

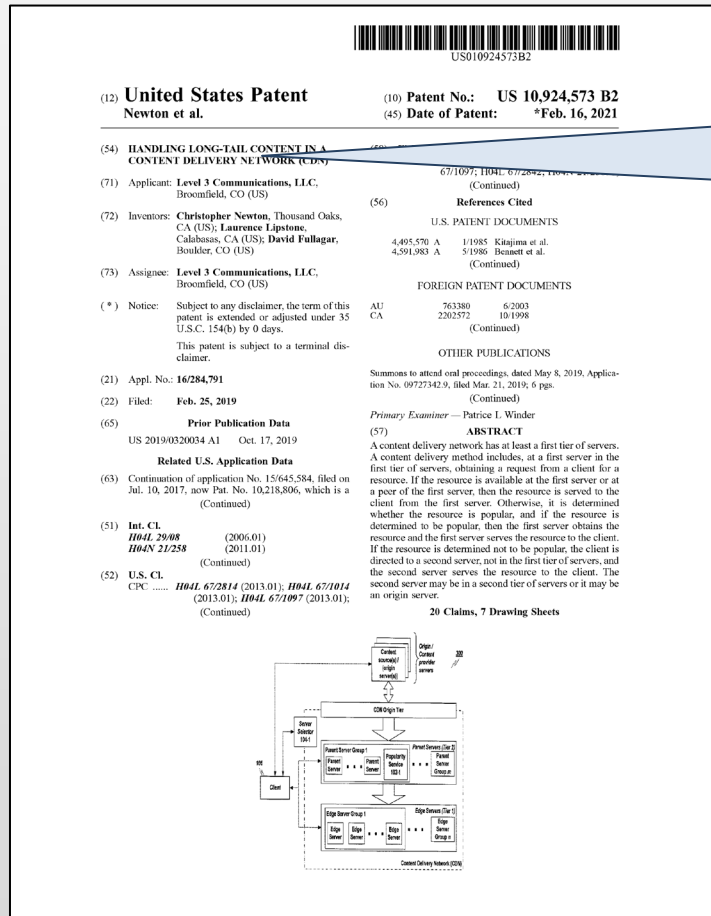
Petitioner's Demonstratives

Google LLC
v.
Sandpiper CDN, LLC

IPR2025-00860
U.S. Patent No. 10,924,573

August 26, 2026

The '573 Patent



HANDLING LONG – TAIL CONTENT IN A CONTENT DELIVERY NETWORK (CDN)

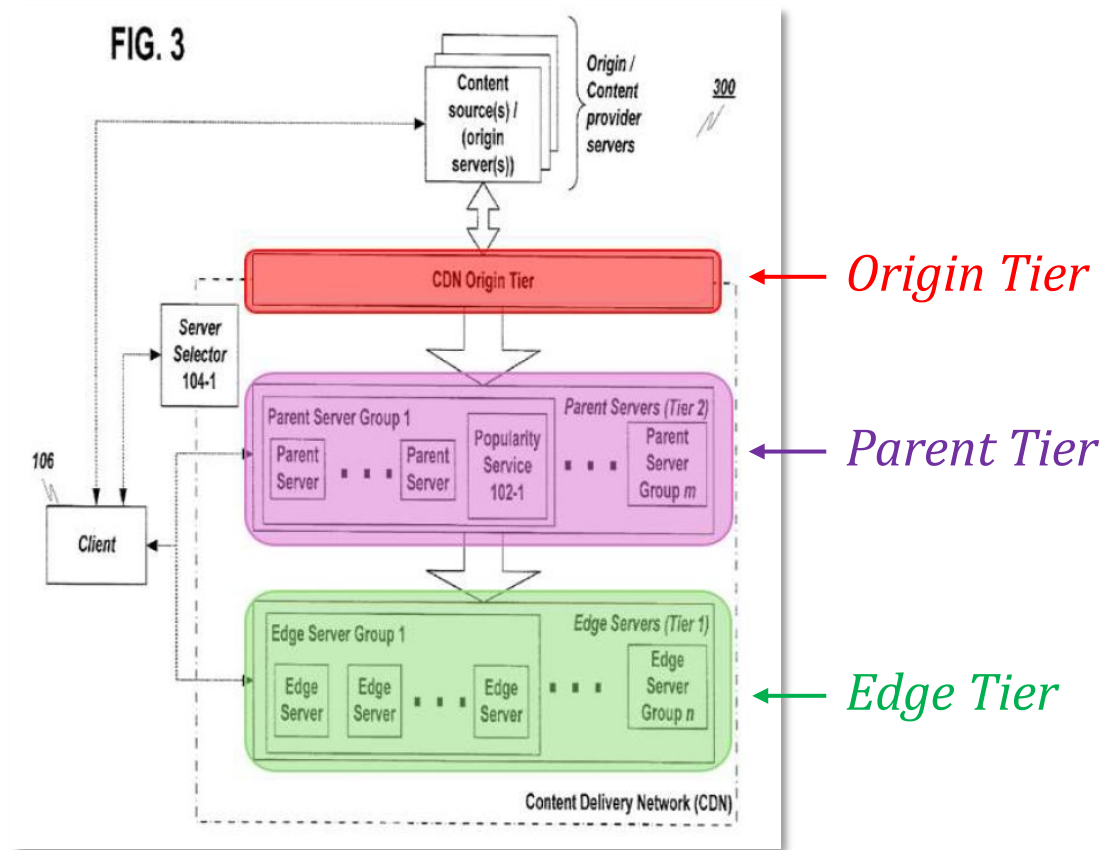
Earliest priority claim: April 4, 2008
Actual priority date: February 25, 2019

20 Claims
(1, 11, and 19 independent)

Ex. 1001, Cover

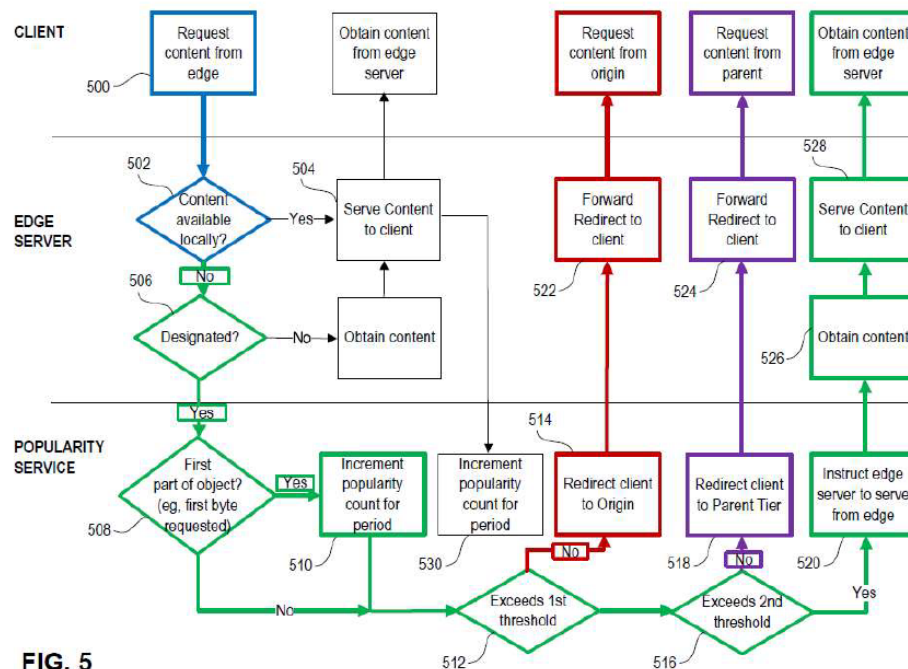
The '573 Patent

The '573 Patent describes a CDN organized into hierarchical tiers with a popularity service.



