makes these video feeds available to “mobile stations” and other “similar users.” *Id.* ¶19. Each user can then request video from “one or more channels of video/audio for viewing a live event.” *Id.* ¶70. The user then “receives both the video/audio stream” using HLS. *Id.* ¶51. Thus, Herzog provides a system where a plurality of different users can each receive a respective copy of a first broadcaster’s live stream of digital content including first live sporting event video–based commentary from a broadcaster client device to a plurality of VCDs. Ex.1001 ¶¶378–79.

This element is satisfied by either Herzog alone, or by Herzog, as modified by Ellis.

b. Element 23.1

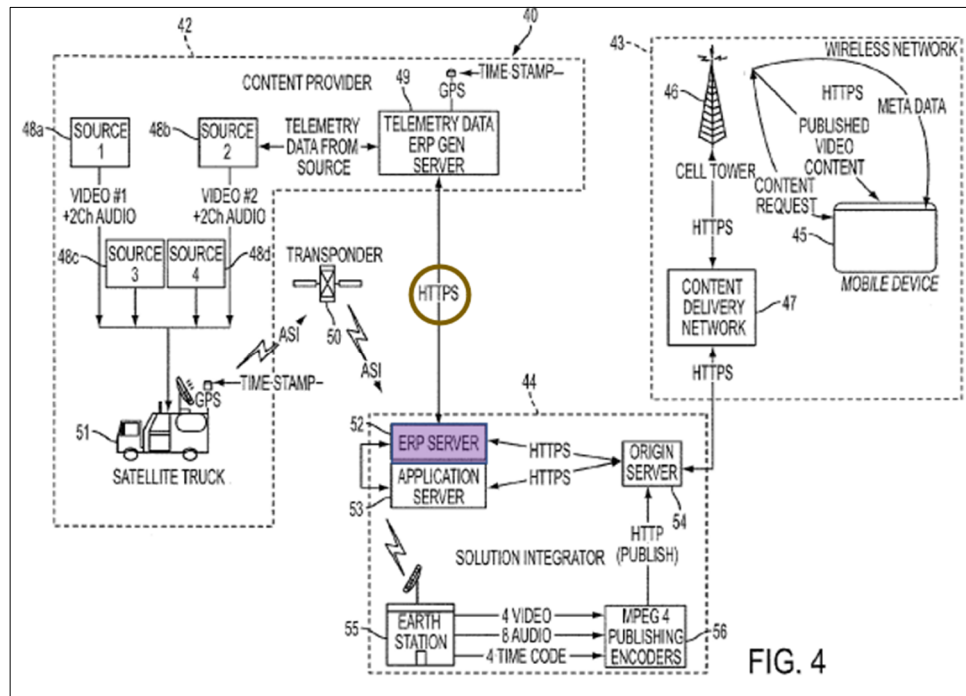
Herzog discloses multiple media sources in the form of, for example, multiple MPEG 4 Publishing Encoders (“Encoders”) as a plurality of media sources. Ex.1015 ¶48. These encoders receive video (and audio) data from the live event over a communication channel. Ex.1015 ¶¶46, 49, Figs. 4–6. The Encoders can “encode” the video and audio data and then the “encoded video is then published to the content delivery network 47 (for example) via IP utilizing [] HLS.” *Id.* ¶¶49, 55. The video

content is published to the user via a telecommunications network or the Internet.

Id. ¶27, Fig. 4. Herzog's content sources perform the same functions as the media sources as recited in this claim element. Ex.1001 ¶¶380–81.

c. Element 23.2

Herzog's **ERP Server** connects to the ERP Gen Server over the internet:



Ex.1015 ¶¶45, 47, 50, Figs. 4(annotated), ¶6. The ERP Server is a generic server that retrieves meta-data from the ERP Gen Server. *Id.* ¶¶16, 47, 50. It contains the same components as the '697's control server: a processor, memory device, and communication interfaces. *Id.* ¶¶13, 85–86, Fig. 8; Ex.1006 55:60–56:3. The ERP Server “acts as an intermediary between the ERP generator server and the origin server.” Ex.1015 ¶50. The meta-data retrieved includes event information germane to the sporting event such as car status, driver status, lap number, car speed, etc. *Id.*

¶45; Ex.1001 ¶382. The ERP Server retrieves this first event (score) information via the internet from an event information provider because it retrieves the information in HTTPS format, an internet communication protocol, from an ERP Server. Ex.1015 ¶¶17, 50. Thus, the ERP server periodically retrieves the metadata in accordance with HTTP protocol. Ex.1001 ¶383. A POSA would have also understood that HTTP is a pull-oriented protocol where receivers such as the ERP server need to explicitly ask for new data. To reduce latency using such an HTTP connection, a POSA would have known to have Herzog's ERP server periodically check for the content availability as often as possible. Ex.1001 ¶384.

