

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28

UNITED STATES DISTRICT COURT
NORTHERN DISTRICT OF CALIFORNIA

GOOGLE LLC,

Plaintiff,

No. C 20-06754 WHA

v.

SONOS, INC.,

Defendant.

**ORDER GRANTING
MOTION FOR PARTIAL
SUMMARY JUDGMENT
AS TO '615 PATENT**

INTRODUCTION

In this patent infringement action, the accused infringer moves for summary judgment of non-infringement and invalidity of claim 13 of U.S. Patent No. 9,967,615. To the extent stated below, the motion is **GRANTED**.

STATEMENT

Patent owner Sonos, Inc. alleges that Google LLC's products infringe its patents, including United States Patent Nos. 10,848,885 and 9,967,615. Pursuant to our "patent showdown" procedure (Dkt. Nos. 68, 206), each side moves for summary judgment on one particular claim-in-suit. A separate order granted Sonos's motion for summary judgment of infringement as to claim 1 of the '885 patent and denied Google's corresponding motion of non-infringement (Dkt. No. 309). This order considers Google's motion for summary judgment of non-infringement and invalidity as to claim 13 of the '615 patent.

1 The technology at issue in this case generally concerns multi-room “smart” speaker
2 technology. Whereas the ’885 patent covers technology related to managing groups of smart
3 speakers, the ’615 patent relates to the act of transferring playback of music or other media
4 content from one device (*e.g.*, a smart phone) to another (*e.g.*, a smart speaker). In particular,
5 claim 13 of the ’615 patent is directed towards transferring playback of a *queue* of media
6 content (*e.g.*, a song playlist) from one device to another.

7 Some knowledge of pertinent terminology is helpful. Sonos accuses Google of infringing
8 by equipping “control devices” with certain apps that are capable of transferring media
9 playback to a “playback device.” Control devices are devices such as smart phones or tablets
10 that can install and control apps. Playback devices are devices such a smart speakers or
11 televisions that can play content. Google refers to the act of transferring playback from the
12 control device to the playback device as “casting.” The accused apps employ “cast”
13 technology that enables control devices to transfer media playback to a “cast-enabled”
14 playback device (Opp. 2).

15 The ability to transfer playback is useful because control devices are not necessarily ideal
16 for media playback. Smart phones, for example, have small screens and produce
17 unexceptional audio. Cast technology solves this problem by allowing users to transfer video
18 to external, larger screens and audio to external, higher-quality speakers.

19 Google Play Music, one of the accused apps, is illustrative. The app offers users a library
20 of songs to play. The details are disputed, but, generally, the app has access to information
21 about a song track currently being played, the tracks that were played previously, and the
22 tracks that are scheduled to play in the future. Such an arrangement of songs (or other content)
23 is commonly referred to as a “queue.” Among other purposes, the app’s access to the queue
24 allows users to skip forward to the next song or skip backward to the previous song.

25 If a Google Play Music user is disgruntled by the smart phone’s speaker and wants to
26 transfer audio playback to a smart speaker, the user can activate a feature on the app to cast
27 from the former to the latter. In addition to transferring playback of the current song, the smart
28 phone also transfers access to the *queue* of songs. Then, once playback has been transferred,

1 our user can control playback through the smart phone. Our user can, for example, skip to the
2 next song in the queue, go back to the previous song in the queue, or shuffle the music in the
3 queue, all while the music is playing on the smart speaker.

4 Sonos also accuses various of Google’s YouTube apps of infringing, including YouTube,
5 YouTube Music, YouTube Kids, and YouTube TV. The YouTube apps work similarly to
6 Google Play Music, except those apps allow *videos* (and accompanying audio) to be cast to
7 another device such as a smart television. A user can, for example, install the YouTube app on
8 a smart phone, play a YouTube video on the phone, and then cast the video, even midstream,
9 to a cast-enabled television. The video would then play on the television, the television would
10 have access to the queue of videos, and the user would be able to control playback through the
11 phone.