EX-1001, 3:53-57, 4:19-31; Pet., 2-3; EX-1006, ¶¶ 41-43.

The '573 Patent



If object has not reached first level of popularity (512): popularity service sends edge server instruction to redirect client's request to origin server (or to CDN origin cache) (514).

If object exceeds first level of popularity but not second, mid-tier threshold (516): popularity service sends edge server instruction to redirect client's request to parent server (518).

If object exceeds mid-tier threshold (is popular): popularity service sends edge server instruction to serve content itself (520) along with "follow me" flag to obtain and cache the resource.

EX-1001, Fig. 5, 6:65-7:30; Pet., 5-6 ; EX-1006, ¶¶44-46.

Representative Claim 1

Element	Claim 1
1(Pre)	A method of content delivery in a content delivery network comprising:
1(a)	receiving, at a first server of a first tier of servers of the content delivery network, a request from a requesting device for a resource available from the content delivery network;
1(b)	accessing a popularity service associated with the content delivery network to determine a popularity designation associated with the requested resource;
1(c)	requesting the resource from a second server of the content delivery network;
1(d)	processing, at the first server of the first tier of servers, a redirect instruction from the second server of the content delivery network to obtain the resource from a content server of the content delivery network;
1(e)	receiving an instruction to not cache the portion of the resource at the first server of the first tier of servers when the portion of the resource is obtained from the content server of the content delivery network; and
1(f)	providing the obtained resource to the requesting device.

Grounds of Unpatentability

Ground	Claims Challenged	Basis	References
1	1-20	§ 102	<i>Newton-471</i>
2	1-20	§ 103	<i>Dilley, Pai, and Wang</i>

Ground 1

Newton-471 Anticipates Claims 1-20

Undisputed: Newton-471 Discloses All Limitations

Element	Representative Claim 1	Disputed?
1(Pre)	A method of content delivery in a content delivery network comprising:	No
1(a)	receiving, at a first server of a first tier of servers of the content delivery network, a request from a requesting device for a resource available from the content delivery network;	No
1(b)	accessing a popularity service associated with the content delivery network to determine a popularity designation associated with the requested resource;	No
1(c)	requesting the resource from a second server of the content delivery network;	No
1(d)	processing, at the first server of the first tier of servers, a redirect instruction from the second server of the content delivery network to obtain the resource from a content server of the content delivery network;	No
1(e)	<u>receiving an instruction to not cache</u> the portion of the resource at the first server of the first tier of servers when the portion of the resource is obtained from the content server of the content delivery network; and	No
1(f)	providing the obtained resource to the requesting device.	No

Determining to Not Cache ≠ Instruction to Not Cache

Examiner: determining whether to cache ≠ instruction to not cache.

The closest prior art is Raju et al., 7,552,233 B2, hereafter referred to as Raju. Raju teaches a policy for managing caching content at the edge of a network. Based on the configured rules of the policy whether to cache the content item is determined. Determining whether to cache a content item is different from “sending an instruction to not cache the content item” as recited in the claims.

EX-1007, 17 (emphasis added).

The other closest prior art of record is Swildens et al., US 2004/0221019 A1, hereafter referred to as Swildens. Swildens teaches a popularity service for determining the popularity of a content item. Whether the content item is cached is determined based on the popularity value. Accessing the popularity value is different from “sending an instruction to not cache the content item” as recited in the claims.

EX-1007, 17 (emphasis added).

Ground 1 Turns on One Issue

Does the priority chain disclose “an instruction to not cache”?

[0044] If the edge server **108** receives a redirect from the popularity service **102** without the “follow me” flag set (cases 1 and 2 above), it simply forwards the redirect to the client **104** (S4a, 522, 524). If the edge server **108** receives a “follow me” redirect, it obtains and caches the resource (at 526) and serves it to the client (at 528).

EX-1011, ¶44.

Google	Sandpiper
No: Determining whether to cache in the absence of a “follow me” tag is different from sending/receiving an affirmative instruction to not cache. Pet. 7-8, 10-11; Pet. Reply, 3-5.	Yes: A “redirect ... without the follow-me tag is an instruction to not cache and to forward the redirect” because “a redirect with the follow-me tag is an instruction to cache the resource and serve it to the client.” POR, 15.

Pet. 7-8, 10-11; POR, 15; Pet. Reply, 3-5; EX-1006, ¶¶54-65.

Without Retrieval, There Is No Resource to “Not Cache”

Two paths in ¶ 44:

- (1) “Follow me” flag → edge server obtains and caches resource
- (2) Redirect with no “follow me” flag → edge server forwards request, does not obtain resource – **there is no resource to cache/not cache**

[0044] If the edge server **108** receives a redirect from the popularity service **102** without the “follow me” flag set (cases 1 and 2 above), it simply forwards the redirect to the client **104** (**S4a, 522, 524**). If the edge server **108** receives a “follow me” redirect, it obtains and caches the resource (at **526**) and serves it to the client (at **528**).

EX-1011, ¶44.

Reasons to Not Cache ≠ Instruction to Not Cache

[0068] There are **several reasons** why a publisher /subscriber may not want the so-called “long-tail” content served from a caching CDN service, for example:

- If cache fills are done from the subscriber’s origin server and the subscriber pays for cache fill bandwidth, unnecessary fills for unpopular resources increase **bandwidth costs** without providing any value.
- Serving an unpopular resource through a CDN cache adds **latency** (due to the cache fill), and risks **forcing actually popular resources out of cache**, potentially causing thrashing. The result is lower efficiency and the risk of degraded service.

[0069] For **related or similar reasons**, a CDN provider generally also does not want to serve long-tail content from an edge cache:

- If cache fills are done from origin storage, unnecessary cache fills consume **bandwidth** both from the storage and cache systems, increasing (doubling) the internal cost to serve. This lowers **efficiency**, requiring relatively more bandwidth internally to serve the same content to the outside.
- The second argument above also applies from the CDN’s perspective: a CDN operator wants to minimize **latency** as well as the risk of thrashing in order to satisfy all of its subscribers.

“All the limitations must appear in the specification.” *Lockwood*

Patent Owner argues:

The fact that the words “instruction to not cache” are not literally present in the '573 patent and the '412 application specifications does not matter. *In re Johnson*, 558 F.2d 1008, 1019 (CCPA 1977); M.P.E.P. § 2173.05(i).

Patent Owner Response, 16.

But see:



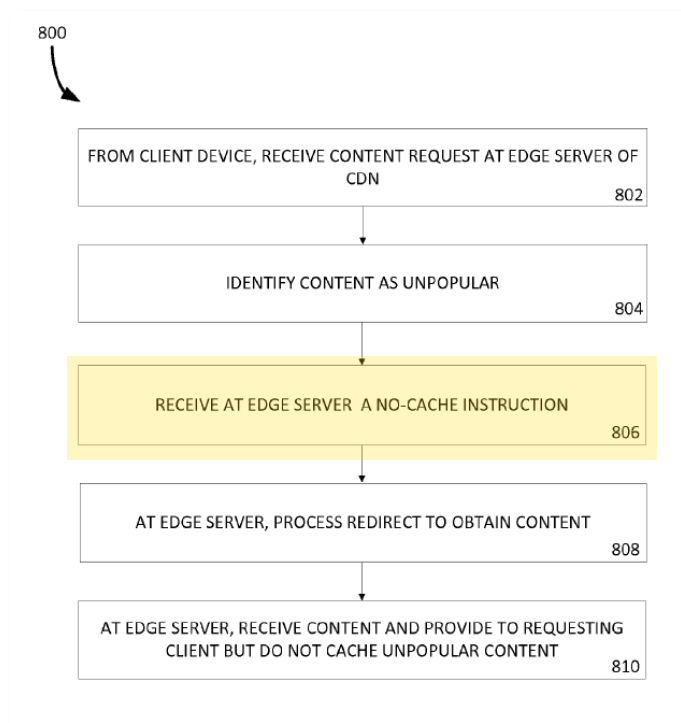
“It is the disclosures of the applications that count. Entitlement to a filing date does not extend to subject matter which is not disclosed, but would be obvious over what is expressly disclosed. It extends only to that which is disclosed....[A]ll the limitations must appear in the specification.”

