US 20140108674A1

(19) **United States**

(12) **Patent Application Publication**

Eggleston et al.

(10) **Pub. No.: US 2014/0108674 A1**

(43) **Pub. Date: Apr. 17, 2014**

(54) **ANYCAST REDIRECT TO UNICAST CONTENT DOWNLOAD**

(71) Applicant: **LIMELIGHT NETWORKS, INC.**, TEMPE, AZ (US)

(72) Inventors: **Jason Eggleston**, Newport Beach, CA (US); **Bradley B. Harvell**, Chandler, AZ (US); **Mohan I. Kokal**, Peoria, AZ (US)

(73) Assignee: **LIMELIGHT NETWORKS, INC.**, TEMPE, AZ (US)

(21) Appl. No.: **14/047,824**

(22) Filed: **Oct. 7, 2013**

Related U.S. Application Data

(63) Continuation of application No. 13/344,497, filed on Jan. 5, 2012, now Pat. No. 8,621,042, which is a continuation of application No. PCT/US2010/062156, filed on Dec. 27, 2010.

(30) **Foreign Application Priority Data**

Nov. 17, 2011 (AU) 2011200629

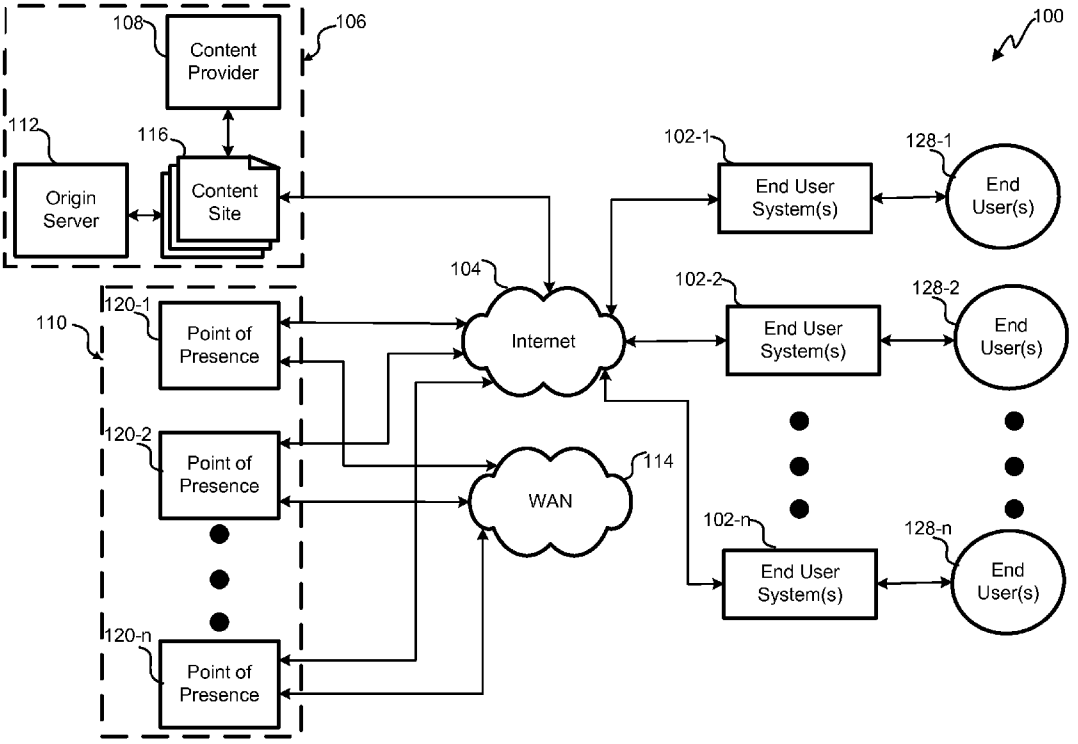
Publication Classification

(51) **Int. Cl.**
H04L 29/12 (2006.01)

(52) **U.S. Cl.**
CPC ***H04L 61/1511*** (2013.01)
USPC **709/245**

(57) **ABSTRACT**

A method and system for delivering content with a content delivery network (CDN) to a user computer is disclosed. After an initial request to a point of presence (POP) of the CDN for the location of the content object, an Anycast Internet protocol (IP) address is assigned. The user computer requests the content object which may find its way to the same or a different POP depending on what is closer in Internet terms. The request returns a POP-specific IP in a redirect such that the request resolves to the POP referenced in the immediately preceding sentence. One or more edge servers deliver the content object to the user computer.



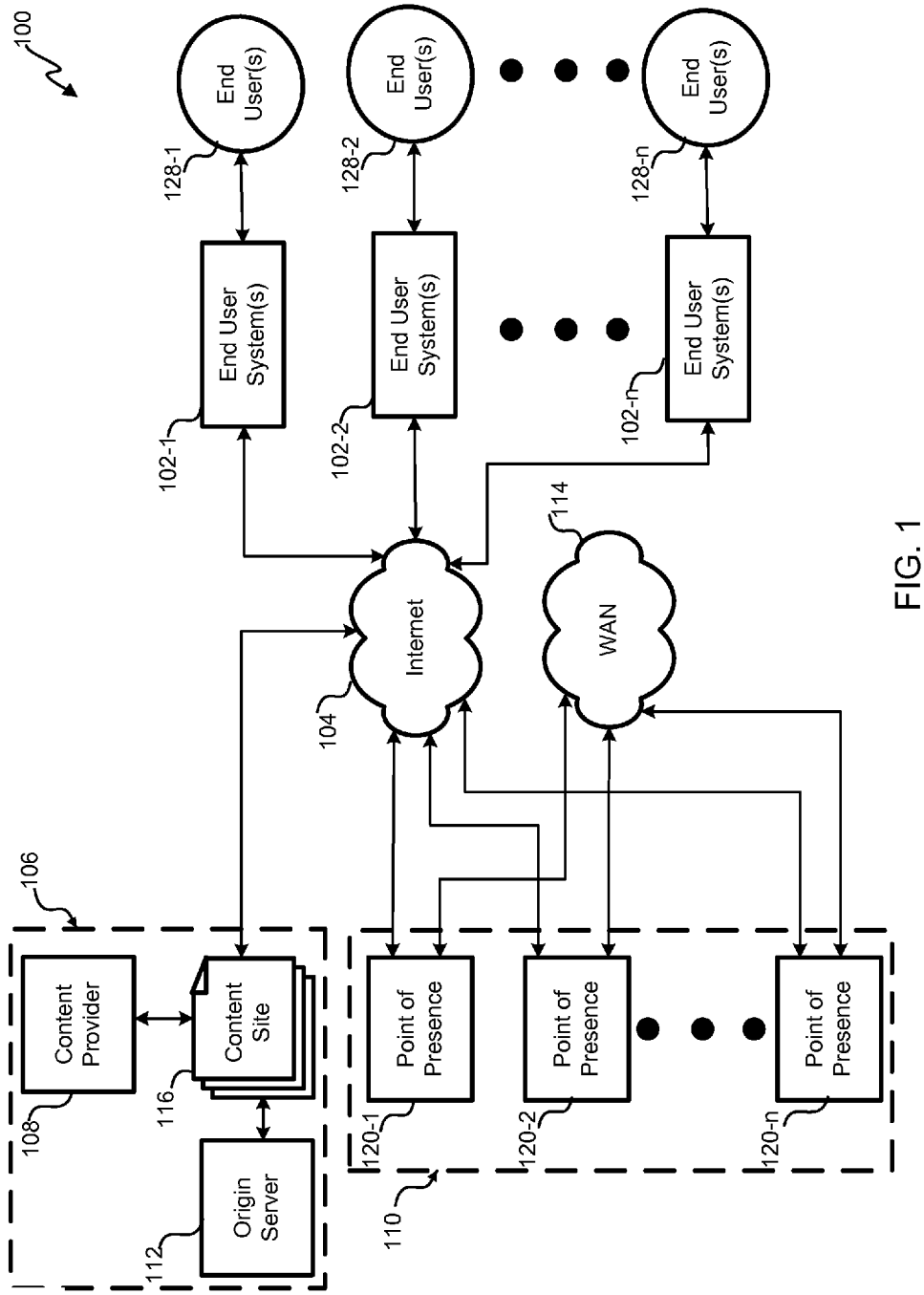

100
End User(s)
128-1
End User(s)
128-2
End User(s)
128-n
End User System(s)
102-1
End User System(s)
102-2
End User System(s)
102-n
114
Internet
104
WAN
106
Content Provider
108
Content Site
116
Origin Server
112
Point of Presence
120-1
Point of Presence
120-2
Point of Presence
120-n
110