12 Claim 13 of the ’615 patent is directed toward “systems, methods, apparatus, and articles
13 of manufacture” to facilitate the transfer of playback from a “control device” to a “playback
14 device” (’615 patent at Abstract). Using Google’s paragraph numbering, claim 13 of the ’615
15 patent recites:

16 **13[pre]**. A tangible, non-transitory computer readable storage
17 medium including instructions for execution by a processor, the
18 instructions, when executed, cause a control device to implement a
method comprising:

19 **13.1** causing a graphical interface to display a control interface
20 including one or more transport controls to control playback by the
control device;

21 **13.2** after connecting to a local area network via a network interface,
identifying playback devices connected to the local area network;

22 **13.3** causing the graphical interface to display a selectable option for
23 transferring playback from the control device;

24 **13.4** detecting *a set of inputs to transfer* playback from the control
25 device to a *particular* playback device, wherein the set of inputs
26 comprises: (i) *a selection of the selectable option for transferring*
playback from the control device and (ii) *a selection of the particular*
playback device from the identified playback devices connected to the
local area network;

27 **13.5** after detecting the set of inputs to *transfer playback* from the
28 control device to the *particular* playback device, causing playback to
be transferred from the control device to the particular playback

1 device, wherein transferring playback from the control device to the
particular playback device comprises:

2 (a) causing one or more first cloud servers to add multimedia
3 content to a *local playback queue* on the particular playback
4 device, wherein adding the multimedia content to the local
5 playback queue comprises the one or more first cloud servers
6 adding, to the local playback queue, one or more resource
locators corresponding to respective locations of the
multimedia content at one or more second cloud servers of a
streaming content service;

7 (b) causing playback at the control device to be *stopped*; and

8 (c) modifying the one or more transport controls of the control
interface to control playback by the playback device; and

9 **13.6** causing the particular playback device to play back the
10 multimedia content, wherein the particular playback device playing
back the multimedia content comprises the particular playback device
11 retrieving the multimedia content from one or more second cloud
servers of a streaming content service and playing back the retrieved
12 multimedia content.

13 The most contested terms are italicized. Sonos filed the application that led to the '615 patent
14 in 2018, but the patent application claims priority through a chain of applications dating back
15 to December 30, 2011. As detailed further below, the parties dispute the date of conception.
16 Sonos asserts July 15, 2011, as the invention date.

17 Google argues that its products do not infringe element 13.5(a) because the accused apps
18 employ a *remote* playback queue as opposed to a *local* playback queue. Google further
19 contends that a 2010 version of the YouTube Remote app either anticipated claim 13 or, when
20 combined with other references, rendered it obvious. This order follows full briefing and oral
21 argument.

22 ANALYSIS

23 Summary judgment is proper when there is no genuine dispute of material fact and the
24 moving party is entitled to judgment as a matter of law. FRCP 56(a). A genuine dispute of
25 material fact is one that “might affect the outcome of the suit under the governing law.”
26 *Anderson v. Liberty Lobby, Inc.*, 477 U.S. 242, 247–48 (1986). In deciding a motion for
27 summary judgment, the court must accept the non-movant’s non-conclusory evidence and
28 draw all justifiable inferences in its favor. *Id.* at 255.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28

1. NON-INFRINGEMENT.

This order starts with Google’s non-infringement arguments. Analysis of patent infringement requires a claim to be properly construed to determine its scope and meaning, which is then compared to the accused device or process. *See Tessera, Inc. v. Int’l Trade Comm’n*, 646 F.3d 1357, 1364 (Fed. Cir. 2011); *Carroll Touch, Inc. v. Electro Mech. Sys., Inc.*, 15 F.3d 1573, 1576 (Fed. Cir. 1993). Accordingly, this order will first construe the disputed term to determine claim 13’s scope and then proceed to assess whether the properly construed claim reads on Google’s accused products.

A. CONSTRUCTION OF “PLAYBACK QUEUE”

| Sonos’s Proposed Construction | Google’s Proposed Construction | Court’s Construction |
|-------------------------------|---|--|
| Plain and ordinary meaning | “An ordered list of multimedia items that is selected by the user for playback” | “A list of multimedia content selected for playback” |

Claim terms generally take “their ordinary and customary meaning,” that is “the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention.” *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312–13 (Fed. Cir. 2005) (en banc). Although construction begins with the claim language itself, “*the specification is the single best*” — and usually dispositive — “guide to the meaning of a disputed term.” *Network-1 Techs., Inc. v. Hewlett-Packard Co.*, 981 F.3d 1015, 1022 (Fed. Cir. 2020) (quoting *Phillips*, 415 F.3d at 1314–15) (emphasis added).