Lockwood v. Am. Airlines, Inc., 107 F.3d 1565, 1571-1572 (Fed. Cir. 1997); *see also Rivera v. ITC*, 857 F.3d 1315 (Fed. Cir. 2017)

Pet. Reply, 2-5; Pet., 9-11; EX-1006, ¶¶60-65.

Newton-471 Teaches an Instruction to Not Cache

This shows the same lead inventor knew how to describe it.



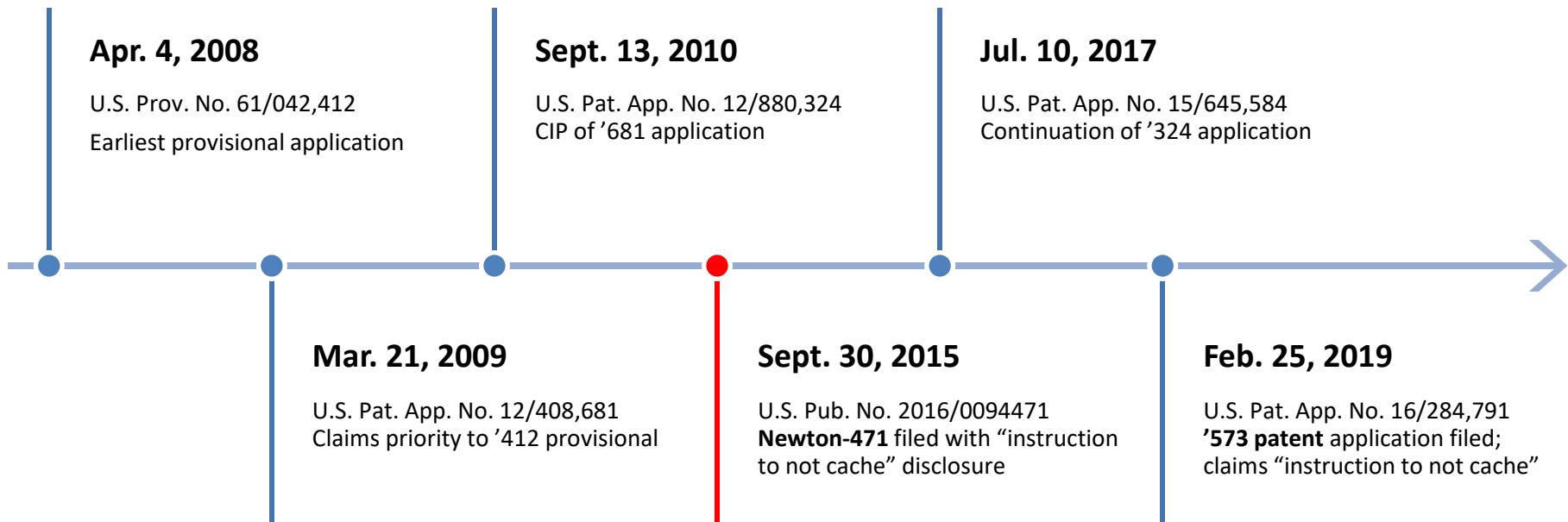
EX-1002, Fig. 8; Pet., 13-14, 21-22; Pet. Reply, 5-6.

[0098] In operation 806, the edge server 704 contacts the mid-tier 708 or other server hosting the content to receive the content in operation 806. In addition to returning the requested content, the mid-tier server 708 may also return a cache or no-cache instruction. In such an instance, the mid-tier 708 may have the content or may need to retrieve the content from the CDN origin 710, the customer origin 714, a different tier or otherwise. Regardless, the mid-tier 708, in conjunction with the popularity service 716, determines or otherwise identifies the popularity of the content. For unpopular content, the mid-tier 708 returns a no-cache instruction to the content server 706.

[0099] In the instance where the mid-tier 708 does not have the content, the mid-tier 708 may return a redirect to a component that hosts the requested content in addition to the no-cache instruction. The edge server 706 may process the redirect on behalf of the client 702. In particular, the edge server 706 requests the content from the host of the content (such as the CDN origin server 710 or the client origin 714) in operation 808. The content host then returns the content to the

EX-1002, ¶¶98-99; Pet., 21-22; Pet. Reply 5-6; EX-1006, ¶¶74-76, 96-99.

Timeline



Pet., 7-8, 10-11; EX-1006, ¶¶48, 56-65, 96-97.

Ground 2

**Dilley, Pai, and Wang
Renders Obvious Claims 1-20**

Ground 2 – Five Disputes

1. **motivation to combine (all claims)**
2. “popularity designation” (all claims)
3. “requesting the resource from a second server . . . based at least on the popularity designation” (claim **19**)
4. “redirect instruction” (claims **1, 11**)
5. “the second server is a peer server” (claims 5 and 15)

Representative Claims 1 and 5

Element	Claim 1	Disputed?
1(Pre)	A method of content delivery in a content delivery network comprising:	No
1(a)	receiving, at a first server of a first tier of servers of the content delivery network, a request from a requesting device for a resource available from the content delivery network;	No
1(b)	accessing a popularity service associated with the content delivery network to determine a popularity designation associated with the requested resource;	Yes
1(c)	requesting the resource from a second server of the content delivery network;	No
1(d)	processing, at the first server of the first tier of servers, a redirect instruction from the second server of the content delivery network to obtain the resource from a content server of the content delivery network;	Yes
1(e)	receiving an instruction to not cache the portion of the resource at the first server of the first tier of servers when the portion of the resource is obtained from the content server of the content delivery network; and	No
1(f)	providing the obtained resource to the requesting device.	No

Element	Claim 5	Disputed?
5	The method as recited in claim 1 wherein the second server is a peer server of the first tier of servers of the content delivery network.	Yes

Representative Claim 19

Element	Claim 19	Disputed?
19(Pre)	A method of content delivery in a content delivery network comprising:	No
19(a)	receiving, at a first server of a first tier of servers of the content delivery network, a request from a requesting device for a resource available from the content delivery network;	No
19(b)	accessing a popularity service associated with the first server of the first tier of the content delivery network to determine a popularity designation associated with the requested resource;	Yes*
19(c)	receiving an instruction to not cache the resource at the first server of the first tier of servers when the portion of the resource is obtained from the content server of the content delivery network;	No
19(d)	requesting the resource from a second server of the content delivery network based at least on the popularity designation associated with the requested resource; and;	Yes
19(e)	providing the obtained resource to the requesting device.	No

**same argument as set forth for limitation 1(b)*

Motivation to Combine Dilley, Pai & Wang

A POSITA would have used Pai's popularity-based distribution in Dilley's tiered architecture to provide an effective placement mechanism and would have incorporated Wang's redirect and instruction to not cache content.

Dilley:

Tiered cache hierarchy that distributes popular content to the edge to reduce origin-server load



Pai:

Popularity-based distribution that dynamically updates popularity values and distributes content across tiers based on those values



Wang:

Redirects and identifies resources to not cache using cache-control attributes to avoid unnecessary resource consumption

Dilley and Pai Are Compatible

Patent Owner wrongly argues that Dilley and Pai are incompatible as Dilley supports tiered *retrieval* and Pai supports popularity-based *placement*.