Alternatively, and to the extent the frequency with which the information is retrieved is considered as limiting to the claimed system, Ellis's SCS periodically retrieves event information germane to a live sporting event, e.g., score information." Ex.1016 ¶¶41–42, 72, 142; Ex.1001 ¶385. Ellis's SCS also comprises a control server. Ex.1016 ¶45("supplemental content source 120 may be configured to transmit signals to user's equipment 102 over any suitable communication paths . . ."). It would have been obvious to a POSA, in view of Ellis, to have included event information in the metadata exchanged by Herzog's system in order to provide "enhanced information access and viewing functionality for sports-related programming. Ex.1016, at ¶8; Ex.1001 ¶385; *see* Ex.1016 ¶¶42, 142.

While a periodic time interval is not mentioned in Ellis, a POSA would have understood that any system that provides real-time sports statistics, such as Herzog, would update the statistics and/or other live event information on a periodic basis. Ex.1001 ¶386. This is especially true given that no specific time interval is specified by the claim.

Alternatively, it would have been obvious to a POSA to implement periodic retrieval of the event information in order to provide the user with up-to-date information thereby enhancing the overall viewing experience, whilst limiting queries to the system and load on the components. Ex.1001 ¶386.

d. Element 23.3

Herzog in view of Spivey and Ellis discloses a **socket server** communicatively coupled to the **control server** (which, as discussed above, is disclosed by Herzog's **ERP Server**). Herzog discloses providing event information over a separate connection to user equipment. Ex.1015 ¶¶48–52, 67, Figs. 4, 6; Ex.1001 ¶387. That process includes the **ERP Server** transmitting data to the **Origin Server**:

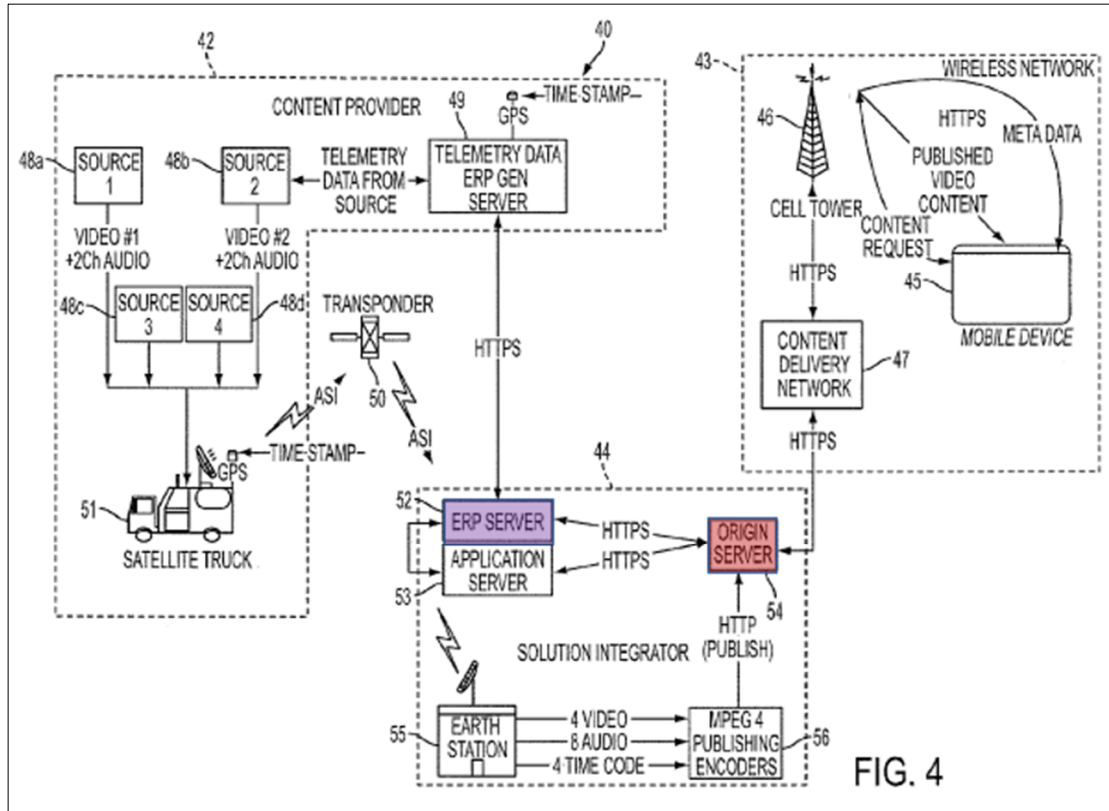


FIG. 4

Ex.1015 ¶¶50–51, Fig. 4(annotated). For at least the same reasons explained above in §VIII.A.2.d, a POSA would have incorporated Spivey’s **MQSD** into Herzog’s **Origin Server** to reduce latency. Ex.1001 ¶387; §VIII.C.1.a, §VIII.A.2.d. Thus, a POSA would have understood that Herzog’s **ERP Server** transmitting HTTPS information to **Spivey’s MQSD** discloses the ‘697’s **socket server** being communicatively coupled to a **control server**. Ex.1001 ¶387.