Here, the only pertinent term in dispute is “playback queue.”¹ As explained above, part of claim 13 relates to transferring a queue of content from one device to another. The relevant portion of limitation 13.5 recites:

after detecting the set of inputs to transfer playback from the control device to the particular playback device, causing playback to be transferred from the control device to the particular playback device, wherein transferring playback from the control device to the

¹ The parties’ dispute over the term “resource locator” does not bear on this order’s infringement and invalidity analysis.

1 particular playback device comprises:

2 (a) causing one or more first cloud servers to add multimedia
3 content to a *local playback queue* on the particular playback
4 device. . .

5 (emphasis added). The parties fiercely dispute whether the accused products use a “local
6 playback queue.” As detailed further below, Google’s cast technology currently manages
7 content queues, broadly speaking, by storing such a queue on a *remote* cloud server on the
8 internet. The parties refer to this remote queue as a cloud queue (*see, e.g.*, Br. 1). The parties
9 agree that the cloud queue is not a “local playback queue,” as required by limitation 13.5(a),
10 because it’s stored remotely on the internet as opposed to being stored locally on the playback
11 device.

12 In order to play music or other content, however, a Google cast-enabled playback device
13 receives some information from a cloud queue, which it subsequently stores locally. The
14 details are disputed, but, as an example, a cast-enabled smart speaker might receive
15 information from the cloud queue about the current song being played, the next song scheduled
16 to play (in case the user wants to skip the next song), and the song that was played before (in
17 case the user wants to go back to the previous song). This is the extent of what the smart
18 speaker knows. The smart speaker won’t, for example, know about the next ten songs
19 scheduled to play, or the ten songs that played before. That more extensive information is only
20 stored in the cloud queue.

21 At stake here is whether the locally-stored information about the previous, current, and
22 next song might *also* be a playback queue. If it is, then it could be the kind of “local playback
23 queue” necessary to infringe. Sonos accordingly advocates against associating the term
24 “playback queue” with any specific requirements, while Google wants a more definite
25 construction to serve as the foundation for its non-infringement arguments. In particular, the
26 parties dispute whether the term “playback queue” requires: (1) an “ordered list”; (2) plural
27 “multimedia items”; and (3) user-selected media.

28 *First*, Sonos asserts that a “playback queue” need not be a list, but this argument does not
conform with the intrinsic evidence. The patent repeatedly associates a queue with a “list” or

1 “playlist.” *See, e.g.*, ’615 patent at 15:57–67 (playback device may have information about “a
2 current play position *within a list* to enable near-seamless ‘handoff’ of music from a portable
3 device to a local playback system” (emphasis added)); 16:32–35 (devices may share “a current
4 point of playback (e.g., now playing a third song in a playlist, fourth song in the playlist, and
5 so on)”).

6 Sonos objects that “the ’615 Patent teaches embodiments where a ‘playback device’
7 queues a *single* resource locator, such as a URL. . . .” (Claim Constr. Br. 12). But a list of one
8 is still a list. The patent cites a publication that states, for instance, that a “media *listing* can
9 include . . . *one or more* additional items of media content.” *See* U.S. Patent App. Publ.
10 2012/0089910 A1 at ¶ 52 (emphasis added); *see also* *FitBit Inc. v. AliphCom*, 2017 WL
11 386257, at *14 (N.D. Cal. Jan. 27, 2017) (Judge Edward J. Davila) (“The ordinary meaning of
12 ‘list’ also supports the idea that the ‘list’ at issue can contain one . . . item[.]. Lists often . . .
13 contain only one item.”).

14 *Second*, Sonos argues that nothing requires a “playback queue” to contain plural
15 multimedia items. This order agrees on that point. The plain language of the claim recites:
16 “adding the multimedia content to the local playback queue comprises . . . adding, to the local
17 playback queue, *one or more resource locators*” (*see* limitation 13.5(a) (emphasis added)).
18 Google objects that a queue must include a “next” media item in case the user wants to skip
19 forward, and accordingly argues that a queue logically must include at least two items (Br. 9
20 (citing Bhattacharjee Decl. ¶¶ 71–73)). But the patent does not have such a restrictive view of
21 a queue. In addition to the claim language cited above, the specification repeatedly describes
22 embodiments where a queue only contains a single audio track. *See, e.g.*, ’615 patent at 11:62–
23 12:3 (a smart speaker “may contain a uniform resource locator . . . that specifies an address to
24 *a particular audio track* in the cloud” (emphasis added)); *see also id.* at 10:42–46; 12:49–63;
25 13:36–40. These citations suggest that the list must contain at least one item, but not
26 necessarily more than one.