Pai:

- Uses popularity-based placement logic to determine where content resides within hierarchy
- Dynamically adjusts popularity values based on requests and redistributes content accordingly

Dilley:

- Uses retrieval logic to request content from the appropriate tier in the hierarchy when requested

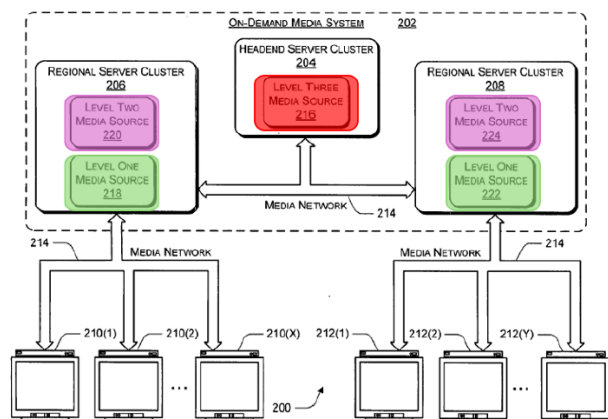


Figure 2

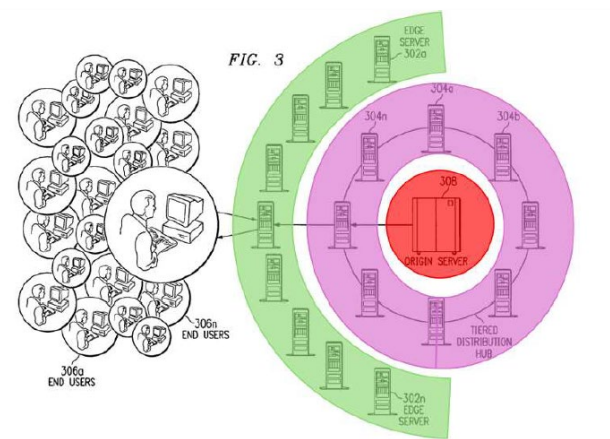


FIG. 3

Bodily Incorporation Is Not the Standard

Patent Owner wrongly argues that Dilley and Pai cannot be combined because Dilley uses parent hubs for popular content and Pai stores popular content at level one servers.

Moreover, Patent Owner's arguments improperly suggest the Petition is trying to bodily incorporate Pai into Dilley. Not so. The Board rejected these arguments at institution, noting Petitioner was not relying on bodily incorporation (it is not), and Pai's popularity service would supplement Dilley's existing tiered architecture in a predictable way (it would have). ID at 24. Nothing in the Response undermines that determination.

Nonetheless, obviousness turns on what the teachings would have suggested to a skilled artisan, not whether a reference can be inserted unchanged into another. *Axonics, Inc. v. Medtronic, Inc.*, 73 F.4th 950, 957 (Fed. Cir. 2023); *Qualcomm Inc. v. Intel Corp.*, No. 2022-1824, 2024 WL 4258364, at *4 (Fed. Cir. Sept. 23, 2024). Here, the combined teachings disclose a CDN hierarchy (Dilley), where popularity (Pai) determines if content should stay at the edge, migrate to a higher tier, or remain farther from the client. Pet. at 38, 44-45.

Pet., 38, 44-45; Pet. Reply, 9-10; EX-1006, ¶¶196, 201, 206-211.

Ground 2 – Five Disputes

1. motivation to combine (all claims)
2. **“popularity designation” (all claims)**
3. “requesting the resource from a second server ... based at least on the popularity designation” (claim 19)
4. “redirect instruction” (claims 1, 11)
5. “the second server is a peer server” (claims 5 and 15)

The Combination Discloses a Popularity Service to Determine a Popularity Designation

Independent Claim 1

1(b) accessing a popularity service associated with the content delivery network **to determine a popularity designation** associated with the requested resource;

Dilley discloses

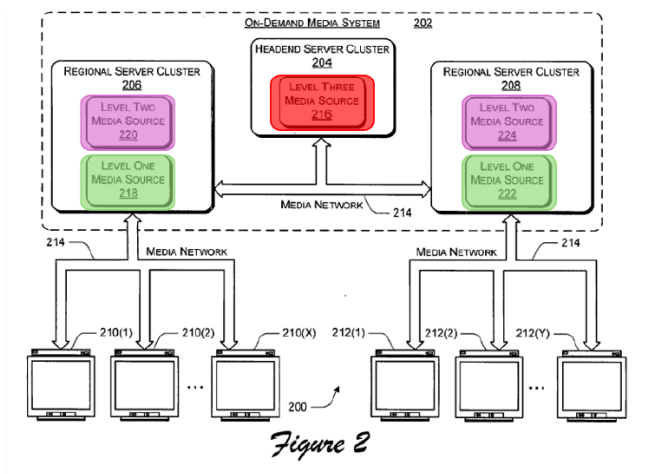
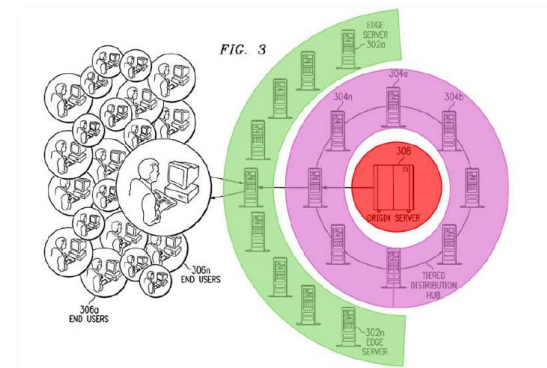
- Tiered CDN architecture with edge, parent, and origin servers, which is “particularly advantageous for flash-popular content”
- Metadata-based retrieval and hierarchy that controls whether and where the content is cached and from where the content is served.

EX-1003, Fig. 3, 7:31-35, 9:57-65; Pet. 34, 37, 43-44, 65; Pet. Reply, 10.

Pai discloses

- Popularity service to determine content popularity values
- Dynamically monitoring and adjusting popularity values
- Using popularity thresholds to determine content placement within a hierarchy

EX-1004, Fig. 2, ¶¶3, 13, 20, 47; Pet., 35, 37-39, 44-46, 60; Pet. Reply, 10.



Pai Determines a Popularity Designation Associated with Requested Content

Independent Claim 1

1(b) accessing a popularity service associated with the content delivery network **to determine a popularity designation** associated with the requested resource;