FIG. 1

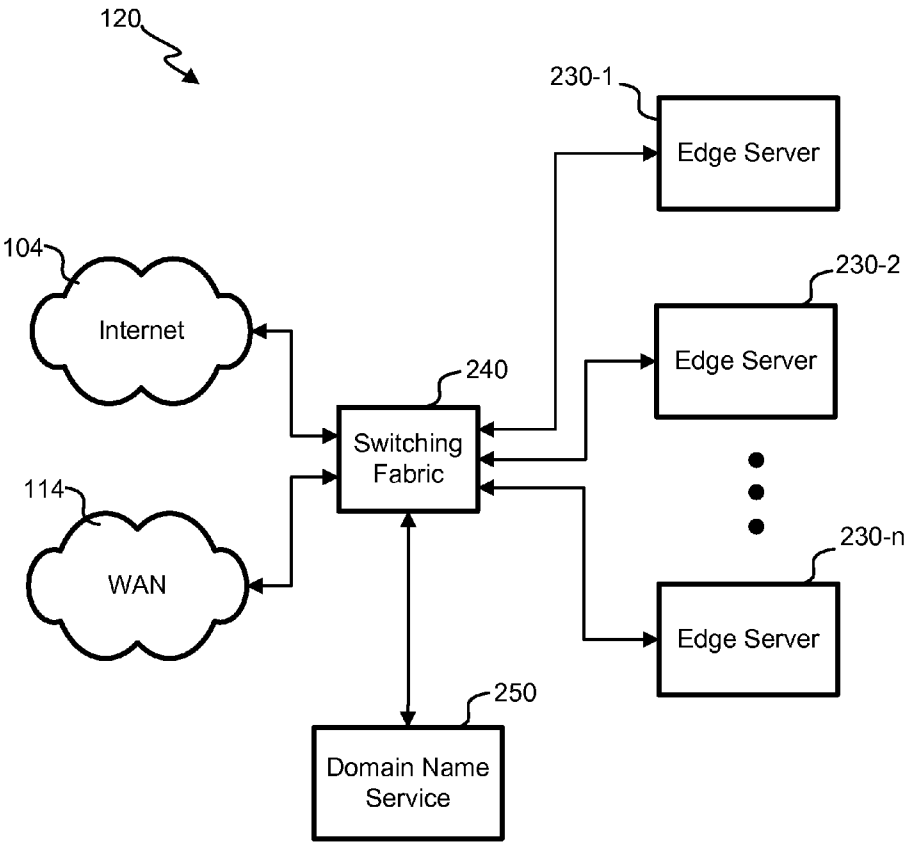

120
230-1
Edge Server
104
Internet
230-2
Edge Server
240
Switching Fabric
114
WAN
230-n
Edge Server
250
Domain Name Service

FIG. 2

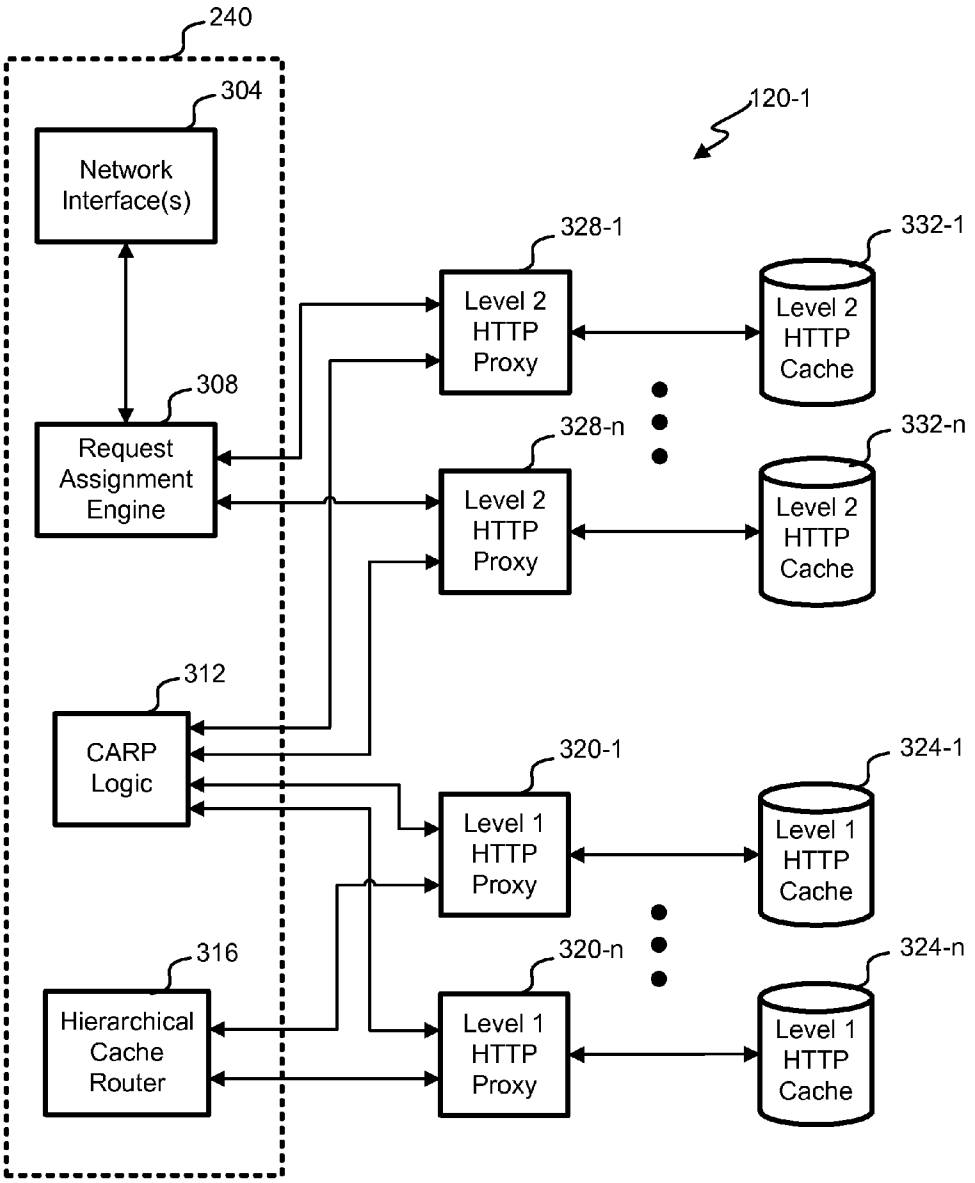

240
304
Network Interface(s)
120-1
328-1
332-1
Level 2 HTTP Proxy
Level 2 HTTP Cache
308
Request Assignment Engine
328-n
332-n
Level 2 HTTP Proxy
Level 2 HTTP Cache
312
CARP Logic
320-1
324-1
Level 1 HTTP Proxy
Level 1 HTTP Cache
316
320-n
324-n
Hierarchical Cache Router
Level 1 HTTP Proxy
Level 1 HTTP Cache

FIG. 3A

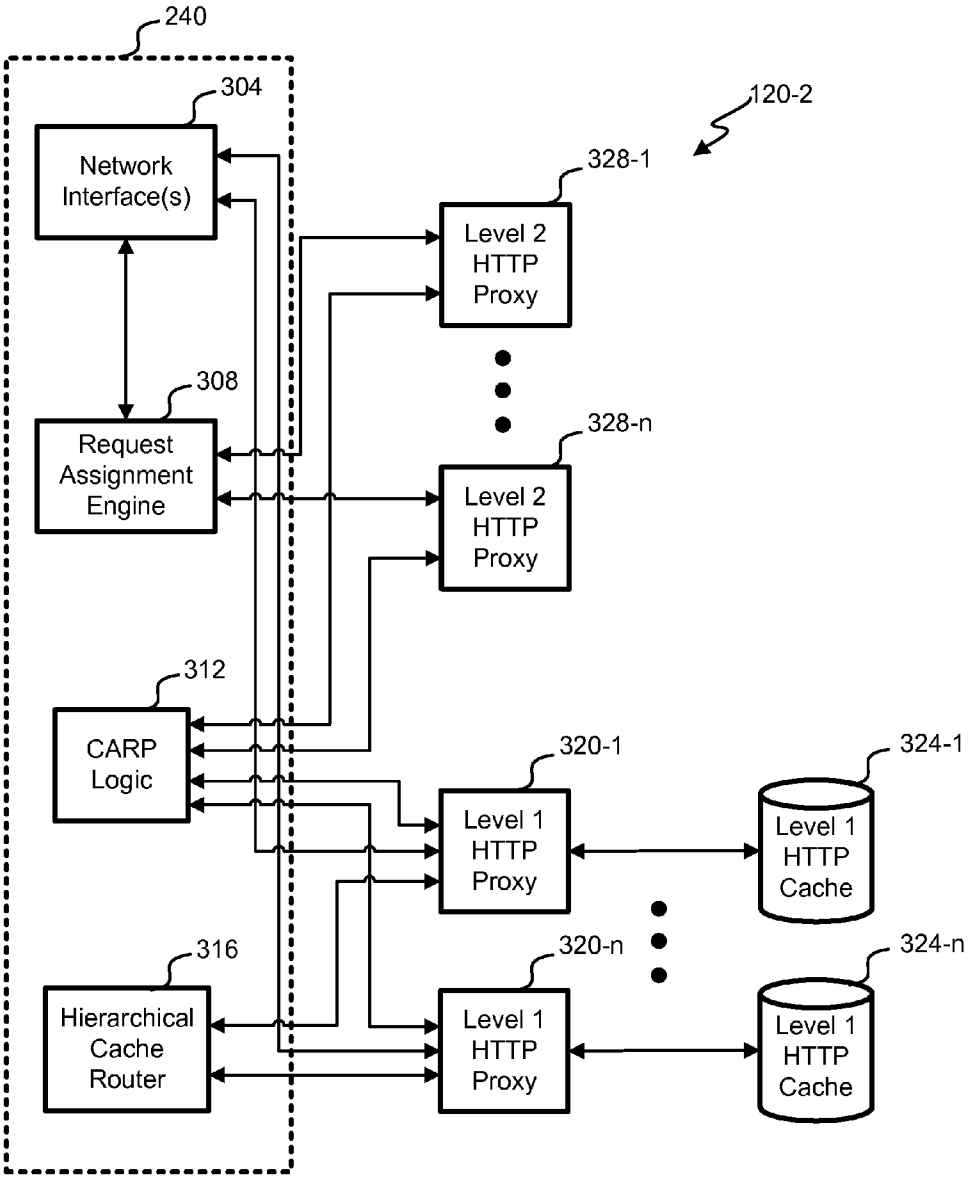

240
304
Network Interface(s)
120-2
328-1
Level 2 HTTP Proxy
308
Request Assignment Engine
328-n
Level 2 HTTP Proxy
312
CARP Logic
320-1
324-1
Level 1 HTTP Proxy
Level 1 HTTP Cache
316
320-n
324-n
Hierarchical Cache Router
Level 1 HTTP Proxy
Level 1 HTTP Cache