27 *Third*, Sonos asserts that the content in the queue need not be selected directly by a user.
28 Google’s position, by contrast, is that a user must directly populate and manage the queue (CC

1 Opp. 11). Google’s argument does not persuade. True, the specification discusses scenarios
2 where a user adds or deletes content from the queue and suggests that a user *can* edit a queue
3 (*see, e.g.*, ’615 patent at 16:25–31 (describing a “queue that the user is editing/managing. . .”).
4 However, the specification also repeatedly describes embodiments in which the third-party
5 application (such as Google Play Music) dictates what media content is in the queue. *See, e.g.*,
6 ’615 patent at 13:1–10; 15:59–62. Moreover, as Sonos points out, nothing in the claim itself
7 refers to a user.

8 In sum, this order agrees with Google that a “playback queue” requires a “list” of content
9 selected for playback, but agrees with Sonos that the list does not necessarily require more than
10 one item of content or require users to select content directly. This order further rejects
11 Google’s proposal to include the term “multimedia item” in the construction. The claim uses
12 the term “multimedia content,” and there is no need to introduce additional ambiguity by
13 importing a new term. Accordingly, this order construes the term “playback queue” as “a list
14 of multimedia content selected for playback.”

15 **B. THE ACCUSED APPS DO NOT USE A “LOCAL PLAYBACK QUEUE”**

16 Having construed “playback queue,” we now turn to Google’s non-infringement
17 arguments. To prove infringement, Sonos must show that Google’s accused products meet
18 each properly construed limitation of claim 13 either literally or under the doctrine of
19 equivalents. *See Deering Precision Instruments, LLC v. Vector Distribution Sys., Inc.*, 347
20 F.3d 1314, 1324 (Fed. Cir. 2003). Here, Sonos asserts both. To establish literal infringement,
21 all of the elements of the claim, as correctly construed, must be present in the accused
22 products. *TechSearch, LLC v. Intel Corp.*, 286 F.3d 1360, 1371 (Fed. Cir. 2002). Sonos may
23 also establish infringement under the doctrine of equivalents by “showing that the difference
24 between the claimed invention and the accused product [is] insubstantial,” including “by
25 showing on a limitation by limitation basis that the accused product performs substantially the
26 same function in substantially the same way with substantially the same result as each claim
27 limitation of the patented product.” *Crown Packaging Tech., Inc. v. Rexam Beverage Can Co.*,
28 559 F.3d 1308, 1312 (Fed. Cir. 2009).

1 As stated above, Google’s primary non-infringement theory is that both its Google Play
2 Music app and YouTube app products do not use a “local playback queue” (*see* limitation
3 13.5). The two categories of accused apps operate in slightly different ways. In plain Greek,
4 Sonos asserts that using the cast feature on the YouTube apps on a control device causes a
5 “WatchNext” server to transmit a “WatchNextResponse” to a cast-enabled playback device.
6 The WatchNextResponse is then stored by the playback device. According to Sonos, the
7 WatchNextResponse “often” includes a string of characters called a “videoId” that corresponds
8 to the item set to currently playback, the item set to playback next, and the item that came
9 before the current item (Opp. 3–5). Similarly, casting from the Google Play Music app on the
10 control device creates an ItemWindowResponse that stores a link to the previous, current, and
11 next media items set for playback (Opp. 7-8). The apps receive this information from a cloud
12 queue, and neither of the apps store any additional media content beyond those three items.

13 Google quibbles about some of the details. Google asserts, for example, that a playback
14 device only “request[s] cloud queue items one-by-one” (Br. 7). Google further states that the
15 “upNextVideoID” variable (*i.e.*, the variable corresponding to the item set to play next) only
16 appears when “the cloud queue has been exhausted,” and that this variable “only exists for a
17 few milliseconds” (Br. 9).