Herzog’s ERP Server “receives the meta–data from ERP generator server 49 in HTTPS communication format” and “acts as an intermediary between the ERP generator server and the origin server.” Ex.1015 ¶50, Fig. 4. Incorporating Spivey’s MQSD into Herzog’s origin server would thus allow the MQSD (socket server) to

receive from the ERP Server (control server) the meta-data which, as explained in §VIII.C.2.d and §VIII.C.1 above, includes score information (first event information). Ex.1001 ¶¶388–90; Ex.1015 ¶83.

- i. First event socket corresponds to the first event information germane to the first live sporting event

For at least the same reasons explained in §VIII.A.2.d, a POSA would have understood that Spivey’s topic queues disclose or render obvious the ‘697’s first event socket that communicates first event information including score information. Ex.1001 ¶¶392–397.

- ii. Transmitting via a first event information communication channel between at least one first event socket of the at least one socket server and the first VCD of the first plurality of VCDs⁴

As discussed in §§VIII.C.2.a–b above, Herzog discloses providing a first VCD with a first copy of digital content. As further explained in §§VIII.C.2.a–b above, Herzog provides a system where a plurality of different users can each receive a respective copy of digital content. Ex.1001 ¶¶398–400. Herzog also discloses, and Ellis renders obvious, the inclusion of score and/or other live event information in the digital content. *See* §VIII.C.2.c. Ex.1001 ¶397.

⁴ It is noted that there is no antecedent basis for the recited “the first plurality of viewer devices.” The claim is directed to “a first viewer client device.”

Both Herzog and Spivey further disclose transmitting the score information to a VCD of the first plurality of VCDs. For example, in Spivey, the MQSD “may push data from topic queues ... to the subscribers of the topics corresponding to those topic queues.” Ex.1017 ¶46; Ex.1001 ¶400–02. In Herzog, the Origin Server may be configured to transmit “the meta–data to the content delivery network 47,” which includes car position, and the user receives the data via HLS. Ex.1015 ¶¶45, 51. A POSA would have known, and considered obvious, that the websocket client–server communication protocol employed by Spivey was an alternative to Herzog’s HTTP polling. Ex.1001 ¶¶401–02.

As previously noted, Ellis discloses at least a first VCD. *See* §VIII.A.1.. To the extent that the claim could also be interpreted as reciting “*a first plurality of viewer client devices*,” Ellis discloses multiple broadcasters providing copies of each broadcaster’s live stream of digital content including live sporting event video–based commentary to a plurality of user devices by providing a broadcast system where any “number of users” may receive signals from a content source. Ex.1016 ¶39. The signals are individual copies of the same live stream of digital content. *Id.* ¶¶40, 62, Fig. 2B; Ex.1001 ¶344; §VIII.A.1.

Thus, Herzog as modified by Spivey, and optionally Ellis, satisfies these elements.

3. Claim 24

a. Elements 24.0–24.1

To the extent the nature or content of the information (e.g., chat) is considered as limiting the claimed system, Ellis teaches sending and receiving chat information because users “may also leave feedback regarding the commentary that other users in the media system may view when they access the same commentary.” Ex.1016 ¶132. Thus, a POSA would have understood, and considered obvious, that Herzog in view of Spivey and Ellis discloses/suggests a system that transmits and receives chat information regarding live sporting event video–based commentary included in the broadcaster’s live stream of digital content in order to enhance interaction between the users/viewers of an event. Ex.1001 ¶¶145–48, 403–06; Ex.1016 ¶126; §VIII.C.1.

Ellis, consistent with Herzog’s separate meta–data stream, also teaches providing this event data and chat information on a separate channel to users. Ex.1001 ¶407; Ex.1015 ¶¶49–53, 59–61, Fig. 5; Ex.1016 ¶¶39–41, Fig. 1. Ellis discloses transmitting the video “broadcasts” over communication path 150 and “game summary information, . . . and any other sports–related information” over communication path 152. Ex.1016 ¶41. Consistent with Herzog, Ellis explains that in “some embodiments, this sports–related information is provided to user equipment 102 as a plurality of real–time data feeds.” Ex.1016 ¶41; Ex.1001 ¶407.

Thus, a POSA would have understood, and considered obvious, that Herzog in view Ellis of Spivey discloses transmitting the information via a chat/system event Internet communication channel between a chat/system event socket of the at least one socket server and the first VCD of the first plurality of VCDs. Ex.1001 ¶407.

Herzog in view of Spivey and Ellis discloses and renders obvious a socket specifically designated for chat information when implementing Spivey's topic queue organizational structure. Ex.1001 ¶408; §VIII.A.3. Thus, a POSA would have understood Spivey discloses, and render obvious, a first chat/system event socket corresponding to the first chat information regarding the first live sporting event video-based commentary. Ex.1001 ¶408.