FIG. 3B

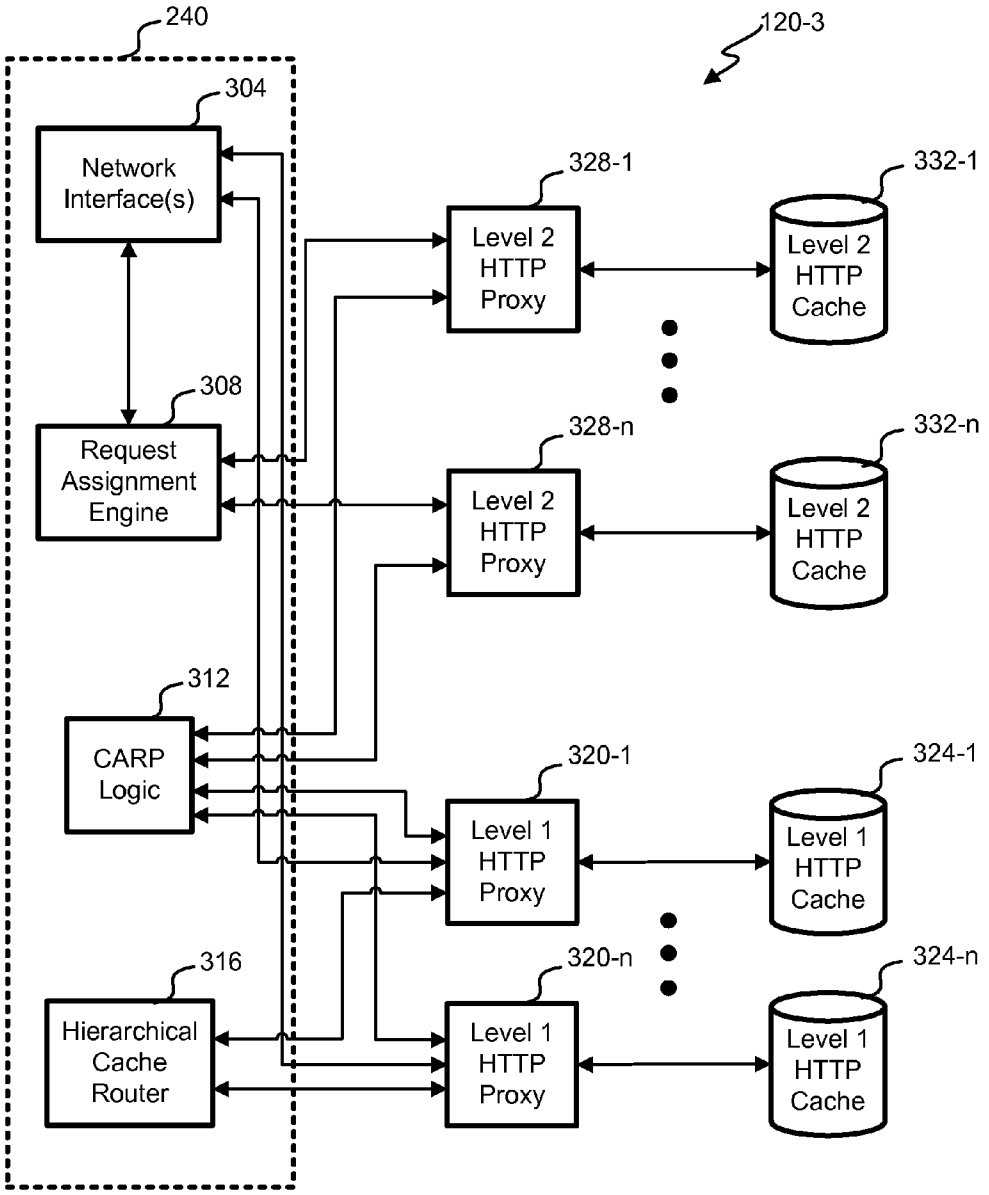

240
120-3
304
Network Interface(s)
328-1
332-1
Level 2 HTTP Proxy
Level 2 HTTP Cache
308
Request Assignment Engine
328-n
332-n
Level 2 HTTP Proxy
Level 2 HTTP Cache
312
CARP Logic
320-1
324-1
Level 1 HTTP Proxy
Level 1 HTTP Cache
316
320-n
324-n
Hierarchical Cache Router
Level 1 HTTP Proxy
Level 1 HTTP Cache

FIG. 3C

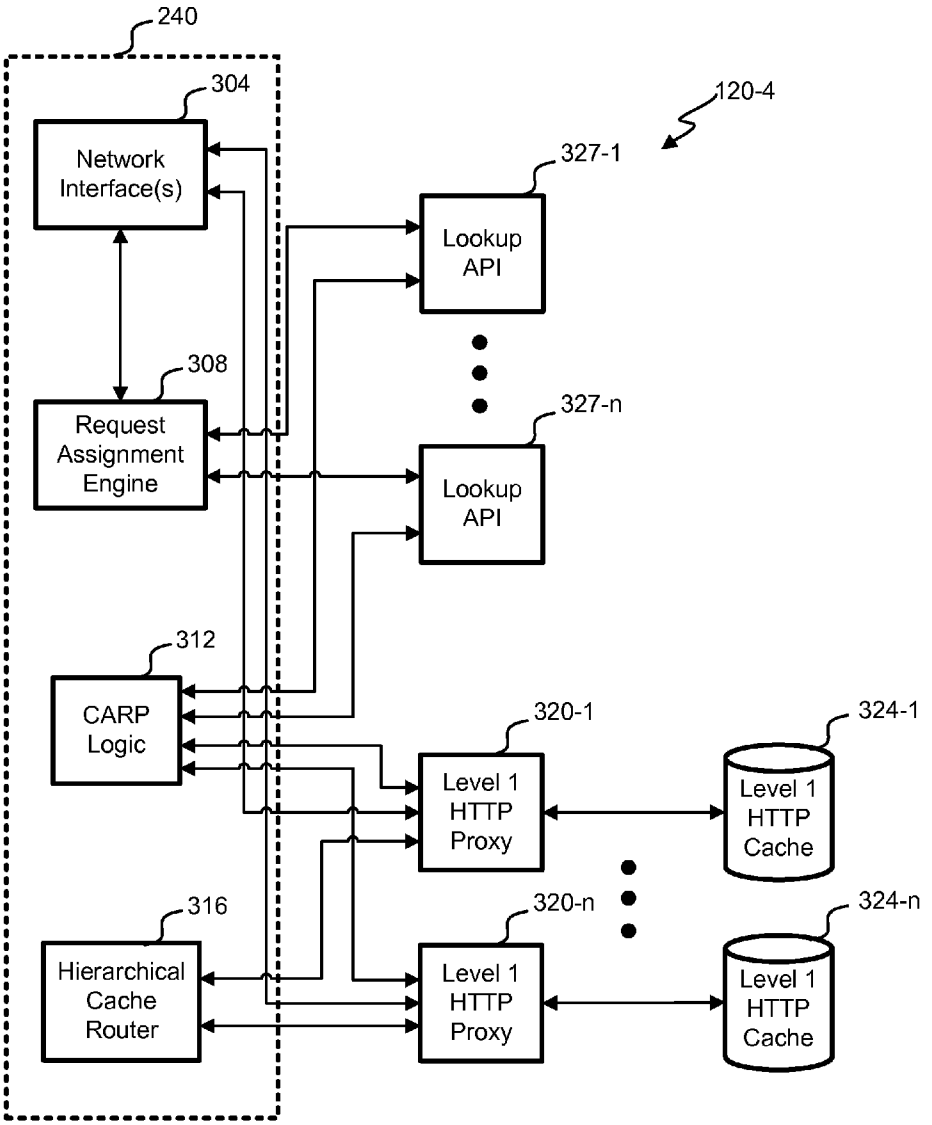

240
304
Network Interface(s)
120-4
327-1
Lookup API
308
Request Assignment Engine
327-n
Lookup API
312
CARP Logic
320-1
Level 1 HTTP Proxy
324-1
Level 1 HTTP Cache
316
Hierarchical Cache Router
320-n
Level 1 HTTP Proxy
324-n
Level 1 HTTP Cache