18 At bottom, though, neither side appears to dispute that Google’s products operate by
19 “retrieving” information from the cloud queue about the current, next, and previous media item
20 (Br. 9; Opp. 3). Nor do the parties dispute that these three items are only a subset of a separate
21 cloud queue. *The focus of the dispute is instead on whether this information stored locally in*
22 *the playback device is a playback queue at all.*

23 The parties dedicated the bulk of their briefing and their time at oral argument to this
24 issue. Upon review, this order concludes that neither the information stored by the
25 WatchNextResponse (in the YouTube apps) nor the information stored by the
26 ItemWindowResponse (in the Google Play Music app) qualify as a playback queue. The
27 groups of three items stored by the respective apps are not lists of multimedia content selected
28 for playback. In each app, the cloud queue stores the list, and the locally-stored information is

1 merely a mirror reflecting a subset of what is happening in the cloud queue. The songs set to
2 play on Google Play Music, for example, are all dictated by the cloud queue. If the user adds
3 or edits a playlist, the cloud queue changes. If the app creates a playlist, the cloud queue
4 adapts. It is only *after* the cloud queue changes that anything can happen to the information
5 stored locally on the playback device (*see* Bhattacharjee Decl. ¶¶ 81–84). This demonstrates
6 that the groups of three items stored in each app are not lists of content selected for playback,
7 but rather merely provide the means to *process* the lists for playback. In short, the cloud queue
8 runs the show.

9 Sonos objects that multiple playback queues can exist simultaneously (Opp. 8). In
10 support, Sonos points out that the specification teaches that there can be “two-way
11 communication” between the local playback queue and a separate queue, “such as keeping a
12 local playback queue synchronized with a queue that the user is editing/managing in the third
13 party application” (’615 patent at 16:22–31). But there is no such “two-way communication”
14 here. Rather, the cloud queue delivers information to the playback device on a one-way street.
15 The cloud queue provides information about the queue to the WatchNextResponse and
16 ItemWindowResponse, and never vice-versa, because there is no locally-stored queue that
17 would allow “two-way” synchronization.

18 Sonos further objects that the specification teaches that “the local playback system” can
19 “periodically fetch[] a short list of tracks to play list” from a “third-party application” (*id.* at
20 16:63–66). This passage, however, explains that the third-party application can “*override* a
21 local playback queue” with the “short list of tracks to play next” (*ibid.*). The passage thus
22 distinguishes a local playback queue from the “short list of tracks.” Moreover, the passage
23 suggests that there must be something to override, and, here, the locally-stored information can
24 never be populated by anything other than the “short list of tracks.” Instead,
25 WatchNextResponse and ItemWindowResponse can never store additional or different items,
26 and never more than three total items (*see* Bhatarcharjee Decl. ¶¶ 21, 73, 119).

27 Sonos has accordingly failed to raise a genuine dispute that Google’s products employ a
28 “local playback queue” as contemplated by claim 13 of the ’615 patent. Thus, Google’s

1 products do not infringe the claim, and this order need not address Google’s additional non-
2 infringement arguments.

3 **2. ANTICIPATION AND OBVIOUSNESS.**

4 This order now turns to Google’s invalidity arguments. Before we move forward,
5 however, a procedural note. After Google filed its motion, Sonos moved to strike some of the
6 motion for improperly asserting new invalidity theories and prior art references. A prior order
7 granted Sonos’s motion in part, striking paragraphs 133 and 138–41 of Dr. Bhattacharjee’s
8 expert report in their entirety (Dkt. No. 315). That material is accordingly not considered here.

9 With this in mind, we now turn to Google’s surviving arguments. Google asserts that
10 one of its previous apps, the YouTube Remote app, anticipated claim 13 of the ’615 patent.
11 Alternatively, Google argues that it would have been obvious to combine the YouTube Remote
12 app with other prior art references to achieve Sonos’s claimed invention.

13 The YouTube Remote app was released in November 2010 (*see* Bobohalma Decl. ¶ 3).
14 The purpose of the app was to allow a smart phone to connect to another device, such as a
15 television or computer, so that a YouTube video being played on the smart phone would
16 appear on a larger screen (Br. 16). For example, a user could mirror a YouTube video onto a
17 television and control playback of the video through the app. In short, the app functioned
18 similarly to what Google now calls casting, as described above. The old process was more
19 cumbersome, however, because it involved an intermediary website to which both devices
20 separately had to connect to enable pairing. Specifically, each device needed to separately
21 navigate to the intermediary website and log in to the same YouTube account. Once the
22 devices were logged in to the same account, they could then pair with each other through the
23 YouTube Remote app (*see* Opp. 14–15).