D. Ground 4: Claim 26 is Obvious over Herzog in View of Spivey, Ellis, Kellicker and Phillips

1. Motivation to Combine

It would been obvious to combine Herzog, Spivey and Ellis for at least the reasons set forth in §VIII.C.1. Further, it would have been obvious to further combine these references with Kellicker and Phillips for at least the reasons set forth in §VIII.A.1 and §VIII.B.1. Ex.1001 ¶¶154–62.

a. Elements 26.0–26.2

As explained in §VIII.B.1 and §VIII.B.2.a, Kellicker discloses a server (120/1250) that receives incoming live media streams, processes the incoming media and serves it to playback devices, including receipt of a live incoming media stream

(1220), and real-time streaming of the media. Ex.1049 ¶¶101, 106. Kellicker discloses that this server can be implemented through a plurality of such servers. *Id.* ¶98(“may be distributed across a plurality of devices”). Thus, Kellicker discloses a plurality of media sources in the form of real-time servers. Ex.1001 ¶¶409–411.

Further, Kellicker teaches that the server(s) may support various streaming protocols including, RTMP, and WebRTC. Ex.1049 ¶106(“computing device 1250 may also support real time messaging protocol (**RTMP**) (and variants thereof), real-time streaming protocol (RTSP), real-time transport protocol (RTP), web real-time communication (**WebRTC**), and MPEG–2 transport stream (MPEG–TS)”). Thus, Kellicker teaches media sources in the form of an RTMP media server and/or a real-time WebRTC server. Ex.1001 ¶412.

Thus, it would have been obvious to a POSA, in view of Kellicker, to have implemented the media sources (i.e., “encoders” 56) of Herzog, at least in part, in the form of either a real-time a RTMP media server or a real-time WebRTC server. A POSA would have been motivated to do so to provide the content source of Ellis with the processing and transmission ability to conform with these well-known and widely utilized protocols. Ex.1049 ¶¶103–106; Ex.1001 ¶412–13. Indeed, Kellicker teaches providing its server(s) with the capability to “enable the computing device 1250 to process data in accordance with multiple coding technologies and protocols.” Ex.1049 ¶104.

It is noted that claim 26 only requires *either* a RTMP server *or* a WebRTC server. Nevertheless, it would have also been obvious to a POSA in view of Kellicker to have modified Herzog to implement the media sources 56 therein in the form of multiple servers, including a RTMP server *and* a WebRTC server. A POSA would have recognized that doing so provides the system with even more flexibility and adaptability to handle different types of incoming media and comply with multiple types of protocols. Ex.1001 ¶414.

b. Element 26.3

Kellicker teaches at least one RTMP media server and at least one WebRTC media server for receiving a livestream in a corresponding protocol and transforming that livestream into another protocol for transmission to user devices. Ex.1049 ¶¶103–106; Ex.1001 ¶416.

Kellicker teaches that the “network interface(s) 1252 [of computing device 1250] may further be configured to transmit data to the one or more other servers 1280... a media/relay server (1281), a server of a content distribution network (1282)” (*e.g.*, a regional or edge server), etc. Ex.1049 at ¶102, FIG. 12. Kellicker explains that the data transmitted to the playback to the servers 1280 includes “media streams (*e.g.*, streams including media content, such as audio, video, text, still images, or a combination thereof).” *Id.* Accordingly, Kellicker teaches that the

RTMP media server and/or the WebRTC media server are communicatively coupled to the one or more other servers 1280. *Id.*; Ex.1001 ¶416.

Kellicker further discloses that the media streams received by the one or more other servers 1280 “may be associated with the same encoding format and transmission protocol or may be associated with different encoding formats and transmission protocols.” *Id.* Kellicker also specifically teaches providing the real-time streaming server(s) with the HLS streaming protocol ability. Ex.1049 at ¶106. It would have been obvious to a POSA to implement this capability and server architecture in the media sources 56 of Herzog in the form of the one or more other servers 1280. A POSA would have been motivated to provide HTTP live streaming (HLS) media streams because, as was well-known to a POSA, HLS provides a reliable, cost-effective means of delivering continuous and long-form video over the Internet. Ex.1001 ¶417.

As explained in Kellicker, “the other servers 1280 include a media server/stream relay server 1281 and a server of a content distribution network (CDN) 1282.” Ex.1049 ¶102. As would have been readily understood by a POSA, a CDN is a caching architecture. Ex.1001 ¶417. Furthermore, it would have been obvious to a POSA to have provided the other HLS-compatible CDN server architecture with caching capability in order to improve responsiveness to requests from client