FIG. 3D

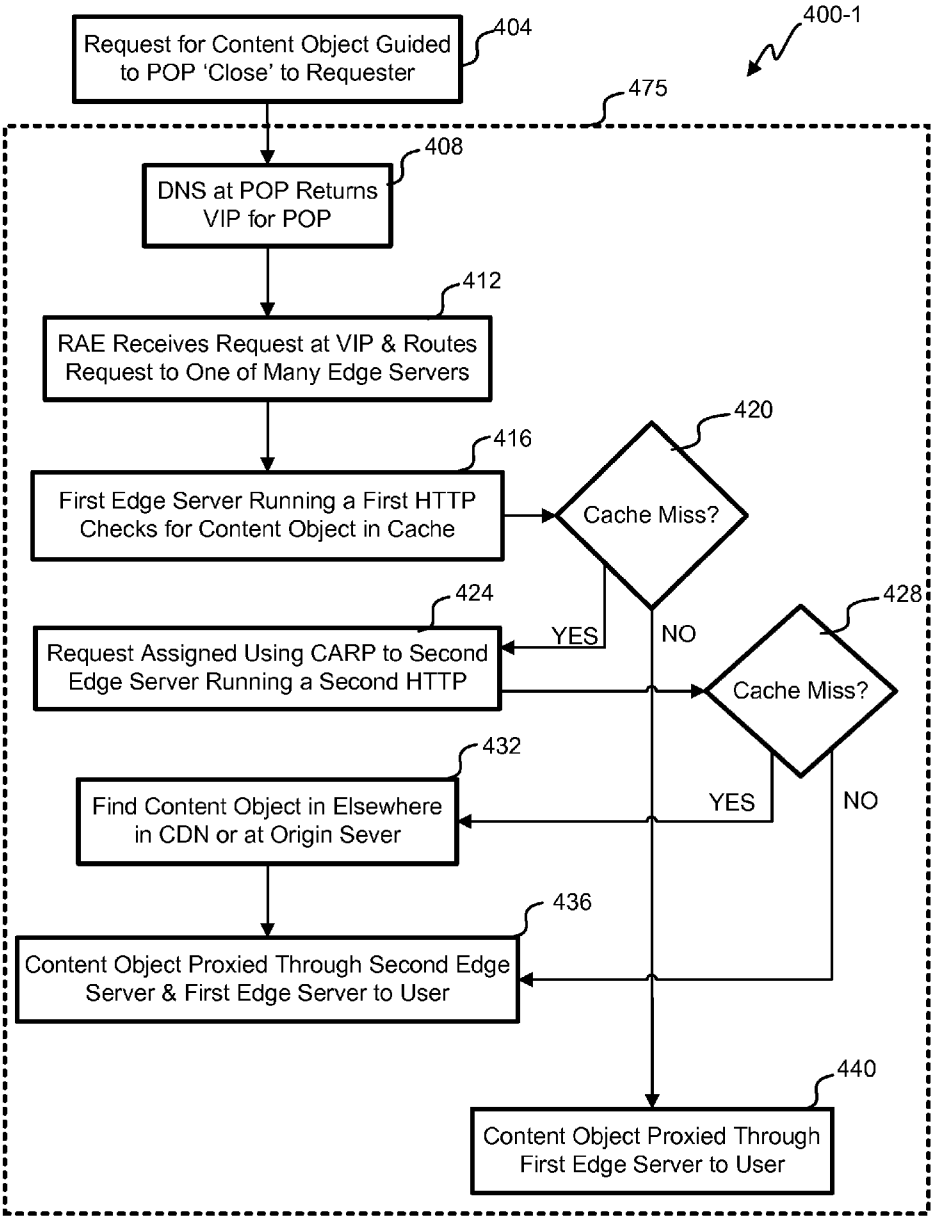

400-1
Request for Content Object Guided to POP 'Close' to Requester
404
475
DNS at POP Returns VIP for POP
408
RAE Receives Request at VIP & Routes Request to One of Many Edge Servers
412
First Edge Server Running a First HTTP Checks for Content Object in Cache
416
Cache Miss?
420
YES
NO
Request Assigned Using CARP to Second Edge Server Running a Second HTTP
424
Cache Miss?
428
YES
NO
Find Content Object in Elsewhere in CDN or at Origin Sever
432
Content Object Proxied Through Second Edge Server & First Edge Server to User
436
Content Object Proxied Through First Edge Server to User
440

FIG. 4A

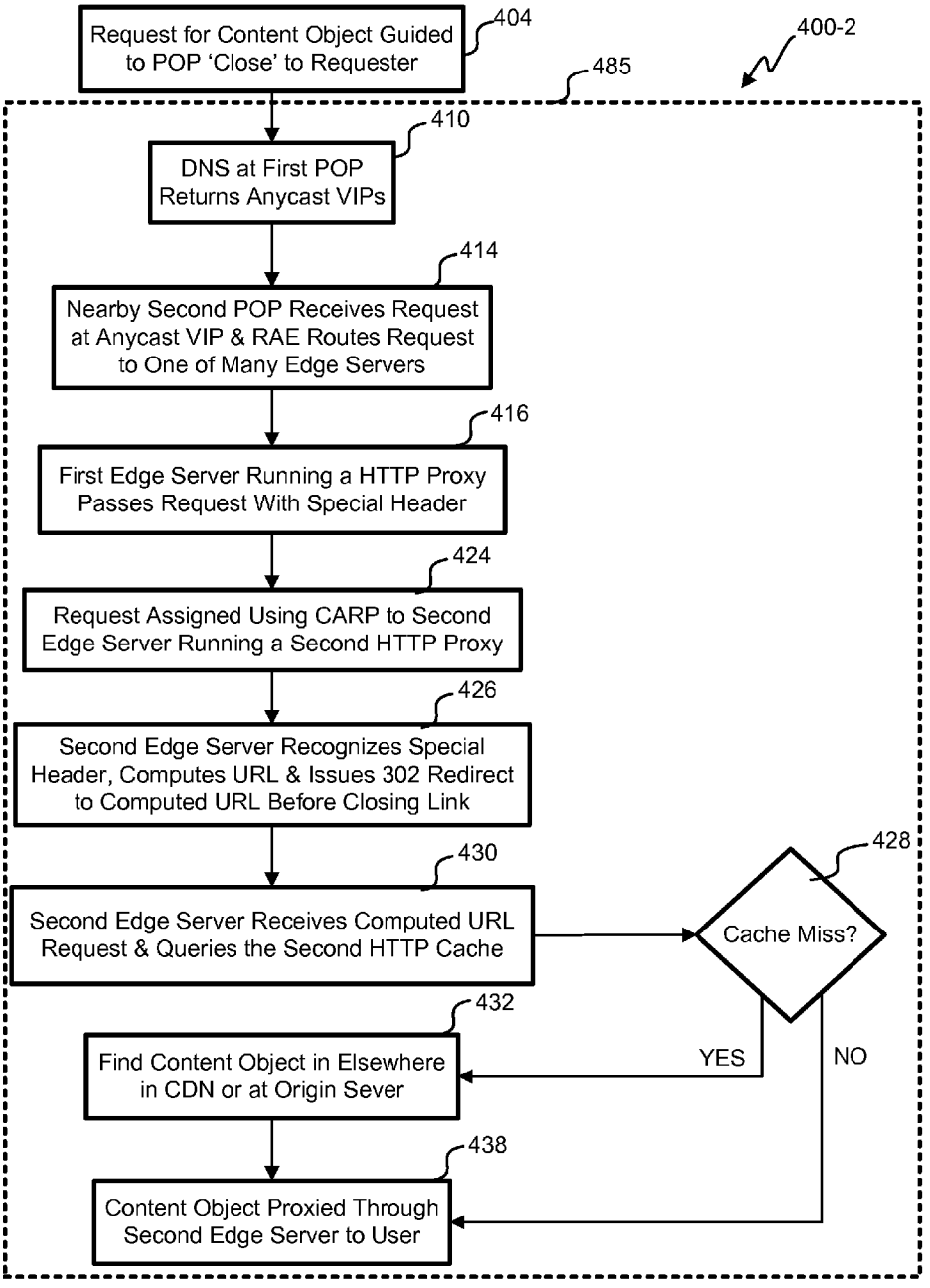

Request for Content Object Guided to POP 'Close' to Requester
404
400-2
485
410
DNS at First POP Returns Anycast VIPs
414
Nearby Second POP Receives Request at Anycast VIP & RAE Routes Request to One of Many Edge Servers
416
First Edge Server Running a HTTP Proxy Passes Request With Special Header
424
Request Assigned Using CARP to Second Edge Server Running a Second HTTP Proxy
426
Second Edge Server Recognizes Special Header, Computes URL & Issues 302 Redirect to Computed URL Before Closing Link
430
Second Edge Server Receives Computed URL Request & Queries the Second HTTP Cache
428
Cache Miss?
YES
NO
432
Find Content Object in Elsewhere in CDN or at Origin Sever
438
Content Object Proxied Through Second Edge Server to User

FIG. 4B

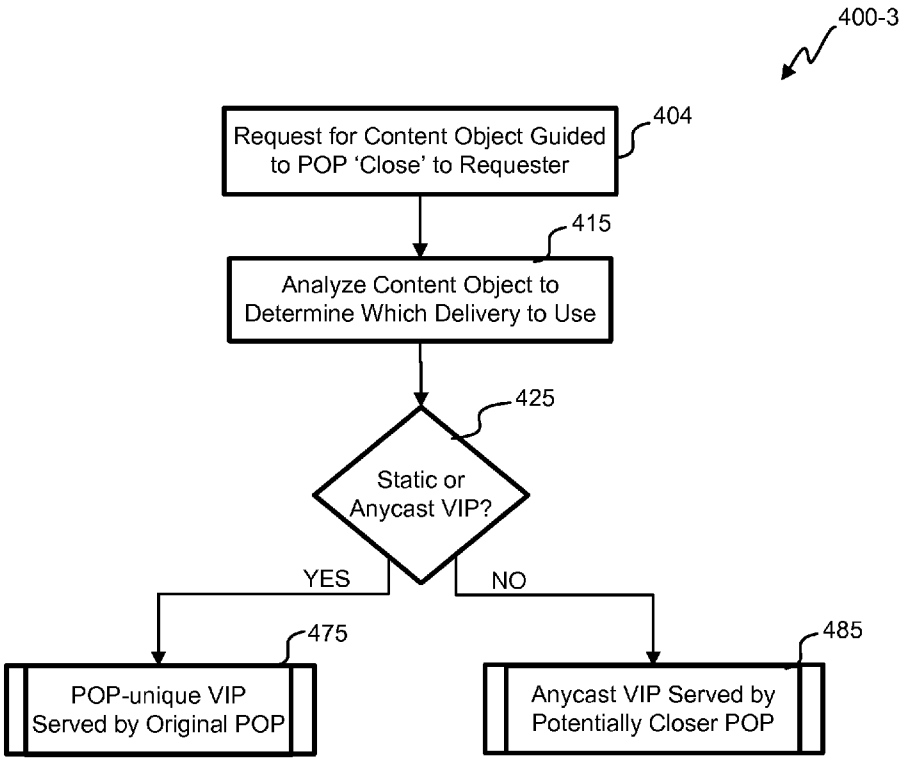

400-3
Request for Content Object Guided to POP 'Close' to Requester
404
Analyze Content Object to Determine Which Delivery to Use
415
Static or Anycast VIP?
425
YES
NO
POP-unique VIP Served by Original POP
475
Anycast VIP Served by Potentially Closer POP
485