24 Google now asserts that the app disclosed each and every limitation of claim 13.
25 “Anticipation requires that a single prior art reference disclose each and every limitation of the
26 claimed invention, either expressly or inherently.” *SRI Int’l Inc. v. Cisco Sys., Inc.*, 930 F.3d
27 1295, 1306 (Fed. Cir. 2019). “Anticipation is a question of fact.” *Atlas Powder Co. v. Ireco,*
28 *Inc.*, 190 F.3d 1342, 1346 (Fed. Cir. 1999). Because a patent is presumed valid, the party

1 asserting invalidity has the burden of proof to show anticipation by clear and convincing
2 evidence. *Core Wireless Licensing S.A.R.L. v. LG Elecs., Inc.*, 880 F.3d 1356, 1364 (Fed. Cir.
3 2018).

4 Sonos replies that the YouTube remote app did not disclose claim limitations 13.2, 13.4,
5 13.5, and 13.6.² This order will first consider Sonos’s arguments as to 13.2, 13.5, and 13.6,
6 and then circle back to 13.4. For the reasons that follow, this order concludes that the app did
7 not disclose limitation 13.4, but that modifying the app the satisfy the limitation would have
8 been obvious in light of the prior art.

9 **A. LIMITATION 13.2**

10 Claim limitation 13.2 recites “after connecting to a local area network via a network
11 interface, identifying playback devices connected to the local area network.” For our purposes,
12 a “local area network,” or LAN, refers to a “local” home Wi-Fi network. This is in contrast to
13 a “wide area network,” or WAN, which refers to network that can be accessed more widely,
14 *e.g.*, a 3G cellular network. The important point is that a LAN and a WAN provide different
15 means to connect a device to the internet.

16 Sonos reads limitation 13.2 to require that the control device affirmatively identify that
17 the playback device is *connected to the same LAN* as the control device, as opposed to a WAN
18 or a different LAN (Opp. 15). Google admits that its system did not do this. Instead, the
19 system allowed the control device and the playback device to separately log in to the
20 intermediary website. As such, it didn’t matter whether the smart phone and television were
21 connected to the internet in different ways. For example, the smart phone could have used a
22 3G network (a WAN) to navigate to the intermediary website, while the television could have
23 used a home Wi-Fi network (a LAN) to navigate to the intermediary website. Thus, the
24 YouTube Remote system did not require the control device to know whether the television was
25 connected to a LAN at all. Sonos insists that this precludes anticipation.

26
27 ² Sonos briefly argues that Google has not met its burden to show anticipation as to limitations
28 13.1 and 13.3 (Opp. 14). But Google stated in its motion that, according to Sonos’s validity
contentions, there was no dispute as those limitations (*see* Br. 17–18). Sonos said nothing to the
contrary in its opposition brief.

1 Google objects that the plain language of the claim does not require any such identifying
 2 of the LAN. This order agrees. The claim only requires the control device to “identify[]
 3 playback devices connected to the local area network” (*see* limitation 13.2). The phrase
 4 “connected to the local area network” modifies “playback devices.” Accordingly, the claim
 5 does not require affirmatively identifying the LAN. Nor does it require identifying *that* the
 6 television is connected to the LAN. All that is required is that the system allowed (i) playback
 7 devices to be identified and (ii) such identified devices to be connected to the same LAN as the
 8 control device.

9 The YouTube Remote system allowed both. It is undisputed that a television, for
 10 example, could have been connected to the same home Wi-Fi network as the smart phone, and
 11 that the smart phone could have transferred playback to that television. Indeed, Google has
 12 provided a 2010 video demonstrating just that (*see* Reply Br. 10).³ In such circumstances, the
 13 control device is “identifying playback devices connected to the [LAN]” (*see* limitation 13.2).
 14 True, as Google acknowledges, the phone and the television could have been connected to the
 15 internet in different ways, but that Google’s system could identify devices connected to the
 16 LAN in some circumstances *and* devices not connected to the LAN in others does not preclude
 17 anticipation. Rather, it shows that Google’s system was flexible and had capabilities in
 18 addition to those recited by the claim. *See Vulcan Eng'g Co. v. Fata Aluminium, Inc.*, 278 F.3d
 19 1366, 1375 (Fed. Cir. 2002) (“It is irrelevant whether an element has capabilities in addition to
 20 that stated in the claim.”). Accordingly, the YouTube Remote app system tracked limitation
 21 13.2.

22 **B. LIMITATIONS 13.5–13.6**

23 Next, Sonos briefly argues that the YouTube Remote system did not disclose limitation
 24 13.5 because the app did not “caus[e] the playback at the control device to be *stopped*” (Opp.
 25 18). Instead, Sonos contends, the smart phone in the 2010 video proffered by Google “appears
 26 to still be in a playback state (albeit paused). . . .” (*ibid.*). Sonos, however, does not elaborate
 27

28 ³ Citing *How to Control Google TV or YouTube Leanback with YouTube Remote*, YouTube
 (Nov. 14, 2010), available at <https://youtu.be/EGdsOslqG2s?t=56> (last visited July 29, 2022).