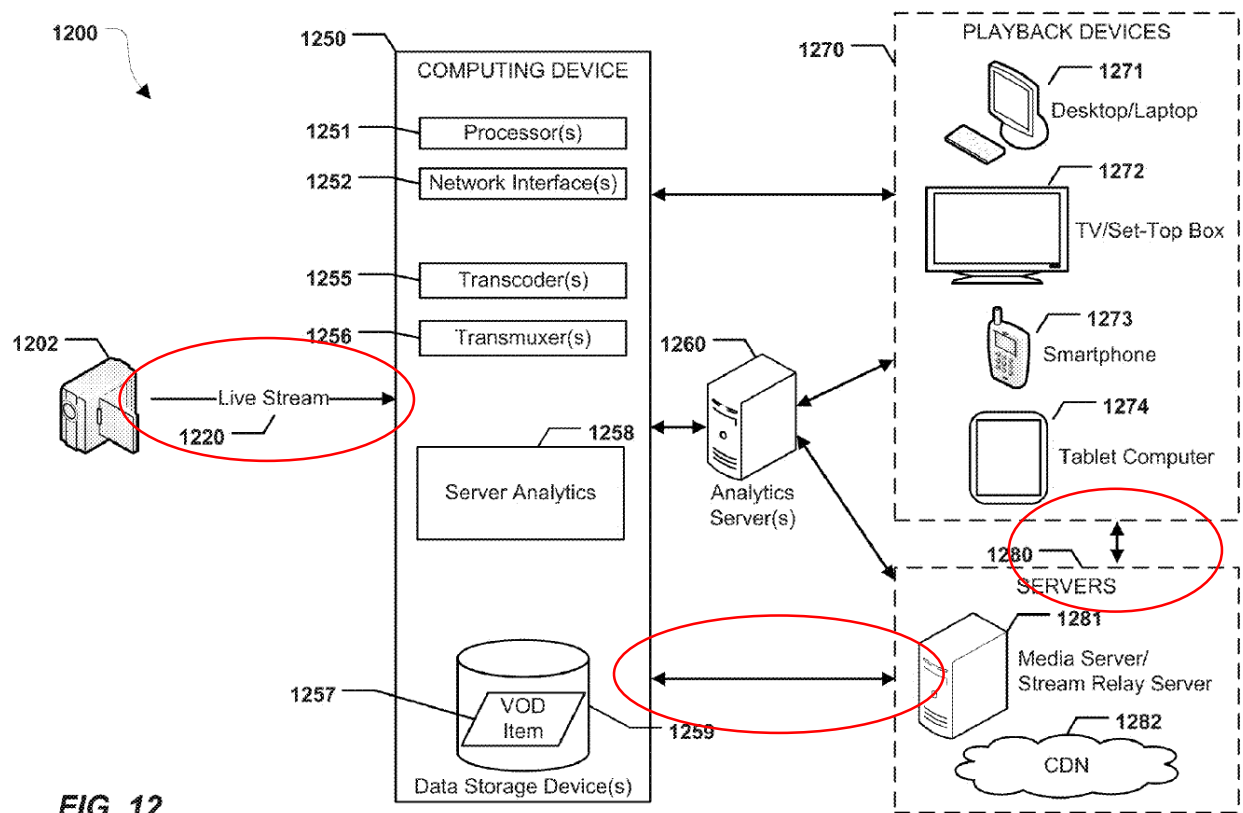
devices to retrieve the streamed media from the other servers 1280, which is one of the fundamental objectives of a CDN server architecture. Ex.1001 ¶418.

c. Element 26.4–26.5

As explained above, Kellicker teaches an HTTP live streaming (HLS) caching server CDN architecture communicatively coupled to at least one RTMP media server and/or at least one WebRTC media server. *See* §VIII.D.1, 1.a–b. It would have been obvious to a POSA to have provided this HLS caching server CDN in the form of a hierarchal server structure wherein “a central or national distribution node may in turn operate as a parent node to one or more child regional distribution nodes.” Ex.1050 ¶31. Consistent with the teachings of Phillips, a POSA would have been motivated to provide this hierarchal server architecture in order to provide “for high–performance streaming . . . to subscribers using one or more Internet–based infrastructures.” *Id.* ¶30. For example, Phillips teaches that this hierarchal server structure provides the ability to optimize defragmentation of content in the CDN. *Id.* ¶57. For instance, “[w]hen edge servers do not hold a copy of a requested segment file, they may be required to retrieve or pull the requested segment file from a parent node.” *Id.* ¶58, 59. A POSA would have been further motivated to provide the claimed parent/child HLS caching server CDN architecture in order to achieve greater CDN efficiency in the delivery of content. *Id.* ¶67 (“the embodiments of the present disclosure can be advantageously implemented that allow CDNs to function

more efficiently by making available what is needed where it is needed”). Ex.1001 ¶419–20.

Kellicker, as modified by Phillips, establishes a communication path from a first broadcaster’s livestream (1220) to a first server architecture (1250) which may be implemented as a plurality of media sources including at least one RTMP media server and/or at least one WebRTC media server, to a CDN in the form of parent/child HLS caching server CDN architecture (1280/1282), and from the CDN to a client/user device (1270):



Ex.1049, Fig.12(annotated).

Specifically, Kellicker discloses that the media server(s) 1250 is/are configured to transmit data to “a server of a content distribution network (CDN) 1282.” *Id.* ¶102. Further, Kellicker discloses that “[i]n an embodiment where the analytics server(s) 1260 is collocated with a streaming server (e.g., the computing device 1250), first HTTP port(s) may be used to communicate a master rendition list and media chunks”. *Id.* ¶110; Ex.1001 ¶¶421–22.

