Comparative Study of Content-Centric vs. Content Delivery Networks

Seongik Hong, Byoung-Joon (BJ) Lee*
Samsung Advanced Institute of Technology
Giheung, Korea
seongik.hong@samsung.com
* Creatrix Design Group, Ottawa, Canada
bjlee@creatrix.com

Chang-Mo (C.M) Yoo, Mi-Sun Do, JangWoo Son
NMC Consulting Group, Inc
Seoul, Korea
{cmyoo, misun.do, son}@netmanias.com

Abstract

Content-Centric Networking (CCN) utilizing network caches is architected to bring significant advantages over current IP-based Internet, especially in reducing the traffic by eliminating redundant data transmissions. However, application layer solutions such as *Content Distribution Networks (CDNs)* that utilize application caches overlaid on top of IP have already been widely deployed to cope with explosive growth of Web traffic. This paper analyzes the potential benefits of CCN over CDN in both qualitative and quantitative manner, also considering feasible CCN router implementation and typical CDN deployment topologies. This is the first such comparison study to date. The results show that CCN provides significant advantages over CDN in costs on network, H/W and S/W, congestion/flow control, traffic engineering, security and etc.

I. Introduction

Today's IP-based Internet is considered the most profound invention since its creation in 1960s. With IP, a packet in the network layer is delivered from a source to a destination using the destination IP address. Due to this address-based delivery scheme, the traffic explosion problem of the current Internet is exacerbated since the duplicated requests for popular contents generate yet more redundant traffic.

To remedy such problem, the Content-Centric Networking (CCN) [1] is proposed which is to replace "*where*" with "*what*". With IP, a packet is delivered from a source to a destination, completely unaware of what the source sends or the destination wants. However, these days the majority of current Internet usage consists of data being disseminated from a small number of sources to a great number of users. In this condition, heavy traffic congestion occurs in the most upstream links towards the servers due to the repeated requests on the relatively small number of popular contents.

The design philosophy of CCN architecture is aligned with such network usage patterns. With CCN, a packet is delivered by the requested content name, not the address. In addition, any intermediate node can reply to the data request packet as long as it has the data in its cache. In this way, CCN can reduce the congestion near the content server by eliminating the redundant data transmissions from the server.

Content Distribution Network (CDN) or Content Delivery Network [3][4][5] also aims to alleviate the concentration of the web request traffic near the servers by rerouting the requests towards the surrogate cache servers deployed near the end users [45], thus enhancing user experience with reduced content access delay and high availability. CDN already serves a large fraction of the Internet content today but runs as an application overlay over the current IP infrastructure.

Since both CCN and CDN address the same problem space but at the different layers of the networking protocol stack, the comparative analysis of various aspects of CCN and CDN benefits has been one of the high priority issues in the future Internet research community [6][39][46]. However, such comparison has remained one of the non-trivial research challenges due to the following reasons: 1) there is no real CCN deployment with referenceable router implementation, 2) specific CDN configuration in commercial deployment is generally not available to the public and, 3) CCN and CDN architectures are continually evolving.

In this paper, we propose a most likely reference implementation of CCN router architecture, a specific CDN topology configuration and evolution scenario of a CDN architecture for both qualitative and quantitative comparison with CCN.

We describe the architecture and operation of CCN and CDN in Sections 2 and 3. In sections 4 through 7, we compare CCN and CDN from qualitative and quantitative viewpoints. Finally, Section 8 provides concluding remarks with a discussion on the future works.

II. Content Centric Network

With CCN [1], a packet has a requested content name in its header, not the IP address of a destination node. Routers in the network have content cache and store contents in the cache, so the intermediate routers can answer the request packets instead of the end nodes. In this way, CCN is thought to dramatically reduce the amount of redundant traffic. Fig. 1. shows how IP and CCN contents request packets are replied, respectively. The IP packet should go to the contents server since in its header the IP address of the contents server is written and the packet is routed by that. However, since the CCN packet is routed by the contents name in its header, if the router cache contains the contents indicated by the header, the router can reply with the corresponding contents.