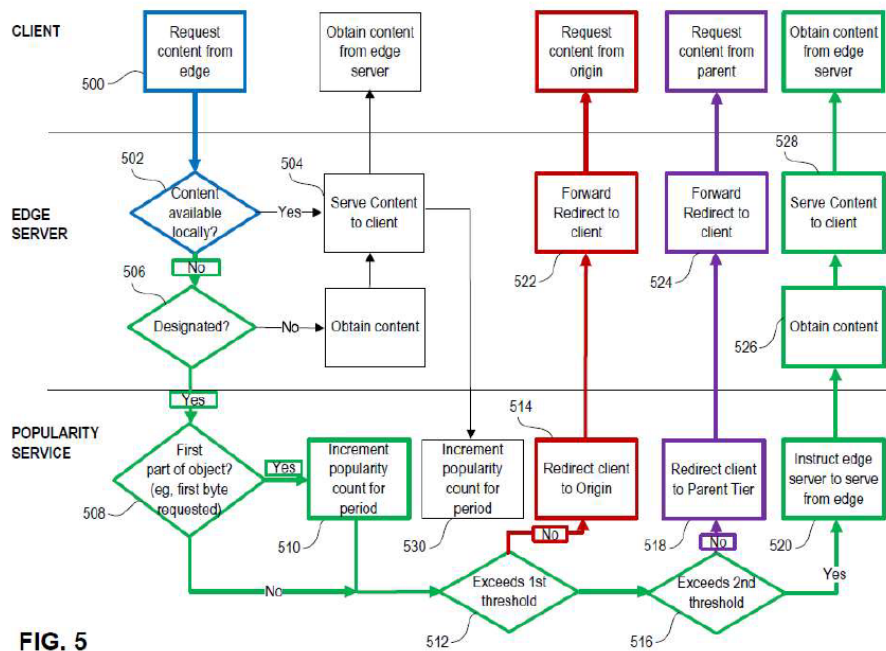


FIG. 5

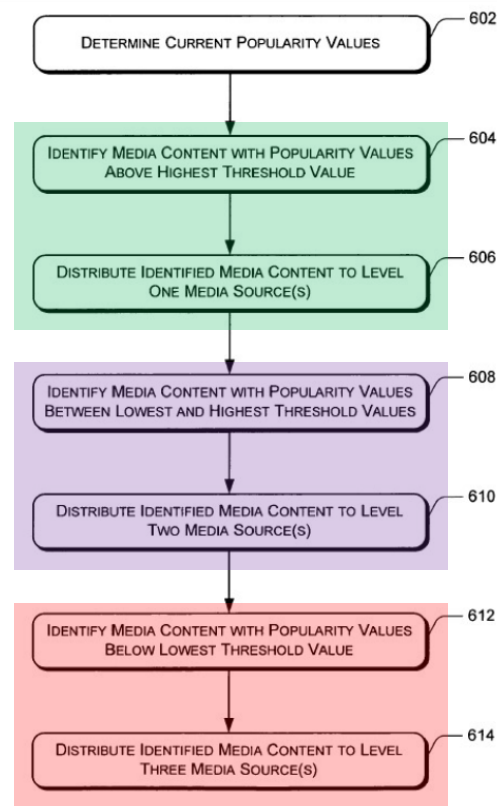


Figure 6

Pet. 5-6, 35, 44-46; EX-1001, Fig. 5, 6:53-7:30; EX-1004, Fig. 6, ¶¶38-45.

Popularity Designation Is a Popularity Threshold Determination

Independent Claim 1

1(b) accessing a popularity service associated with the content delivery network **to determine a popularity designation** associated with the requested resource;

Google's Position	Sandpiper's Position
Popularity designation = a popularity threshold determination, e.g., whether the resource is popular or exceeds a threshold. Pet., 44-46; Pet. Reply, 11-12.	Popularity designation = a prerequisite eligibility check to determine if the resource is even eligible for popularity-based routing. POR, 26-27; Sur-reply, 5-6, 8.

Step 506 Does Not Access the Popularity Service

Independent Claim 1

1(b) accessing a popularity service associated with the content delivery network to determine a popularity designation associated with the requested resource;

If the object is not available on the edge server 108 or on a peer, the edge server 108 ascertains whether this object is served based on its popularity (i.e., whether this object has been designated so that the object's popularity will be used to determine where it will be served from) (at 506). If so, then the request is sent to the popularity service 102 associated with the edge server 108, in this case, to the popularity server for the this group (S3a).

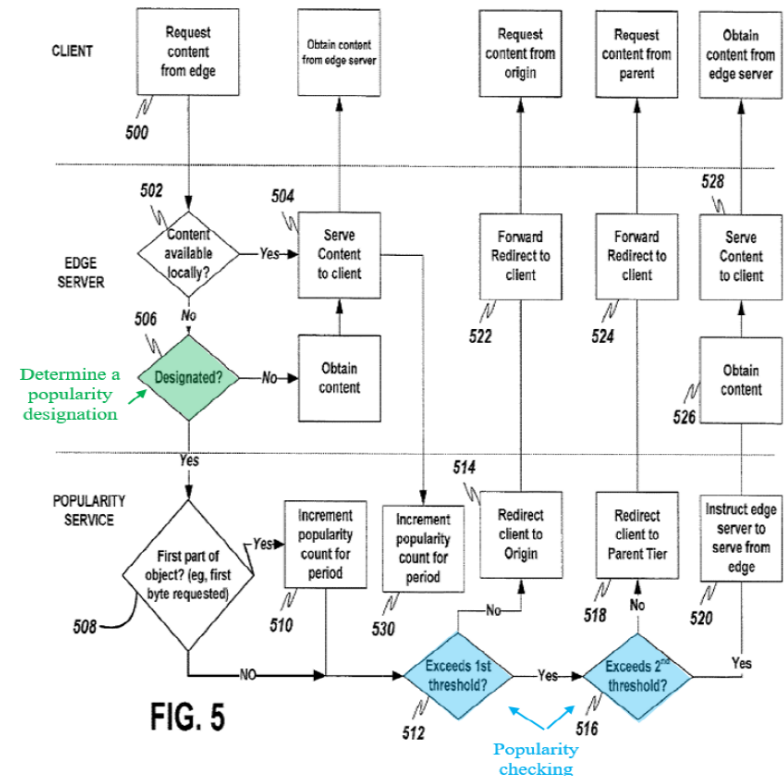


FIG. 5

EX1001, FIG. 5 (annotated).

EX-1001, 6:27-34; Pet., 5-6; POR, 26-27; Pet. Reply, 12-13.

Dependent Claims Confirm Petitioner's Interpretation

Independent Claim 1

1(b) accessing a popularity service associated with the content delivery network **to determine a popularity designation** associated with the requested resource;

2. The method as recited in claim 1 wherein the popularity designation associated with the requested resource indicates that the requested resource is not popular.

3. The method as recited in claim 1 wherein the popularity designation associated with the requested resource indicates that the requested resource is popular, the method further comprising: receiving an instruction to cache the resource at the first server of the first tier of servers when the resource is obtained from the content server of the content delivery network.

6. The method as recited in claim 1 wherein the popularity designation associated with the requested resource comprises an indication whether a current popularity value for the resource exceeds a first predetermined popularity threshold.

Patent Owner Imports a Prerequisite Eligibility Requirement

POR contends that the “designation” occurs *before* popularity checking:

The '573 patent describes a specific operation. The system “determine[s] a popularity designation” for a requested resource—i.e., whether the specific resource is even eligible for popularity-based routing. EX1001, 6:27-31 (“whether this object has been designated so that the object’s popularity will be used to determine where it will be served from”). After determining a resource is so “designated” the system performs popularity-based routing: comparing the resource’s popularity (e.g., a “popularity count”) against specific thresholds to determine which server to fulfill the request. EX1001, 6:53-7:24, FIG. 5; EX2016, ¶64.

As the '573 patent explains, “not necessarily all” content uses the popularity service; rather, a “popularity check” is performed only for “certain designated content.” EX1001, 5:47-50, 5:62-64. Thus, this claimed “*determining a popularity designation*” (Step 506 of Figure 5) occurs *before* the “popularity checking” (Steps 512 and 516 of Figure 5) is ever initiated. EX1001, 6:27-41, FIG. 5; EX2016, ¶65.

Patent Owner's Sur-Reply Changes Its Theory

Sur-Reply argues that popularity checking can be part of the “designation”:

In fact, Petitioner improperly narrows the meaning of “popularity designation” as merely a restrictive “binary” eligibility determination. Reply, 12.

On the contrary, the “popularity designation” can have multiple functions besides indicating baseline eligibility. As Petitioner itself pointed out, the popularity designation indicates whether the requested resource is popular or not, and it “comprises” an indication of “whether a current popularity value ... exceeds popularity threshold.” Reply, 12-13; EX1001, Claims 2, 3, 12, 13, 16. Importantly, the use of the open-ended term “comprises” in dependent claim 16 confirms that it does not restrict or exhaustively define the scope of the parent claim’s term.