FIG. 4C

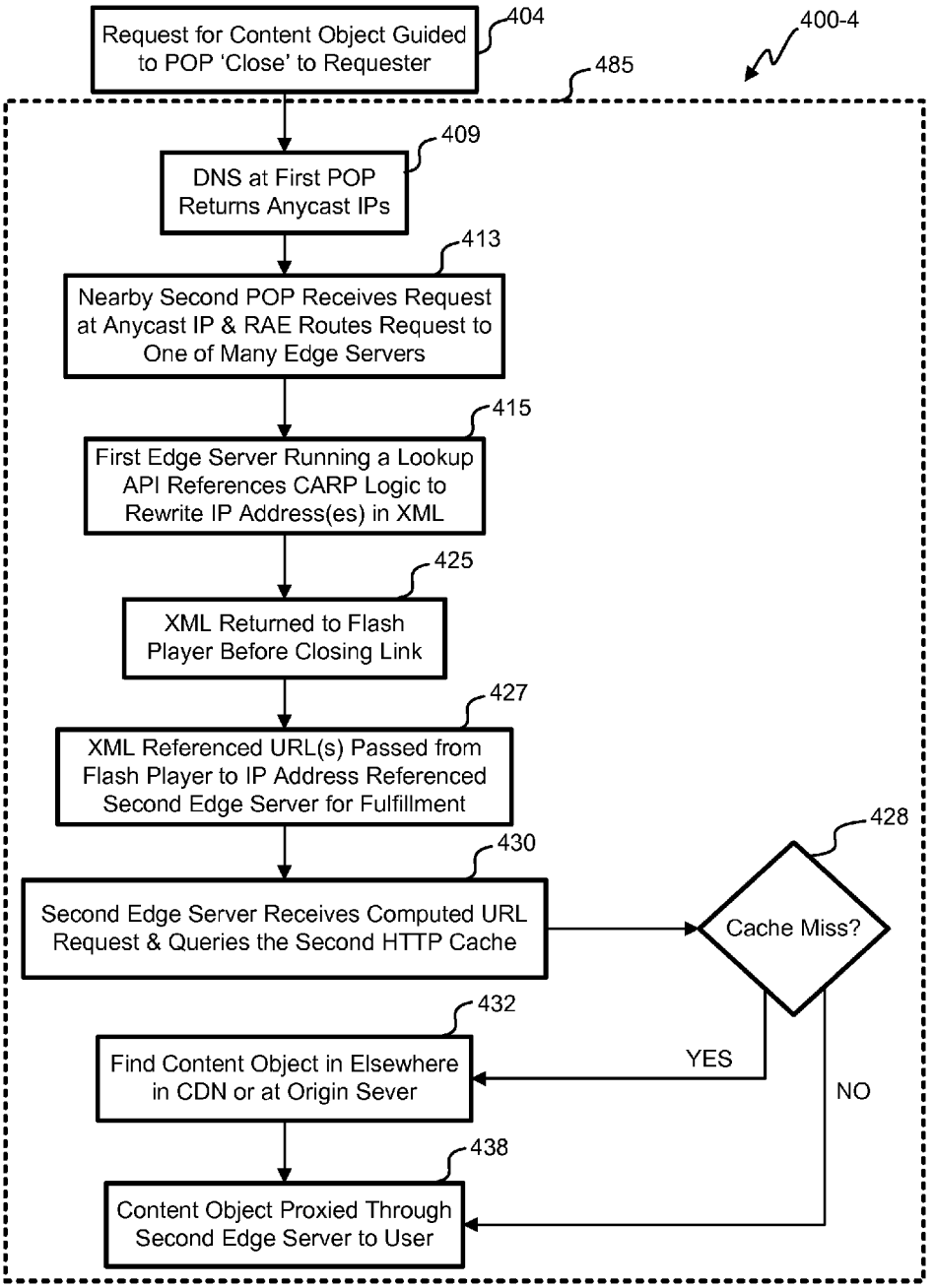

Request for Content Object Guided to POP 'Close' to Requester
404
400-4
485
DNS at First POP Returns Anycast IPs
409
Nearby Second POP Receives Request at Anycast IP & RAE Routes Request to One of Many Edge Servers
413
First Edge Server Running a Lookup API References CARP Logic to Rewrite IP Address(es) in XML
415
XML Returned to Flash Player Before Closing Link
425
XML Referenced URL(s) Passed from Flash Player to IP Address Referenced Second Edge Server for Fulfillment
427
Second Edge Server Receives Computed URL Request & Queries the Second HTTP Cache
430
Cache Miss?
428
YES
NO
Find Content Object in Elsewhere in CDN or at Origin Sever
432
Content Object Proxied Through Second Edge Server to User
438

FIG. 4D

ANYCAST REDIRECT TO UNICAST CONTENT DOWNLOAD

[0001] This application is a continuation of U.S. patent application Ser. No. 13/344,497, filed Jan. 5, 2012, entitled "ANYCAST REDIRECT TO UNICAST CONTENT DOWNLOAD," which is related by priority to PCT/US10/062156 filed Dec. 27, 2010, and Australian Patent No. 2011200629 issued Nov. 17, 2011, all of which are hereby incorporated by reference in their entirety for all purposes.

BACKGROUND

[0002] This disclosure relates in general to a content delivery network (CDN) and, but not by way of limitation, to selection of an edge server for a CDN.

[0003] CDNs will use Anycast to find a point of presence (POP) that can deliver content hosted or cached by the CDN. To find the POP, a user computer will query a domain name service (DNS) that may or may not be close in network terms to the user computer. A POP can be assigned that is inefficient because the DNS was located in an unexpected place on the Internet masking the network location of the user computer. An inefficient POP negatively affects quality of service (QoS) perceived by the user. Anycast DNS is desirable because it is fast.

[0004] Anycast can be used to choose the edge server. The edge server is often in close network proximity to the location of the user's DNS and hopefully the user computer. During download of large files or playback of long streams, clients can have problems staying connected to an assigned edge server. Unexpectedly changing to another edge server because of Anycast route changes or server selection can interrupt the download or stream.

[0005] Within a network, HTTP proxy caches are used to relay information. In some cases, there are multiple layers of HTTP proxy caches to move a content object. Each extra hop of multiple proxy layers in a route consumes bandwidth. Ethernet bandwidth is relative plentiful, but can still become overloaded.

SUMMARY

[0006] In one embodiment, the present disclosure provides a method and system for delivering content with a content delivery network (CDN) to a user computer. After an initial request to a point of presence (POP) of the CDN for the location of the content object, an Anycast Internet protocol (IP) address is assigned. The user computer requests the content object which may find its way to the same or a different POP depending on what is closer in Internet terms. The request returns a POP-specific IP address in a redirect such that the request resolves uniquely to the POP referenced in the immediately preceding sentence. One or more edge servers deliver the content object to the user computer.

[0007] In another embodiment, the present disclosure provides a method for assigning a server of a content delivery network (CDN). A request for a content object is received at a first point of presence (POP), which is one of a plurality of POPs that comprise the CDN and are distributed across the Internet. A first Internet protocol (IP) address is returned, which could route to any of the plurality of POPs using Anycast. The request is received at a second POP using the first IP address. A user computer is redirected to request the content object from the second POP with a second IP address unique to the second POP. The request is received at the second POP using the second IP address. The server associated with the second IP address is determined. The content object is served from the second POP using the server to the user computer.

[0008] In yet another embodiment, the present disclosure provides a machine-readable physical medium having machine-executable instructions, comprising code for:

[0009] receiving a request for a content object at a first point of presence (POP), which is one of a plurality of POPs that comprise the CDN and are distributed across the Internet;

[0010] returning a first Internet protocol (IP) address, which could route to any of the plurality of POPs using Anycast;

[0011] receiving the request at a second POP using the first IP address;

[0012] redirecting a user computer to request the content object from the second POP with a second IP address unique to the second POP;

[0013] receiving the request at the second POP using the second IP address;

[0014] determining the server associated with the second IP address; and

[0015] serving the content object from the second POP using the server to the user computer.

[0016] In still yet another embodiment, the present disclosure provides a CDN for delivering a plurality of content objects to user computers. The CDN includes: a plurality of POPs distributed across the Internet including a first POP and a second POP, a domain name service to return a first Internet protocol (IP) address, switch fabric that receives the request at a second POP using the first IP address, and a plurality of servers that redirect a user computer to request the content object from the second POP with a second IP address unique to the second POP. The first IP address could route to any of the plurality of POPs using Anycast. The first IP address is associated with a request for a content object at the first POP. The plurality of servers include a server the second POP receives the request at the second IP address. The switch fabric determines the server associated with the second IP address. The server delivers the content object from the second POP to the user computer.