Similarly, Phillips teaches that content that is requested for delivery by the CDN is received in the form of a “manifest” (i.e., an HLS file suite) that is made available to a “content delivery node” of the CDN server architecture. In the event that the content delivery node does not possess the requested files, they can be retrieved from another node in the CDN architecture, such as a regional or edge server node (i.e., “child” server). Ex.1050 Abstract, ¶31, Fig. 1. To the extent the nature or contents of the files is considered as further limiting the claimed system, a POSA would have understood that a “manifest” or a “master rendition list” constitutes an “HLS file suite.” Ex.1001 ¶423. Phillips’s CDN architecture discloses mother servers (e.g., national server, regional servers), which are mother servers to children servers (e.g., regional servers, edge servers). A POSA would have readily appreciated that the mother–child relationship is defined by the hierarchy itself. The regional child servers are in–turn coupled to their children (e.g., edge servers). These edge servers form an HLS content delivery network (HLS

CDN), such that, as discussed below, “the system establishes the first video Internet communication channel... between the first viewer client device and the HLS CDN.” Ex.1050 ¶31, FIGS. 1, 12; Ex.1001 ¶682.

Furthermore, with respect to the “HLS file suite” this represents nothing more than conventional HLS file transfer protocol, as admitted by PO. Ex.1006 38:47–55. A POSA would have understood that the transfer of media according to the HLS standard, as taught by both Phillips and Kellicker, includes the use of exchanged HLS file suites, or manifests/master rendition lists. Ex.1001 ¶423–24.

Thus, the arrangement of Kellicker, as modified by Phillips, includes a mother server of a CDN that is “communicatively coupled . . . to receive . . . a first HLS file suite representing the first broadcaster’s live stream of digital content.” The node which first received the output from the RTMP and WebRTC media sources would operate as an HLS mother server. Ex.1001 ¶425.

Alternatively, this aspect of the claimed invention would have been obvious to a POSA in order to render the server HLS compatible. Ex.1001 ¶¶426–28.

d. Element 26.6

As explained above in §§VIII.C.2.a–b, Herzog as modified by Spivey and Ellis establishes the first video communication channel to provide the first copy of digital content to the first viewer client device. As further explained at §§VIII.D.1.b–c, Herzog as modified by Spivey, Ellis, Kellicker and Phillips establishes that the

first video communication channels provided between the first viewer client device and the HLS CDN.

As explained at §VIII.D.1.c, Ellis as modified by Spivey, Kellicker and Phillips establishes that the first copy of the first broadcaster's live stream of digital content is based on the (first) HLS file suite. Ex.1001 ¶¶429–30.

Thus, Herzog as modified by Spivey, Ellis, Kellicker and Phillips satisfies this element.

IX. SECONDARY CONSIDERATIONS OF NON-OBVIOUSNESS

Petitioner is aware that PO has made conclusory assertions of secondary considerations in its Preliminary Responses filed in the Genius IPRs. However, PO clearly fails to establish a nexus or a presumption of a nexus. Thus, PO's conclusory assertions in this regard are unavailing. Moreover, assuming PO were to present similar arguments with its Preliminary Response in this case, Petitioner would not have the opportunity to respond. Thus, a decision on the adequacy of such evidence at the institution phase without a fully developed record, is impracticable. Ex.1001 ¶¶690. Indeed, the number of “[i]ndependently made, simultaneous inventions, made within a comparatively short space of time, are persuasive” “indicia of obviousness.” *Geo. M. Martin v. Alliance Machine*, 618 F.3d 1294, 1304–1305 (Fed. Cir. 2010) (quotations omitted).

X. THE BOARD SHOULD NOT EXERCISE DISCRETIONARY DENIAL

A. The ‘697 Has Not Been Subject to a Prior Petition by the Current Petitioner

Furthermore, there is no “significant relationship” between the prior Petitioner (Genius) and the present Petitioner (Sportradar). *Videndum Production Solutions, Inc. v. Rotolight Limited*, IPR2023–01218, Paper 12 (April 19, 2024). *See also, Petitioner’s Showing of Good Cause for Parallel Petitions* filed concurrently herewith.

B. *Fintiv*

Petitioner has stipulated to PO that, if this IPR is instituted, Petitioner will not pursue invalidity on the grounds raised here or on any other ground that reasonably could have been raised in this IPR. Ex.1021. Thus, the Board should not exercise discretionary denial. Ex.1013.

XI. CONCLUSION

For the foregoing reasons, Petitioner respectfully requests that the Board institute IPR and cancel the Challenged Claims.

Date: December 14, 2024

Respectfully submitted,

/Scott W. Cummings/

Scott W. Cummings

Reg. No. 41,567

BRYAN CAVE LEIGHTON PAISNER LLP

scott.cummings@bclplaw.com

Alexander Walden

Provisionally recognized PTAB attorney
(affidavit to follow)

BRYAN CAVE LEIGHTON PAISNER LLP

alexander.walden@bclplaw.com

Frank Fabiani

Reg. No. 75,934

BRYAN CAVE LEIGHTON PAISNER LLP

frank.fabiani@bclplaw.com

Attorneys For Petitioner Sportradar AG

CERTIFICATE OF COMPLIANCE

This Petition complies with the type–volume limitations as mandated in 37 C.F.R. §42.24. According to the word processing system used to prepare this document, the brief contains 13,376 (14,000 limit) words.