Ground 2 – Five Disputes

1. motivation to combine (all claims)
2. “popularity designation” (all claims)
3. **“requesting the resource from a second server . . . based at least on the popularity designation” (claim 19)**
4. “redirect instruction” (claims 1, 11)
5. “the second server is a peer server” (claims 5 and 15)

The Combination Discloses Retrieval Based at Least on the Popularity Designation

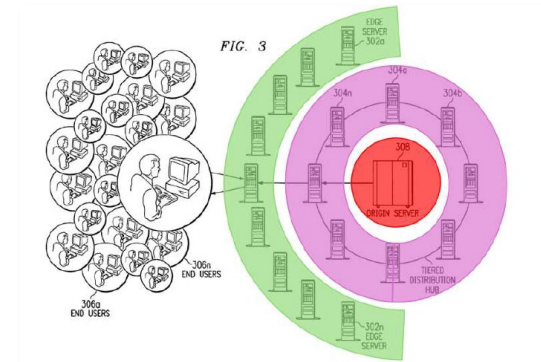
Independent Claim 19

19(d) requesting the resource from a second server of the content delivery network **based at least on the popularity designation** associated with the requested resource; and

Dilley discloses

- Tiered CDN architecture with edge, parent, and origin servers, which is “particularly advantageous for flash-popular content”
- Metadata-based retrieval and hierarchy that controls whether and where the content is cached and from where the content is served.

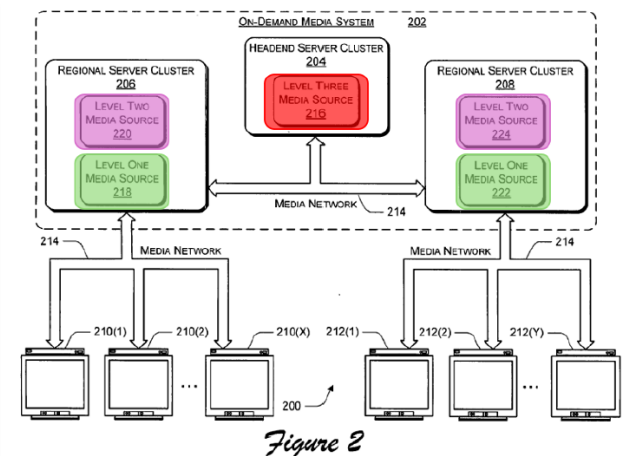
EX-1003, Fig. 3, 7:31-35, 9:57-65; Pet. 34, 37, 43-44, 65, 83-85; Pet. Reply, 10.



Pai discloses

- Popularity service to determine content popularity values
- Dynamically monitoring and adjusting popularity values
- Using popularity thresholds to determine content placement within a hierarchy
- Retrieving and serving content based on its popularity

EX-1004, Fig. 2, ¶¶3, 13, 17, 20, 47; Pet., 35, 37-39, 44-46, 60, 83-85; Pet. Reply, 10;
EX-1006, ¶¶184, 188.



Patent Owner's "Placement vs. Delivery" Distinction Is Artificial

Independent Claim 19

19(d) requesting the resource from a second server of the content delivery network **based at least on the popularity designation** associated with the requested resource; and

- A CDN stores content so that it can later be retrieved. Retrieval follows placement. Placing popular content closer to users affects future delivery and retrieval.

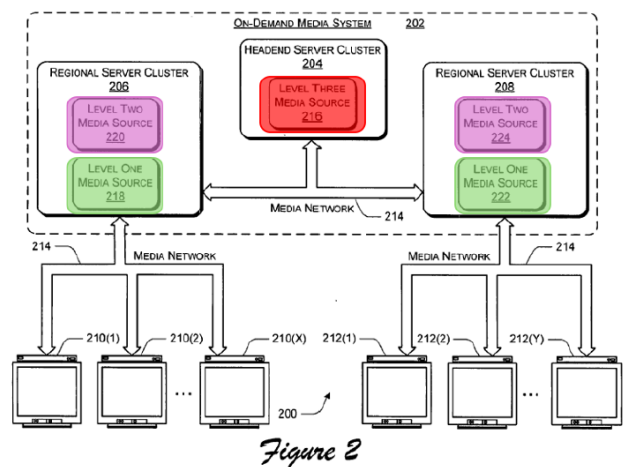


Figure 2

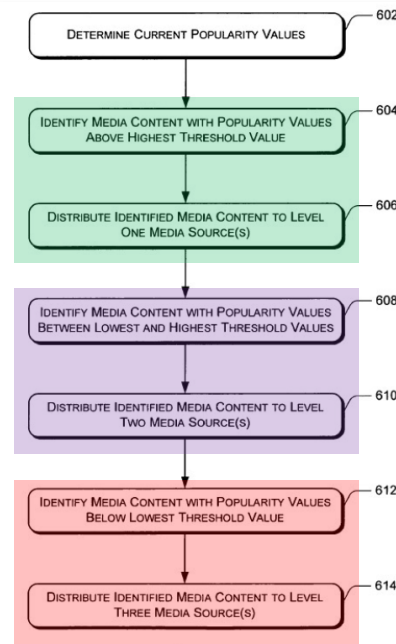


Figure 6

[0003] Popularity-based on-demand media distribution is described. Media content is distributed among multiple on-demand media sources, each source being configured to serve the media content to one or more clients. The media content is distributed based on popularity such that more popular media content is served from one on-demand media source and less popular media content is served from another on-demand media source. Over time, the popularity

[0020] In the illustrated example, level three media source 216 is configured to maintain and serve the least popular media content; level two media sources 220 and 224 are configured to maintain and serve more popular media content; and level one media sources 218 and 222 are configured to maintain and serve the most popular media content. As an

Patent Owner Attacks References Individually

Patent Owner attacks references individually, arguing that Dilley or Pai alone must independently disclose the full claim 19 retrieval step.

First, Dilley determines *a second server* (a parent server) based on object-specific metadata, not based on a “*popularity designation*.” Section VIII.A.1.

POR, 31.

Pai is equally deficient. Pai discloses a mechanism for distributing or storing media content within a CDN based on popularity values. EX1004, ¶¶13, 20, 39,

POR, 32.

Therefore, neither Dilley nor Pai teaches “*requesting the resource from a second server ... based at least on the popularity designation associated with the requested resource*,” as recited in claim 19, *alone or in combination*. Wang is not

POR, 33.

Pet. Reply, 13-16.

Dr. Mowry Does Not Fill Gaps; He Explains the Combination

Dr. Mowry explained how a skilled artisan would have understood and combined Dilley and Pai:

- Dilley ties server selection to popularity to prevent the origin from becoming “overwhelmed with requests for popular content.” EX-1006, ¶353
- Pai dynamically distributes and manages content based on popularity such that requests can be served from given tiers based on popularity thresholds (most popular at level 1, less popular at level 2, least popular at level 3). EX-1006, ¶354
- A POSITA would have understood the combination performs the claimed routing given the distribution of popular content at the edge, mid-level popular content at mid-tier servers, and the least popular content at the origin farthest from requesting end-users. EX-1006, ¶355

Patent Owner's Argument Misses the Point

Patent Owner argues “closely related” does not satisfy the particularity requirement of obviousness

- But, in the combined system, retrieval follows popularity-based placement
- Indeed, the combination incorporates Dilley's retrieval hierarchy with Pai's popularity-based placement
- Relatedness provides yet another reason why a skilled artisan would have combined these systems

Pet., 83-84; EX-1006, ¶¶352-353; Pet. Reply, 7-8, 14-16.