[0017] Further areas of applicability of the present disclosure will become apparent from the detailed description provided hereinafter. It should be understood that the detailed description and specific examples, while indicating various embodiments, are intended for purposes of illustration only and are not intended to necessarily limit the scope of the disclosure.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] The present disclosure is described in conjunction with the appended figures:

[0019] FIG. **1** depicts a block diagram of an embodiment of a content distribution system;

[0020] FIG. **2** depicts a block diagram of an embodiment of a point of presence (POP);

[0021] FIGS. **3**A, **3**B, **3**C, and **3**D depict functional block diagrams of embodiments of a POP; and

[0022] FIGS. **4**A, **4**B, **4**C, and **4**D illustrate flowcharts of embodiments of a process for delivering a content object from a POP.

[0023] In the appended figures, similar components and/or features may have the same reference label. Further, various

components of the same type may be distinguished by following the reference label by a dash and a second label that distinguishes among the similar components. If only the first reference label is used in the specification, the description is applicable to any one of the similar components having the same first reference label irrespective of the second reference label.

DETAILED DESCRIPTION

[0024] The ensuing description provides preferred exemplary embodiment(s) only, and is not intended to limit the scope, applicability or configuration of the disclosure. Rather, the ensuing description of the preferred exemplary embodiment(s) will provide those skilled in the art with an enabling description for implementing a preferred exemplary embodiment. It is understood that various changes may be made in the function and arrangement of elements without departing from the spirit and scope as set forth in the appended claims.

[0025] Referring first to FIG. **1**, a block diagram of an embodiment of a content distribution system **100** is shown. The content originator **106** offloads delivery of the content objects to a content delivery network (CDN) **110** in this embodiment. The content originator **106** produces and/or distributes content objects and includes a content provider **108**, a content site **116**, and an origin server **112**. The CDN **110** can cache, redistribute and/or host content in various embodiments for third parties such as the content originator **106** to offload delivery and typically provide better quality of service (QoS).

[0026] In this embodiment, the content distribution system **100** locates the content objects (or portions thereof) and distributes the content objects to an end user system **102**. The content objects are dynamically cached or processed within the CDN **110** to improve the QoS without necessarily replicating the whole content object, unless subsequently requested by the end user **128**. A content object is any content file or content stream and could include, for example, video, pictures, data, audio, software, and/or text. The content object could be live, delayed or stored. Throughout the specification, references may be made to a content object, content, content stream and/or content file, but it is to be understood that those terms could be used interchangeably wherever they may appear.

[0027] Many content providers **108** use a CDN **110** to deliver the content objects over the Internet **104** to end users **128**. The CDN **110** includes a number of points of presence (POPs) **120**, which are geographically distributed through the content distribution system **100** to deliver content. Various embodiments may have any number of POPs **120** within the CDN **110** that are generally distributed in various locations around the Internet **104** to be proximate to end user systems **102**. Multiple POPs **120** use the same IP address such that an Anycast routing scheme is used to find a POP **120** likely to be close, in network terms, to the end user for each request. In addition to the Internet **104**, a wide area network (WAN) **114** or other backbone may couple the POPs **120** with each other and also couple the POPs **120** with other parts of the CDN **110**.

[0028] When an end user **128** requests a web page through its respective end user system **102**, the request for the web page is passed either directly or indirectly via the Internet **104** to the content originator **106**. The content originator **106** is the source or re-distributor of content objects. The content site **116** is a web site accessible by the end user system **102**. In one embodiment, the content site **116** could be a web site where the content is viewable with a web browser. In other embodiments, the content site **116** could be accessible with application software other than a web browser. In this embodiment, the content provider **108** directs content requests to a CDN **110** after they are made or formulates the delivery path by embedding the delivery path into the URLs for a web page. In any event, the request for content is handed over to the CDN **110** by using an Anycast IP address corresponding to one, two or more POPs **120**.

[0029] Once the request for a content object is passed to the CDN **110**, the request is associated with a particular POP **120** within the CDN **110** using the Anycast routing scheme. In some cases, the POP **120** found with Anycast is not particularly close to the end user system **102**, being only close to the DNS server used by the end user system **102**. The particular POP **120** may retrieve the portion of the content object from the content provider **108** if not already within the CDN **110**. Alternatively, the content provider **108** may directly provide the content object to the CDN **110** and its associated POPs **120** through pre-population, i.e., in advance of the first request. In this embodiment, the content objects are provided to the CDN **110** and stored in one or more CDN servers such that the portion of the requested content may be served from the CDN **110**. The CDN servers include edge servers that actually serve end user requests. The origin server **112** holds a copy of each content object for the content originator **106**. Periodically, the content of the origin server **112** may be reconciled with the CDN **110** through a cache, hosting and/or pre-population algorithm. Some content providers **108** could use an origin server within the CDN **110** to host the content and avoid the need to maintain an accessible copy of the content object.

[0030] Once the content object is retrieved from the origin server **112** by the CDN **110**, the content object is stored within the particular POP **120** and is served from that POP **120** to the end user systems **102**. Streamed content objects can have real time or near real time information or can be previously stored. The end user system **102** receives the content object and processes it for use by the end user **128** or an automated processing systems. The end user system **102** could be a personal computer, media player, handheld computer, Internet appliance, phone, IPTV set top, web server, processing system, streaming radio or any other device that receives and/or plays content objects. In some embodiments, a number of the end user systems **102** could be networked together. Although this embodiment only shows a single content originator **106** and a single CDN **110**, it is to be understood that there could be many of each in various embodiments.

[0031] With reference to FIG. **2**, a block diagram of an embodiment of a POP **120** is shown. There are a number of edge servers **230** coupled to each other and the Internet **104** and WAN **114** using switching fabric **240**. Edge servers **230** number in the thousands in a given POP **120**. They could be divided by function and/or customer. Loading algorithms can be used to divide load among the edge servers **230** in any number of ways. The edge servers **230** perform caching, streaming, hosting, storage, and/or other functions within the POP **120**. An edge server **230** is typically a rack-mounted computer that could have varying levels of processing power, memory and storage. Software running on the edge server **230** includes, for example, HTTP proxy caching, media servers, Flash™ servers, Java™ application servers, Silverlight™ servers, etc.

[0032] The switching fabric **240** is used for several functions. Incoming requests for content objects are routed to the edge servers **230** using the switching fabric. This could be done using routing, redirection or domain name service (DNS). In this embodiment, requests are assigned to a virtual Internet protocol (VIP) address of the POP **120** that the switching fabric **240** assigns to a particular edge server **230** to service. Load balancing, round-robin, and/or other techniques could be used by the switching fabric **240** to route requests to edge servers **230**.

[0033] Communication within the POP **120** also uses the switching fabric **240**. Edge servers **230** could have multiple software applications running that communicate with other edge servers **230**. For example, embodiments could have HTTP caching proxies arranged in different levels such that a cache on one edge server could communicate through another edge server to deliver a content object to an end user system **102**.

[0034] The DNS **250** is used to assign an IP address or a Virtual (VIP) address to requests for domains or CNAMEs. The VIP address could be unique to the POP **120** or shared among a number of POPs **120** to allow Anycast routing of the request to a possibly closer POP **120**. The DNS **120***s* as part of a domain resolution process that can include any number of DNS servers elsewhere on the Internet **104**.

[0035] Referring next to FIG. **3**A, a functional block diagram of an embodiment of various functions within a POP **120-1** is shown. There are two levels of HTTP proxy caching in this embodiment where each HTTP proxy has an HTTP cache. The HTTP proxy caching may be performed on edge servers. Switching fabric **240** includes network interface(s) **304**, a request assignment engine (RAE) **308**, cache array routing protocol (CARP) logic **312**, and a hierarchical cache router **316**. Although the CARP logic **312** is shown separately for illustration purposes, in this embodiment, each level 2 HTTP proxy **328** has its own CARP logic **312** that can deterministically locate the level 1 HTTP proxy **320** to use on a cache miss. Other embodiments could have a separate server with the CARP logic **312**.

[0036] The network interface **304** is coupled to the Internet **104** and/or WAN **114** to allow communication with the POP **120**. The RAE **308** uses routing, DNS, round robin, random assignment, and/or load balancing to assign content object requests between a number of level 2 HTTP proxies **328**. Each HTTP proxy **328** is coupled with a level 2 HTTP cache **332** that is generally small and configured for popular content. If a content object is found in the level 2 HTTP cache **332**, it is coupled to the end user system **102** through the level 2 HTTP proxy **328**.

[0037] Where the level 2 HTTP cache **332** does not store the content object, the level 2 HTTP proxy **328** uses the CARP logic **312** to be assigned a level 1 HTTP proxy **320** to locate the content object. The CARP logic **312** divides the name space between the level 1 HTTP proxy using a hash of the URL to deterministically find the level 1 HTTP proxy that handles requests for a particular content object. Other embodiments could divide the namespace by directory, customer, file type, etc.

[0038] The level 1 HTTP proxy **320** looks for the content object requested in its corresponding level 1 HTTP cache **324**, which is typically larger than the level 2 HTTP cache **332**. Where the content object is found on the level 1 HTTP cache **324**, it is relayed through the level 1 HTTP proxy **320** and the level 2 HTTP proxy **328**, etc. to the end user system **102**. Should the content object not be located on the level 1 HTTP cache **324**, the content object is located using the hierarchical cache router **316** and loaded onto the level 1 HTTP cache **324**. The hierarchical cache router **316** is used to locate the content object elsewhere in the POP **120**, CDN **110** or origin server **112**.

[0039] With reference to FIG. **3**B, a functional block diagram of an embodiment of various functions within a POP **120-2** is shown. This embodiment does not have caching for the level 2 HTTP proxy **328**. Content requests are assigned to the POP **120** using Anycast VIPs and converted to static VIPs unique to the POP **120**. The Anycast VIPs are common to all or some of the POPs **120** in the CDN **110**. When requested by the end user system **102** the are likely to find the POP **120** closest in Internet terms and likely to provide better QoS.