December 14, 2024

/Scott W. Cummings/

Scott W. Cummings

Reg. No. 41,567

BRYAN CAVE LEIGHTON PAISNER LLP

scott.cummings@bclplaw.com

CERTIFICATE OF SERVICE

In compliance with 37 C.F.R. §§42.105, 42.6(e), the undersigned hereby certifies that a copy of the foregoing Petition, supporting exhibits, and Power of Attorney were served on the 14th day of December 2024, via Federal Express® directed to PO at the correspondence address of record:

149890 – Smith Baluch LLP
376 Boylston St
Suite 401
Boston, MA 02116

A courtesy copy was also served by electronic mail on PO's counsel of record in the Texas litigation:

Britton F. Davis
KING & SPALDING LLP
1401 Lawrence Street
Suite 1900
Denver, CO 80202
(720) 535-2300
bfdavis@kslaw.com

Christopher C. Campbell
KING & SPALDING LLP
1700 Pennsylvania Avenue, NW
Suite 200
Washington, DC 20006
(202) 626-5578
ccampbell@kslaw.com

Ryan A. Schmid
KING & SPALDING LLP
1700 Pennsylvania Avenue, NW
Suite 200
Washington, DC 20006
(202) 626-5578
rschmid@kslaw.com

Matthew D. Wood
KING & SPALDING LLP
500 West 2nd Street
Suite 1800
Austin, TX 78701
(512) 457-2000
mwood@kslaw.com

/Teresa C. Rodriguez/

TABLE OF EXHIBITS⁵

Exhibit No.	Description
Exhibit 1001	Declaration of Dr. Omid Kia, Ph.D.
Exhibit 1002	<i>Curriculum Vitae</i> of Dr. Omid Kia, Ph.D.
Exhibit 1003	U.S. Patent No. 10,805,687
Exhibit 1004	Claim Listing of U.S. Patent No. 10,805,687
Exhibit 1005	Certified File History of U.S. Patent No.10,805,687
Exhibit 1006	U.S. Patent No. 10,425,697
Exhibit 1007	Claim Listing of U.S. Patent No. 10,425,697
Exhibit 1008	Certified File History of U.S. Patent No. 10,425,697
Exhibit 1009	U.S. Patent No. 11,039,218
Exhibit 1010	RESERVED
Exhibit 1011	Certified File History of U.S. Patent No. 11,039,218
Exhibit 1012	RESERVED
Exhibit 1013	Director Vidal Memorandum re: <i>Sotera</i> Stipulations
Exhibit 1014	SportsCastr’s Complaint for Patent Infringement against Genius Sports
Exhibit 1015	U.S. Patent Pub. No. 2015/0163379 (“Herzog”)
Exhibit 1016	U.S. Patent Pub. No. 2014/0229992 (“Ellis”)
Exhibit 1017	U.S. Patent Pub. No. 2016/0036910 (“Spivey”)

⁵ Brackets ([]) are used to refer to the sequential page numbers added to exhibits pursuant to 37 C.F.R. §42.63(d)(2)(i).

Exhibit 1018	U.S. Patent Pub. No. 2015/0163379 (“Mitic”)
Exhibit 1019 – 1020	RESERVED
Exhibit 1021	Genius Sports’ proposed <i>Sotera</i> Stipulation to SportsCastr
Exhibit 1022	James F. Kurose & Keith W. Ross, <i>Application Layer</i> , in <i>Computer Networking a Top–Down Approach</i> (Michael Hirsch <i>et al.</i> eds., 4th ed. 2012) (“Kurose”)
Exhibit 1023	John Dilley <i>et al.</i> , <i>Globally Distributed Content Delivery</i> , IEEE Internet Computing (Sept.–Oct. 2002).
Exhibit 1024	Mário Marques da Silva, <i>Multimedia Communications and Networking</i> (2012) (“Silva”)
Exhibit 1025	<i>A Brief History of the Internet</i> , ONLINE LIB. LEARNING CENTER. (Feb. 9, 2024), https://www.usg.edu/galileo/skills/unit07/internet07_02.phtml (“Online Library”)
Exhibit 1026	Hisayuuki Ohmata <i>et al.</i> , <i>Hybridcast: A new media experience by integration of broadcasting and broadband</i> , 2013 ITU Kaleidoscope: Building Sustainable Communities, April 2013 (“Ohmata”)
Exhibit 1027	Henning Schulzrinne <i>et al.</i> , <i>RTP: A Transport Protocol for Real–Time–Applications</i> , DataTracker (January 1996), https://datatracker.ietf.org/doc/html/rfc1889 (“Schulzrinne”)
Exhibit 1028	Brijesh Kumar, <i>Making Sense of Video Streaming Protocols</i> , LinkedIn. (Aug. 11, 2016), https://www.linkedin.com/pulse/making-sense-video-streaming-protocols-dr-brijesh-kumar/ (“Kumar”)
Exhibit 1029	Ian Fette & Alexey Melnikov, <i>The WebSocket Protocol</i> , RFC Editor. (Dec. 2016), https://www.rfc-editor.org/rfc/rfc6455.html .
Exhibit 1030	Marcin Warczygłowa, <i>Real–time Web Application with Websockets and Vert.X</i> (Nov. 10, 2015), https://blog.allegro.tech/2015/11/real-time-web-application-with-websockets-and-vert-x.html .