- *Unified* is distinguishable:

We agree with Patent Owner that Petitioner cannot show a motivation to combine Kim and Huang based solely on the fact that those references relate to deblocking reconstructed video images. Although the extent to which references belong to the same or related fields of technology is one of the factors that precedent requires us to consider, that alone is not enough for Petitioner to meet its burden of presenting a sufficient rationale to support an obviousness conclusion. *See Securus Techs., Inc. v. Glob. Tel*Link Corp.*,

Unified Patents Inc. v. DivX, LLC, IPR2019-01379, Paper 52 (P.T.A.B. Feb. 8, 2021) (emphasis added)

Ground 2 – Five Disputes

1. motivation to combine (all claims)
2. “popularity designation” (all claims)
3. “requesting the resource from a second server ... based at least on the popularity designation” (claim 19)
4. **“redirect instruction” (claims 1, 11)**
5. “the second server is a peer server” (claims 5 and 15)

The Petition Incorporates Wang's Redirect into Dilley

Independent Claim 1

1(d) processing, at the first server of the first tier of servers, a **redirect instruction** from the second server of the content delivery network to obtain the resource from a content server of the content delivery network;

Dilley discloses

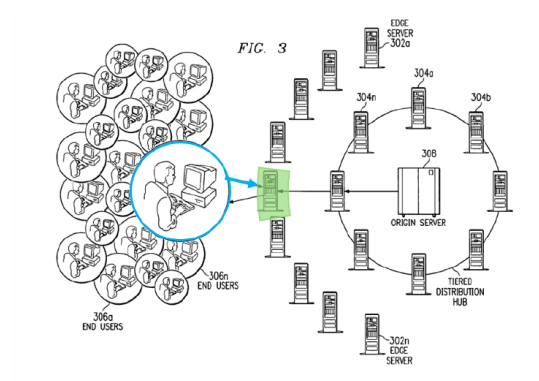
- Tiered CDN architecture with edge, parent, and origin servers
- Cache-miss escalation through the CDN hierarchy

EX-1003, Fig. 3, 5:4-21, 7:31-40; Pet. 34-35, 42-43, 50.

Wang discloses

- Hierarchical content lookup in a three-tiered CDN
- Redirect instructions and HTTP redirect mechanisms
- Cache/no-cache attributes

EX-1005, ¶¶47-49, 123-124; Pet., 35-36, 39-40, 50-53.



In case of a not cacheable attribute, the caching engine one will http-redirect the subscriber to the origin server.

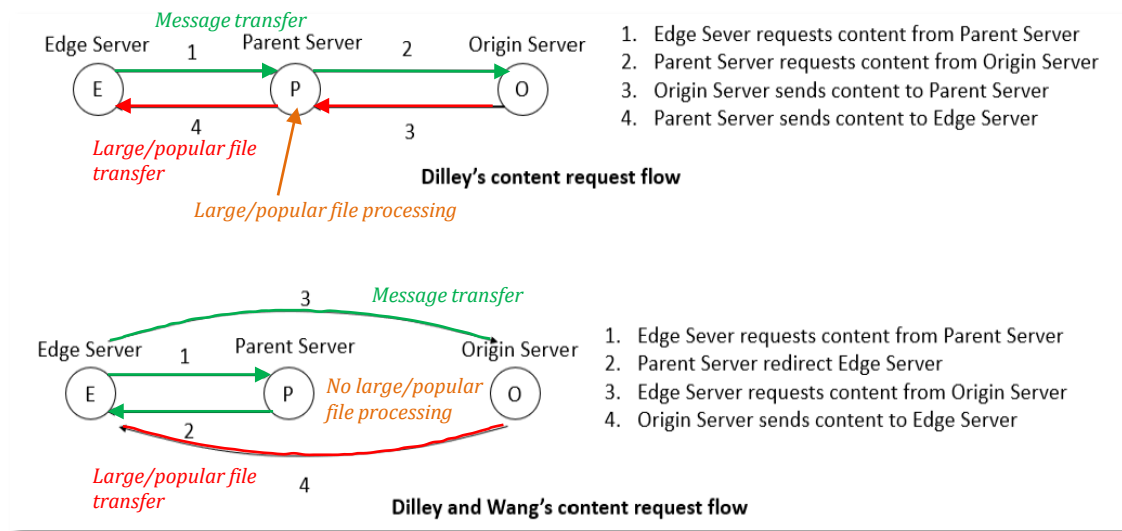
[0124] If a cacheable content is indicated, the caching engine one will then initiate a new http session on behalf of the subscriber to the origin server and cache the content if “cacheable” is also specified in the http response from the origin server. The redirect message is also supported by RTSP, but may not always be supported by other existing

Wang's Redirect Provides Predictable Benefits

Wang's redirect:

- Allows the edge server obtain content directly from the origin server. EX-1006, ¶255
- Prevents parent server from fetching and relaying large content, thereby reducing latency and processing burdens. EX-1006, ¶256
- Permits parent servers to remain available for other requests. EX-1006, ¶257

EX-1006, ¶¶255-257; Pet., 51-52; POR 48 (annotated).



Annotations added to identify **message transfers**, **large/popular file transfers**, and **file processing**.

Wang Was Part of the Petition's Combination

server instead. EX-1006, ¶¶250-253; EX-1003, 7:15-30. Indeed, a POSITA would have been motivated to implement this redirect instruction from Wang. *See e.g.*, EX-1005, ¶¶[0123]-[0130]; EX-1006, ¶¶253-254.

Id. In this instance, Wang explains that the level 1 server will process a redirect instruction to obtain the content from the origin server. EX-1005, ¶[0123] (explaining, for not-cacheable content, the level 1 server will redirect the request to the origin server), ¶[0124] (explaining the same redirect process to the origin server is supported for cacheable content).

obtain the content from the origin content server. EX-1006, ¶255. And a POSITA would have understood that Wang's redirect procedure would have added benefits because the edge server would then process the redirect instruction. It would thus be expecting the response directly from the origin server. Such a design would decrease latency as it would remove the parent server as a "middleman," and would allow the edge server to communicate with and receive content directly from the origin server. EX-1006, ¶¶255-256. In addition, this process would allow the parent server to remain available to service other requests received by the CDN. EX-1006, ¶257.

Incorporation of Wang's known redirect

Benefits to incorporating Wang

Pet., 50-51; *see also* *Pet.*, 39-40.

Dilley's Disclosures Does Not Preclude Obviousness

Patent Owner argues that Dilley's origin retrieval thwarts the combination with Wang. POR, 41-45; Sur-reply, 15-18.

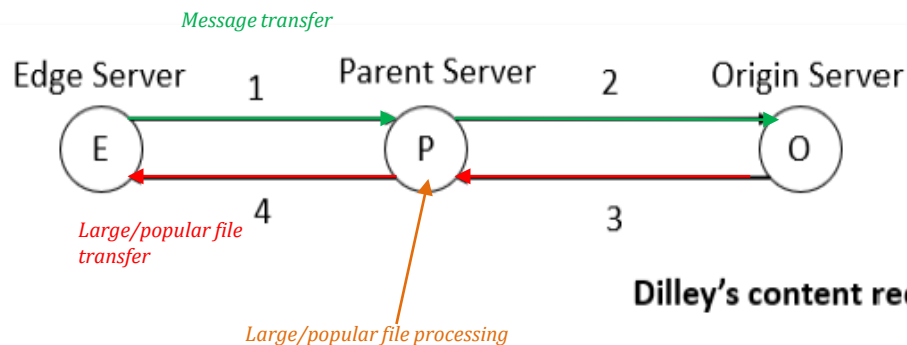
Not so. Dilley's retrieval path does not preclude a POSITA from modifying it with Wang given the benefits and reasons for doing so. Pet. 36-40, Pet. Reply, 6-7, 18-19.