[0040] The request assignment engine **308** passes the content object request, typically a URL, to one of the level 2 HTTP proxies **328**. In this embodiment, the level 2 HTTP proxies **328** do not have an associated cache. The level 2 HTTP proxy **328** uses the CARP logic **312** to pass the request to the level 1 HTTP proxy **320**. The level 1 HTTP proxy **320** calculates a redirect URL that identifies the request uniquely and issues a **302**-redirect back to the end user system **102** before closing the connection. Some embodiments could use a 301-redirect instead of a 302-redirect.

[0041] The redirect will cause the end user system **102** to request the URL directly from the level 1 HTTP proxy **320** without using Anycast. The network interface **304** sends those requests direct to the appropriate level 1 HTTP proxy **320** when received. As before, the level 1 HTTP proxy **320** will search the level 1 HTTP cache **324** for the content object before resorting to the hierarchical cache router **316** on a cache miss.

[0042] Referring next to FIG. **3**C, a functional block diagram of an embodiment of various functions within a POP **120-3** is shown. This embodiment operates in two modes with one mode used for some content objects much as the embodiment of FIG. **3**A operates and another mode for other content objects much as the embodiment of FIG. **3**B operates. A determination is made in the level 2 HTTP proxy **328** as to whether a **302**-redirect will be used to get off the Anycast VIP. Other embodiments could make the determination in the RAE **308** or network interface **304**.

[0043] In this embodiment, small content objects stay on an Anycast VIP with two-level HTTP proxy and large content objects move to a VIP for the level 1 HTTP proxy **320** through the redirect process. The level 2 HTTP proxy **328** uses a code inserted into the URL to signal to the level 1 HTTP proxy **320** how to handle a particular request, but other embodiments could use a side channel, protocol handshake and/or physical path isolation. Beyond size deciding which mode, other embodiments could use application, format, encoding, customer, directory, level of QoS, loading of the HTTP proxies **320**, **328**, loading of the Ethernet connections in the POP, likelihood the Anycast VIP would reroute mid-delivery, etc. to decide which mode to operate in.

[0044] Referring next to FIG. **3**D, a functional block diagram of an embodiment of various functions within a POP **120-4** is shown. This embodiment redirects a request at Anycast IP address to an XML file that includes one or more URLs that are converted to POP-specific IP addresses. An Anycast IP or Anycast VIP address is received in a request and directed by the RAE **308** to one lookup API **327** chosen from a number of lookup APIs **327**. The lookup API **327** deter-

mines or retrieves XML associated with the request, for example, the XML could be a playlist having a number of URLs. The lookup API **327** interacts with the CARP logic **312** to determine which level 1 HTTP proxy **320** would be used for each URL. The CARP logic **312** could be integral to the lookup API **327**. The URLs are rewritten by the lookup API **327** and the XML is returned to the end user system **102**. Requests are made by a Flash™ player or other application by the end user system **102** using the URLs having the POP-specific IP addresses rather than the Anycast IP addresses that started the process.

[0045] With reference to FIG. **4**A, a flowchart of an embodiment of a process **400-1** for delivering a content object from a POP **120** is shown. This embodiment corresponds to the functional block diagram of FIG. **3**A. An end user system **102** through a browser, player or other application requests a content object that is hosted by the CDN **110**. Through DNS, redirection and/or routing, the request reaches a POP **120** close in Internet terms to the end user system **102**. More precisely, the POP **120** is likely close to the DNS used by the end user system **102**, which may be very distant from the end user system **102** such that the POP **120** is a poor choice for delivering the content object. This embodiment uses Anycast to find the POP **120** in block **404**. The DNS **250** at the POP **120** returns a VIP unique to the POP **120** that points to the edge servers.

[0046] With the VIP, the end user system **102** requests the URL with the VIP inserted. The RAE **308** receives the request before routing the request to one of the many edge servers **230** in block **412**. The first edge server **230** that receives the request checks a level 2 HTTP cache **332** for the content object in block **416**. Where the level 2 HTTP cache **332** has the content object as determined in block **420**, processing continues to block **440** where the content object is proxied through the level 2 HTTP proxy **328** running on the edge server **230** to the end user system **102**.

[0047] Where there is a cache miss in block **420**, processing continues to block **424** where the level 2 HTTP proxy **328** passes the request to a CARP logic **312**. The CARP logic **312** calculates a hash on the URL and finds one of the many level 1 HTTP proxies **320** corresponding to the hash. The level 1 HTTP proxy **320** checks its level 1 HTTP cache **324** for the content object. If the content object is missing, in whole or in part, it is found elsewhere in the CDN **110** or at the origin server **112** in block **432**. Once the content object is located in the level 1 HTTP cache **324** or found elsewhere, the content object is proxied through the level 1 HTTP proxy **320** and the level 2 HTTP proxy **328** to the end user system **102**.

[0048] With reference to FIG. **4**B, a flowchart of an embodiment of a process **400-2** for delivering a content object from a POP **120** is shown. This embodiment corresponds to the functional block diagram of FIG. **3**B. The depicted portion of the process **400-2** begins in step **404** where the request or domain name resolution is delivered to a POP **120** likely close to the end user system **102**. The DNS **250** at the POP **120** returns Anycast VIPs to the end user system **102**. The Anycast VIPs can resolve to any number of POPs **120** throughout the CDN **110**. The Anycast VIPs are different from IP addresses in that the RAE **308** could assign them to any number of edge servers **230**. Other embodiments could use Anycast IP addresses that reference servers that appear in a number of POPs.

[0049] The end user system **102** request the content object at the Anycast VIP. This will resolve though the Internet to a nearby POP **120**. The nearby POP **120** may or may not be the POP **120** that was used in block **410** for the DNS resolution. In block **414**, the received request is processed by the RAE **308** for assignment to one of many edge servers **230**. The RAE **308** in this embodiment does round robin assignment between a number of the edge servers **230** with the right capability that were assigned to the customer.

[0050] In block **416**, a level 2 HTTP proxy **328** on the edge server **230** passes the request with a special header indicating the request came from an Anycast VIP to the CARP logic **312**. Using CARP, the CARP logic **312** assigns the request to a second edge server **230** running level 1 HTTP proxy **320**. After recognizing the special header, the level 1 HTTP proxy **320**, a unique URL is computed that refers back to the level 1 HTTP proxy **320** in block **426**. A 302-redirect is issued to the end user system **102** to relay back the computed URL through the two HTTP proxies **328**, **320** before the link is closed.

[0051] The end user system **102** requests the computed URL to issue it back to the level 1 HTTP proxy **320** in block **430**. The level 1 HTTP cache **324** is also queried. On a cache miss determined in block **428**, the content object is found and loaded into the level 1 HTTP cache **324** in block **432**. For a cache hit or after the content object is loaded in the cache in block **432**, the content object is proxied through the level 1 HTTP proxy to the end user system **102**.

[0052] With reference to FIG. **4**C, a flowchart of an embodiment of a process **400-3** for delivering a content object from a POP **120** is shown. This embodiment corresponds to the functional block diagram of FIG. **3**C. After finding the POP in block **404**, the content object is analyzed to determine whether an Anycast VIP should be used or not in block **415**. The determination in block **425** for this embodiment uses an Anycast VIP for large content objects and a POP-unique VIP for small content objects. Depending on that determination, either a first group of blocks **475** from FIG. **4**A are performed or a second group of blocks **485** from FIG. **4**B are performed. Any number of factors can be used in the determination of block **425** to decide which mode to operate in.

[0053] A number of variations and modifications of the disclosed embodiments can also be used. For example, two banks of edge servers are shown in FIG. **2**, but it is to be understood that the banks could be the same bank. One hardware edge server **230** could simultaneously run both any number of level 2 HTTP proxy and any number level 2 HTTP proxy. Additional functions could also run on a hardware server such as DNS, RAE, CARP logic, etc. In another example, we discuss the Anycast VIPs changing from a first POP to a second POP, but that is not necessarily the case. Often, the POP chosen during DNS resolution is also the POP that will receive the redirected request.

[0054] With reference to FIG. **4**D, a flowchart of an embodiment of a process **400-4** for delivering a content object from a POP **120** is shown. The depicted portion of the process begins in block **404** where the request for content uses DNS, Anycast or routing to find a POP **120** close to the requestor or at least their DNS server. The DNS server **250** at the POP **120** returns Anycast IP addresses for the domain, but could return VIP addresses in other embodiments.

[0055] With the Anycast IP addresses, the end user device **102** requests the URL. In this embodiment, a Flash™ player on the end user device **102** is requesting a XML playlist containing a number of URLs that could be video or audio clips, for example. In block **413**, a second POP nearby the end user device **102** receives the request on the Anycast IP

address. The second POP could be the same as the first POP in some cases, but could be in a completely different geographic location in some instances. The RAE **308** routes the request to one of many lookup APIs **327**.

[0056] With the URL request, the lookup API **327** retrieves or formulates a XML file corresponding to the URL in block **415**. The URLs are rewritten after using the CARP logic **312** to determine the server(s) in the POP that would be servicing each URL. After conversion, the IP addresses are POP-specific. In block **425**, the rewritten XML is returned to the Flash player before closing the link on the Anycast IP address.

[0057] In block **427**, the URLs are passed from the Flash™ player to the POP-specific IP address for fulfillment. The level 1 HTTP proxy **320** will locate the content object referenced by the URL in the level 1 HTTP cache **324** or elsewhere and return it in blocks **430**, **428**, **432**, and **438**. For each URL these blocks **430**, **428**, **432**, **438** will reference the appropriate level 1 HTTP proxy **320** and return the content object. While this embodiment uses an XML file with redirection to POP-specific IP addresses, any type of file or control information could be used to redirect away from Anycast IP addresses in other embodiments.