1 as to the distinction between playback being “paused” and “stopped,” and presents no evidence
2 that there is any difference between the two. Absent such, this order concludes that they mean
3 the same thing. Thus, the YouTube Remote app tracked limitation 13.5 to that extent.

4 Sonos goes on to argue that Google has failed to show disclosure as to both limitations
5 13.5 and 13.6 because Google’s arguments as to those limitations rely on an “‘API’ document
6 and/or source code” that is dated July 12, 2010. The version of the YouTube Remote system
7 that Google asserts as anticipating prior art is dated November 9, 2010. To link the API
8 document with the later version of the YouTube Remote app, Google provided a declaration
9 from an engineer who did not work at Google until July 2011, a year after the date ascribed to
10 the API document (Reply Br. 18; Levai Decl. ¶¶ 2, 9). Sonos asserts that such “an
11 uncorroborated 2022 declaration of an interested witness . . . who did not even work at
12 Google” at the time “is insufficient corroboration” (Opp. 18). Not so. The engineer, Janos
13 Levai, worked on the prior art product shortly after the relevant time period and then worked
14 on subsequent iterations of the product for five years (*see* Levai Decl. ¶¶ 2, 9). He can testify
15 about how the app worked. Consequently, Sonos has failed to show a genuine dispute as to
16 limitations 13.5 and 13.6.

17 **C. LIMITATION 13.4**

18 We now circle back to limitation 13.4, which is our only remaining disputed limitation.
19 That limitation recites a “set of inputs” that comprise “(i) a selection of the selectable option
20 for transferring playback from the control device and (ii) a selection of the particular playback
21 device from the identified playback devices connected to the local area network.” The
22 YouTube Remote app offered the user the option to select a “Connect” button to transfer
23 playback from the smart phone to the playback devices available for pairing (Br. 19; Opp. 16–
24 17). Sonos does not dispute that a selection of “Connect” was “a selection of the selectable
25 option for transferring playback from the control device” (Opp. 17). Sonos contends, however,
26 that the app did not allow the selection of a “*particular* playback device from the identified
27 playback devices connected to the local area network” (*ibid.*). In other words, Sonos asserts
28 that, in the event multiple devices were available for playback (*e.g.*, multiple televisions), the

1 user had no ability to select a subset of those devices for playback. The user was instead
2 forced to transfer playback to *all* available devices (Schmidt Decl. ¶ 154).

3 In reply, Google points to YouTube Remote source code dated December 1, 2011 —
4 which pre-dates the '615 patent's claimed priority date of December 30, 2011 — that it asserts
5 allowed users to select particular devices (Reply Br. 11–12 (citing Bhattacharjee Decl. ¶
6 170)). Sonos does not dispute the substance of the code, but contends that it is not prior art
7 because its own asserted priority date is July 15, 2011 (*see, e.g.*, Opp. 18 n.8). Sonos further
8 asserts that Google never contested this earlier invention date in its motion. Google replies that
9 it stated in its invalidity contentions that it was Sonos's burden to show the earlier priority date,
10 and that it was only using the July 15 date to frame its arguments (Reply Br. 12).

11 This order sides with Sonos here. Google acknowledged the July 15 date in its opening
12 motion (*see* Br. 19 n.8). Further, Google's motion only discusses the December 1, 2011,
13 source code in the context of its obviousness argument (*see* Bhattacharjee Decl. ¶ 170). Taken
14 together, this shows that Google treated July 15 as the applicable priority date. If Google
15 wanted to rely on the December 1 source code for its anticipation argument, it should have
16 made that clear in its opening motion. Instead, Google engaged in a bait-and-switch in its
17 reply brief (*compare* Br. 19–20, *with* Reply Br. 11–12). Thus, Google cannot now rely on the
18 December 1, 2011, source code for its anticipation case. The YouTube Remote system,
19 consequently, did not disclose limitation 13.4 because Google has failed to show that it
20 allowed users to select a “particular playback device.”

21 Nevertheless, this order concludes that it would have been obvious to combine the
22 YouTube Remote app system with disclosures in United States Patent No. 9,490,998 to allow
23 the selection of individual devices.