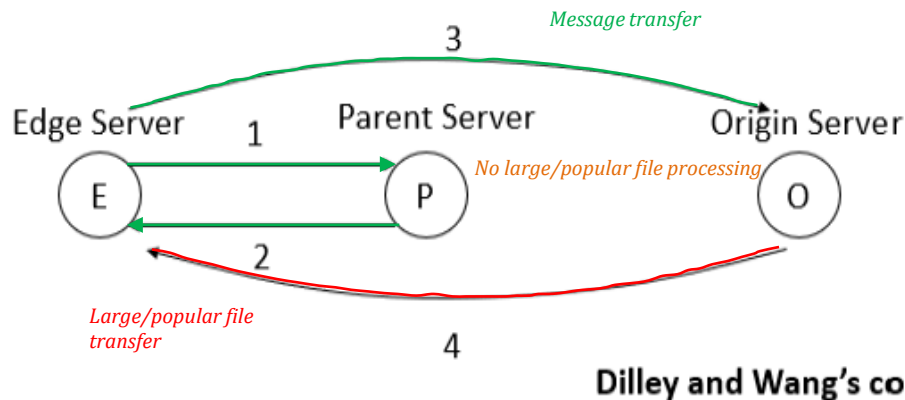
USAA argues that a skilled artisan would not be motivated to implement Garcia using a thin-client application because Garcia's remote check deposit **would already be achieved** by using the mobile phone's camera We disagree. PNC's petition asserted, and the Board found, that Byrne's thin-client application provided benefits that would render it obvious.

See United Servs. Auto. Ass'n v. PNC Bank N.A., No. 2023-2171, 2025 WL 339662, at *3 (Fed. Cir. Jan. 30, 2025) (emphasis added)

Wang Does Not Inject Redundant Architecture



1. Edge Server requests content from Parent Server
2. Parent Server requests content from Origin Server
3. Origin Server sends content to Parent Server
4. Parent Server sends content to Edge Server



1. Edge Server requests content from Parent Server
2. Parent Server redirect Edge Server
3. Edge Server requests content from Origin Server
4. Origin Server sends content to Edge Server

Annotations added to identify message transfers, large/popular file transfers, and file processing.

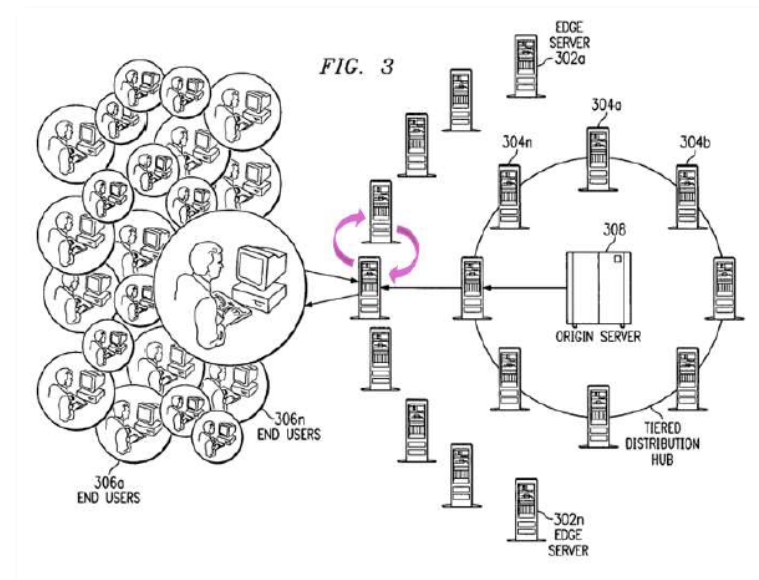
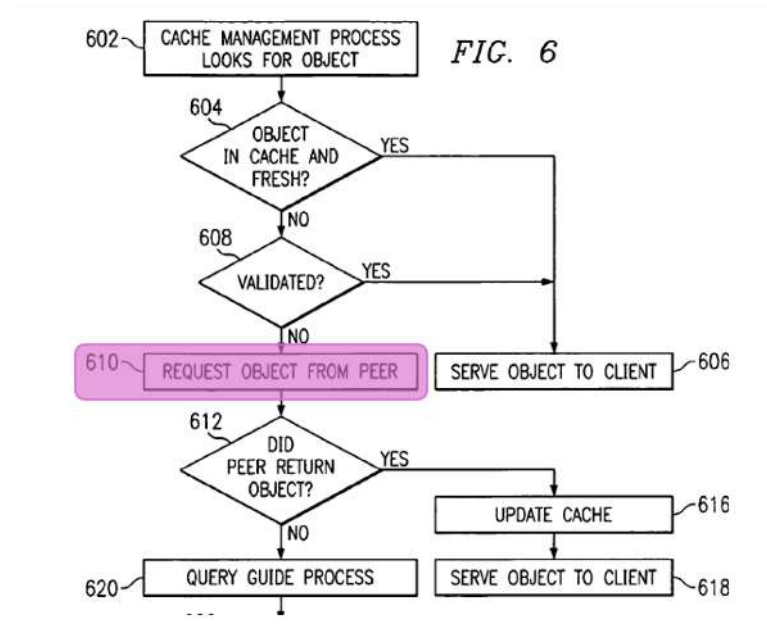
Ground 2 – Five Disputes

1. motivation to combine (all claims)
2. “popularity designation” (all claims)
3. “requesting the resource from a second server ... based at least on the popularity designation” (claim 19)
4. “redirect instruction” (claims 1, 11)
5. **“the second server is a peer server” (claims 5 and 15)**

The Combination Teaches That the Second Server Is a Peer

Dependent Claim 5

5. The method as recited in claim 1 wherein **the second server is a peer server** of the first tier of servers of the content delivery network.



Pet. 67-69; EX-1003, Fig. 3, Fig. 6, 8:11-18; EX-1006, ¶¶258-260, 305-307; Pet. Reply, 22-23.

The Combination Teaches That the Second Server Is a Peer

A POSITA would have understood that these redirects were instructions to obtain the content from the origin content server. EX-1006, ¶255. And a POSITA would have understood that Wang's redirect procedure would have added benefits because the edge server would then process the redirect instruction. It would thus be expecting the response directly from the origin server. Such a design would decrease latency as it would remove the parent server as a "middleman," and would allow the edge server to communicate with and receive content directly from the origin server. EX-1006, ¶¶255-256. In addition, this process would allow the parent server to remain available to service other requests received by the CDN. EX-1006, ¶257.

Each of the above benefits would apply equally to the example where the second server is a peer server such that a POSITA would recognize the benefits of having the peer server sending a redirect instruction to the edge server, directing the edge server to obtain the resource from a parent or origin content server instead. EX-1006, ¶258.

Patent Owner Ignores Dilley and the Combination

Patent Owner argues that a peer driven redirect to the origin would bypass the parent server and eliminate the parent server's buffering role to insulate the origin server.

While Petitioner correctly notes that Dilley teaches peer-server queries when an edge server lacks an object, Petitioner intentionally omits Dilley's mandatory next step when a peer query fails to return a valid object. Reply, 23. In Dilley's actual architecture, if "no peer copy of the object is available," the edge server continues to query its own localized "guide process" to establish a connection upward to a parent server. EX1003, 8:18-25, FIG. 6.

* * *

defeats Dilley's core principle of operation. Dilley explicitly mandates that parent servers must be utilized within the cache hierarchy to protect the origin server from unnecessary requests and avoid overwhelming its internal infrastructure. EX1003, 5:16-21. Imputing a peer-to-peer redirect that bypasses the parent tier entirely strips away the network's buffering capabilities, forcing low-level edge nodes to hit the origin server prematurely. Because doing so would actively destroy the

Patent Owner Ignores Dilley and the Combination

Patent Owner ignores Dilley's disclosure that allows peer servers to function as another node in the request path before redirecting to an origin server.

As illustrated in FIG. 4, a given content delivery network 400 may comprise a set of regions 402a-n, each of which comprises a set of content (or "edge") servers 404a-n. CDN customers include a set of content providers (CPs), each of which typically operates a set of one or more origin servers 406. Typically, when a given requested object is not available from a given content server 404, the object may be retrieved from another edge server in the region or, failing that, from the origin server. This known operation is modi-