[0058] Specific details are given in the above description to provide a thorough understanding of the embodiments. However, it is understood that the embodiments may be practiced without these specific details. For example, circuits may be shown in block diagrams in order not to obscure the embodiments in unnecessary detail. In other instances, well-known circuits, processes, algorithms, structures, and techniques may be shown without unnecessary detail in order to avoid obscuring the embodiments.

[0059] Implementation of the techniques, blocks, steps and means described above may be done in various ways. For example, these techniques, blocks, steps and means may be implemented in hardware, software, or a combination thereof. Also, it is noted that the embodiments may be described as a process which is depicted as a flowchart, a flow diagram, a data flow diagram, a structure diagram, or a block diagram. Although a flowchart may describe the operations as a sequential process, many of the operations can be performed in parallel or concurrently. In addition, the order of the operations may be re-arranged. A process is terminated when its operations are completed, but could have additional steps not included in the figure. A process may correspond to a method, a function, a procedure, a subroutine, a subprogram, etc. When a process corresponds to a function, its termination corresponds to a return of the function to the calling function or the main function.

[0060] Furthermore, embodiments may be implemented by hardware, software, scripting languages, firmware, middleware, microcode, hardware description languages, and/or any combination thereof. When implemented in software, firmware, middleware, scripting language, and/or microcode, the program code or code segments to perform the necessary tasks may be stored in a machine readable medium such as a storage medium. A code segment or machine-executable instruction may represent a procedure, a function, a subprogram, a program, a routine, a subroutine, a module, a software package, a script, a class, or any combination of instructions, data structures, and/or program statements. A code segment may be coupled to another code segment or a hardware circuit by passing and/or receiving information, data, arguments, parameters, and/or memory contents. Information, arguments, parameters, data, etc. may be passed, forwarded, or transmitted via any suitable means including memory sharing, message passing, token passing, network transmission, etc.

[0061] Moreover, as disclosed herein, the term "storage medium" may represent one or more memories for storing data, including read only memory (ROM), random access memory (RAM), magnetic RAM, core memory, magnetic disk storage mediums, optical storage mediums, flash memory devices and/or other machine readable mediums for storing information. The term "machine-readable medium" includes, but is not limited to portable or fixed storage devices, optical storage devices, wireless channels, and/or various other storage mediums capable of storing that contain or carry instruction(s) and/or data.

[0062] While the principles of the disclosure have been described above in connection with specific apparatuses and methods, it is to be clearly understood that this description is made only by way of example and not as limitation on the scope of the disclosure.

1. (canceled)

2. A delivery network for delivering a plurality of content objects to user computers, the delivery network comprising:

a plurality of POPs distributed across the Internet;

a first server configured to:

receive a Domain Name Service (DNS) request wherein:

the DNS request is from an end-user system;

the DNS request is for identifying a server to provide a first content object to the end-user system; and

the first server is located in a first POP of the plurality of POPs; and

return an anycast IP address to the end-user system;

a second server, the second server configured to:

receive a request for the first content object, wherein:

the request for the first content object is from the end-user system;

the request for the first content object uses the anycast IP address; and

the second server is in a second POP of the plurality of POPs;

identify that the first content object references two additional content objects, a second content object and a third content object; and

return a response to the end-user system, the response comprising:

a first Uniform Resource Locator (URL), wherein:

the first URL is used to locate the second content object;

the first URL resolves to a first IP address;

the first IP address is a unicast IP address; and

the first IP address identifies a third server; and

a second URL, wherein:

the second URL is used to locate the third content object;

the second URL resolves to a second IP address;

the second IP address is a unicast IP address;

the second IP address identifies a fourth server; and

the fourth server is different from the third server;

the third server, wherein the third server is configured to receive a request for the second content object; and

the fourth server, wherein the fourth sever is configured to receive a request for the third content object.

3. The delivery network for delivering a plurality of content objects to user computers as recited in claim **2**, wherein both the third server and the fourth server are in a third POP of the plurality of POPs.

4. The delivery network for delivering a plurality of content objects to user computers as recited in claim **2**, wherein the third server is the same as the second server.

5. The delivery network for delivering a plurality of content objects to user computers as recited in claim **4**, wherein the first POP is different from the second POP.

6. The delivery network for delivering a plurality of content objects to user computers as recited in claim **2**, wherein the first URL and the second URL are part of a markup language file returned to the end-user system.

7. The delivery network for delivering a plurality of content objects to user computers as recited in claim **2**, wherein returning the response to the end-user system by the second server is done before closing a link with the end-user system, wherein the link was created with the anycast IP address.

8. A method for assigning multiple servers of a delivery network for fulfillment of a single content request, the method comprising:

receiving a Domain Name Service (DNS) request, wherein:

the DNS request is from an end-user system;

the DNS request is for identifying a server to provide a first content object to the end-user system;

the DNS request is received at a first server in a content delivery network;

the content delivery network comprises a plurality of (Points of Presence) POPs distributed geographically; and

the first server is located in a first POP of the plurality of POPs;

returning an anycast IP address to the end-user system;

receiving a request for the first content object, wherein:

the request for the first content object is from the end-user system;

the request for the first content object uses the anycast IP address;

the request for the first content object is received at a second server; and

the second server is in a second POP of the plurality of POPs;

identifying that the first content object references two additional content objects, a second content object and a third content object; and

returning a response to the end-user system, the response comprising:

a first Uniform Resource Locator (URL), wherein:

the first URL is used to locate the second content object;

the first URL resolves to a first IP address;

the first IP address is a unicast IP address; and

the first IP address identifies a third server; and

a second Uniform Resource Locator (URL), wherein:

the second URL is used to locate the third content object;

the second URL resolves to a second IP address;

the second IP address is a unicast IP address;

the second IP address identifies a fourth server; and

the fourth server is different from the third server.

9. The method for assigning multiple servers of the delivery network for fulfillment of the single content request as recited in claim **8**, wherein:

the third server is in a third POP of the plurality of POPs;

the fourth server is in fourth POP of the plurality of POPs; and

the third POP is different than the fourth POP.

10. The method for assigning multiple servers of the delivery network for fulfillment of the single content request as recited in claim **8**, wherein the third server is the same as the second server.

11. The method for assigning multiple servers of the delivery network for fulfillment of the single content request as recited in claim **8**, wherein the method further comprises receiving a request for the second content object at the second POP.

12. The method for assigning multiple servers of the delivery network for fulfillment of the single content request as recited in claim **8**, wherein the first URL and the second URL are part of markup language file returned to the end-user system.

13. The method for assigning multiple servers of the delivery network for fulfillment of the single content request as recited in claim **12**, further comprising terminating a link with the end-user system, wherein:

the link was formed with the end-user system using the anycast IP address;

terminating the link is performed after returning the response to the end-user system;

terminating the link is performed before the third server receives a request from the end-user system for the second content object; and

terminating the link is performed before the fourth server receives a request from the end-user system for the third content object.

14. The method for assigning multiple servers of the delivery network for fulfillment of the single content request as recited in claim **8**, wherein returning the response to the end-user system is performed by the second server.

15. A computer-memory device having instructions that when executed perform the following steps for assigning multiple servers of a delivery network for fulfillment of a single content request:

receive a request for the first content object, wherein:

the request for the first content object is from an end-user system;

the request for the first content object uses an anycast IP address;

the anycast IP address was returned to the end-user system from a first server;

the first server is in a first point of presence (POP) of a plurality of POPs;

the plurality of POPs is distributed geographically;

the request for the first content object is received at a second server; and

the second server is in a second POP of the plurality of POPs;

identify that the first content object references two additional content objects, a second content object and a third content object; and

return a response to the end-user system, the response comprising:

a first Uniform Resource Locator (URL), wherein:

the first URL is used to locate the second content object;

the first URL resolves to a first IP address;

the first IP address is a unicast IP address; and
the first IP address identifies a third server; and
a second URL, wherein:
the second URL is used to locate the third content object;
the second URL resolves to a second IP address;
the second IP address is a unicast IP address;
the second IP address identifies a fourth server; and
the fourth server is different from the third server.

16. The computer-memory device having instructions for assigning multiple servers of a delivery network for fulfillment of a single content request as recited in claim **15**, the instructions further performing the following steps when executed:

receive a Domain Name Service (DNS) request, wherein:
the DNS request is from an end-user system;
the DNS request is for identifying a server to provide the first content object to the end-user system; and
the DNS request is received at the first server; and
return the anycast IP address to the end-user system.

17. The computer-memory device having instructions for assigning multiple servers of a delivery network for fulfillment of a single content request as recited in claim **15**, wherein the fourth server and the third server are in different POPs of the plurality of POPs.

18. The computer-memory device having instructions for assigning multiple servers of a delivery network for fulfillment of a single content request as recited in claim **15**, wherein the first POP is the same as the second POP.

19. The computer-memory device having instructions for assigning multiple servers of a delivery network for fulfillment of a single content request as recited in claim **15**, wherein the first URL and the second URL are part of a markup language file returned to a player on the end-user system.

20. The computer-memory device having instructions for assigning multiple servers of a delivery network for fulfillment of a single content request as recited in claim **15**, wherein returning the response to the end-user system is performed after closing a link with the end-user system, wherein the link was formed with the end-user system using the anycast IP address.

21. The computer-memory device having instructions for assigning multiple servers of a delivery network for fulfillment of a single content request as recited in claim **15**, wherein the second POP is closer to the end-user system than the first POP.

* * * * *