24 A claimed invention is obvious if “the differences between the subject matter sought to
25 be patented and the prior art are such that the subject matter as a whole would have been
26 obvious at the time the invention was made to a person having ordinary skill in the art.” 35
27 U.S.C. § 103(a) (pre-AIA). Unlike anticipation, which “requires all elements of a claim to be
28 disclosed within a single reference,” “[o]bviousness can be proven by combining existing prior

1 art references” to disclose all the elements of a claim. *Cohesive Techs. Inc. v. Waters Corp.*,
2 543 F.3d 1351, 1364 (Fed. Cir. 2008).

3 The ’998 patent is prior art. It was filed on March 7, 2011, and claims priority to an
4 earlier provision application filed in November 2010. The patent’s inventors were involved
5 with the development of the YouTube Remote system, and the patent relates to controlling
6 playback on a playback device through a control device. The ’998 patent disclosed that

7 [i]n some examples, the user may also utilize the remote control
8 application of remote control 75 to *select one or more previously*
9 *paired controlled devices*, and to send control messages to one or
10 more paired controlled devices. For example, the user may interact
11 with user interface 84 and/or display 88 to interact with and control
12 any available controlled devices.

13 (see ’998 patent at 10:62–11:6 (emphasis added)). Thus, the patent disclosed that a “user
14 interface” of a “remote control” (e.g., a smart phone) can display “previously paired controlled
15 devices” (e.g., a television) so that a user may select and control “one or more paired
16 controlled devices” (*ibid.*) The patent, therefore, taught the “selection of the particular
17 playback device from the identified playback devices” as contemplated by the ’615 patent.

18 Sonos raises three objections to this conclusion. *First*, Sonos argues that the passage is
19 ambiguous insofar as it could be read to refer to

20 the ability to control one “controlled device,” if that is the only
21 “previously paired” “controlled device,” or the ability to control all
22 “controlled devices” collectively, if multiple “controlled devices”
23 have been “previously paired” with a “remote control” in a session

24 (see Schmidt Decl. ¶ 173). Put differently, Sonos does not read the passage to teach the
25 selection of a particular device when multiple devices are connected in a session. This
26 contorted interpretation does not convince. The most straightforward reading of the passage is
27 that it disclosed the ability to “select one or more” devices among the “previously-paired
28 devices.”

29 *Second*, Sonos argues that the passage does not “mention or even suggest[] *transferring*
30 playback from a ‘remote control’ to a ‘controlled device’” and “[c]onsequently . . . does not
31 teach the selection of a particular ‘controlled device’ to *transfer* playback to” (Opp. 19). This
32 is missing the forest for the trees. As described above, the YouTube Remote app system,

1 which is prior art, disclosed the transfer of playback. The passage from the '998 patent
2 disclosed the selection of a particular device. They achieve the claim together.

3 *Third*, Sonos argues that it would not have been obvious to combine the '998 patent's
4 disclosure with the YouTube Remote system. Specifically, Sonos argues that the YouTube
5 Remote "system architecture" both "teaches away from Google's proposed modifications and
6 also renders such modifications more complicated than Google posits" (Opp. 21–22). Sonos
7 further argues that there would have been no motivation to integrate the modification because
8 it "was not even a prominent feature" of the YouTube Remote app (*id.* at 22). The problem
9 with these assertions is that Google produced source code achieving the proposed modification
10 just a few months after Sonos's asserted priority date, and Google then released the
11 functionality to the public a year later (*see* Reply Br. 14). This adequately demonstrates that a
12 person of skill in art would have been motivated to add the feature.

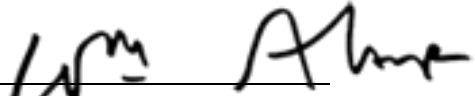
13 Sonos's remaining snippets of argument are conclusory and without merit. In sum, this
14 order finds that it would have been obvious to combine the teachings of the '998 patent with
15 the YouTube Remote system to achieve the claimed invention. Google's motion for summary
16 judgment of invalidity of claim 13 of the '615 patent is accordingly **GRANTED**.

17 **CONCLUSION**

18 To the foregoing extent, Google's motion for summary judgment is **GRANTED**.

19 **IT IS SO ORDERED.**

20
21 Dated: August 2, 2022.

22
23 
24 _____
25 WILLIAM ALSUP
26 UNITED STATES DISTRICT
27 JUDGE
28