Exhibit 1031	Randall Stewart & Chris Metz, <i>SCTP: new transport protocol for TCP/IP</i> , 5 IEEE 64, (2001) (“Stewart”)
Exhibit 1032	Lydon Ong, John Yoakum &, <i>An Introduction to the Stream Control Transmission Protocol (SCTP)</i> , Datatracker. (May. 2002), https://datatracker.ietf.org/doc/rfc3286/ (“Ong”)
Exhibit 1033	Jan Newmarch, <i>Introduction to Stream Control Transmission Protocol</i> , LINUX J. (Sep. 1, 2007), https://www.linuxjournal.com/article/9748 (“Newmarch”)
Exhibit 1034	Kyung-Hoe Kim et al, <i>A transmission control SCTP for real-time multimedia streaming</i> , 54 Computer Networks 1418, (2010) (“Kim”)
Exhibit 1035	High Scalability, <i>How Facebook Live Streams to 800,000 Simultaneous Viewers</i> , ByteByteGo (Jun. 27, 2016), https://blog.bytebytego.com/?utm_source=hs&utm_medium=ann&utm_campaign=top (“ByteGo”)
Exhibit 1036	Hardeep Singh Parmar & Michael C. Thornburgh, <i>Adobe’s Real Time Messaging Protocol</i> , Veriskope. (Dec. 21, 2002) (“Parmer”)
Exhibit 1037	Ilya Grigorik, <i>WebRTC</i> , in High Performance Browser Networking, (2013) (“Grigorik”)
Exhibit 1038	Sam Dutton, <i>Get started with WebRTC</i> , web.dev. (Jul. 23, 2012), https://web.dev/articles/webrtc-basics (“Dutton”)
Exhibit 1039	Andy Salo, <i>MPEG DASH: A Technical Deep Dive and Look at What’s Next</i> , RGB Networks, (2012) (“Salo”)
Exhibit 1040	Jan Ozer, <i>Upstream, Justin.tv, Livestream, and Bambuser: Streaming Unplugged</i> , Streaming Media Magazine., Jun.–Jul. 2012 (“Ozer”)
Exhibit 1041	Christopher Mueller, <i>MEPG–DASH vs. Apple HLS vs. Microsoft Smooth Streaming vs. Adobe HDS</i> , BITMOVIN. (Mar. 29, 2015), https://bitmovin.com/mpeg-dash-vs-apple-hls-vs-microsoft-smooth-streaming-vs-adobe-hds/ (“Mueller”)
Exhibit 1042	Stefan Saroiu et al., <i>An analysis of Internet Content Delivery Systems</i> , ACM SIGOPS Operating System Review, Vol. 36 (Dec. 31, 2002), https://doi.org/10.1145/844128.84415 (“Saroiu”)

Petition for *Inter Partes* Review of U.S. Patent No. 10,425,697

Exhibit 1043	Charles Marion & Julien Jomier, <i>Real-time Collaborative Scientific WebGL Visualization with Websocket</i> , Ass’n Computing Machinery, August 4–5, 2012 (“Marion”)
Exhibit 1044	Mark Sweney, <i>First ads appear on YouTube Clips</i> , The Guardian, Aug. 22, 2007 (“Sweney”)
Exhibit 1045	Richard Sandomir, <i>The Innovation That Grew and Grew</i> , N.Y. Times, June 12, 2014 (“Sandomir”)
Exhibit 1046	Victoria Pimentel & Bradford G. Nickerson, <i>Communicating and Displaying Real-Time Data with WebSocket</i> , 16 IEEE (2012) (“Pimentel”)
Exhibit 1047	Lucian Popa <i>et al.</i> , <i>HTTP as the narrow waist of the future internet</i> , 2010 ACM SIGCOMM Workshop, October 2013 (“Popa”)
Exhibit 1048	Gilstrap Order of September 30, 2024 authorizing addition of Challenged Claims to the Complaint
Exhibit 1049	U.S. Patent Publication No. 2017/0250882 (“Kellicker”)
Exhibit 1050	U.S. Patent Publication No. 2015/0127845 (“Phillips”)
Exhibit 1051	U.S. Patent No. 9,246,965 (“Stoica”)
Exhibit 1052	R. Pantos <i>et al.</i> , “HTTP Live Streaming,” (“Pantos”) published April 4, 2016
Exhibit 1053	Wayback Machine Pantos archive dated April 4, 2016
Exhibit 1054	US Patent No. 9,106,607, (“Lepeska”)
Exhibit 1055	EP2920708 (“Basile”)
Exhibit 1056	U.S. Patent No. 9,973,785 (“Yang”)