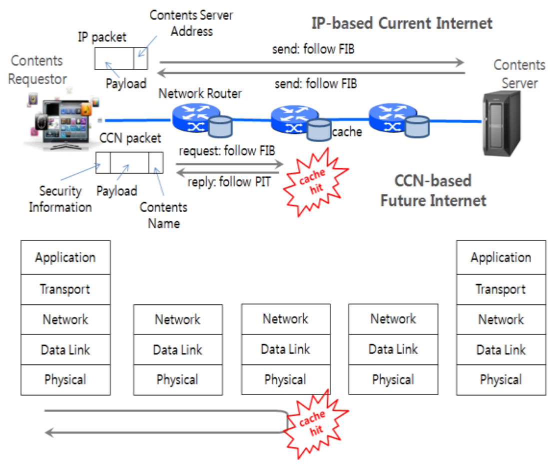

Fig. 1. IP packets should go to the server the destination address specifies. CCN requests can be answered by any intermediate routers containing the requested contents.

Fig. 2. describes the basic CCN forwarding engine model [1]. When a CCN router receives an Interest packet, it first checks its *Content Store (CS)*. If the requested content exists, it returns the content to the source. Otherwise, it checks whether the same entry exists in the *Pending Interest Table (PIT)*. If so, it adds the arrival face to the existing PIT entry. The PIT keeps track of Interests forwarded upstream toward content sources so that returned data can be sent downstream to its requestors. If there is no corresponding entry in PIT, it forwards the Interest packet to a face according to the matching *Forwarding Information Base (FIB)* entry. The FIB is used to forward Interest packets toward potential sources of matching data.

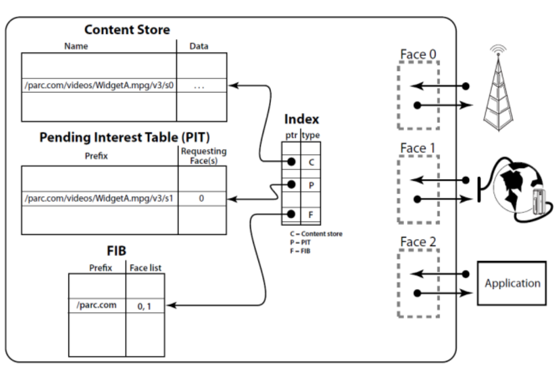

Fig. 2. CCN Forwarding Engine Model [1].

III. CONTENT DELIVERY NETWORK

CDN intends to improve Web performance by providing cached contents at geographically dispersed edge servers to the requestors. Content owners make contracts with CDN providers to distribute their contents more efficiently. Clients' requests can be forwarded to the nearby surrogate CDN servers by using Domain Name System (DNS) redirection or Uniform Resource Locator (URL) rewriting techniques [3]. There are several types of CDN providers as follows [8].

A. Pure Play CDN [4]

Some CDN companies operate cache servers around the world and provide CDN services to customers who want to distribute their contents faster to the user locations. Akamai Technologies, Inc. [5] is one of the dominant CDN market leaders, and its network is considered as one of the world's largest CDN platforms serving large amount of web traffic. We call a CDN provider such as Akamai *Pure Play CDN*. Usually, in this type of CDN, DNS redirection techniques which will be explained in the later sections are used and services are provided by the CDN servers scattered geographically.

B.Operator IPTV CDN [7]

As network operators want to extend their services beyond the communication and networking area, they begin to provide IPTV services utilizing their existing network infrastructure such as Asymmetric Digital Subscriber Line (ADSL) and cables. They offer large amount of contents from various sources, and thus are interested in leveraging the CDN technology. The topology and operations of this type of CDN is similar to the pure play CDN except that it is operated by IPTV carriers. This type of CDN is usually geographically restricted inside the service area of a carrier.

C. Wholesale CDN [8]

In a wholesale CDN model, the network operator uses their CDN facilities to offer B2B content delivery services to content providers seeking to deliver online contents to users such as local broadcasters. The topology and operations are exactly same as the pure play CDN. The only difference is that in this type of CDN network operators run the CDN instead of the content providers.

D. Operator Transparent Caching [9]

With transparent caching, content is stored and served from the edge node of the operator's network. The node does the deep packet inspection and help save transit network bandwidths and accelerating contents delivery to the subscriber. In operator transparent caching, there is no functionality of re-direction to the contents servers. If a request packet happens to pass by the specific edge node equipped with cache, the request can be answered by that node. Thus it is hard to be considered as a CDN.

E.Mobile CDN [10]

A mobile content delivery network or mobile content distribution network (Mobile CDN) is a network of servers that save contents in their storage and reply data back to end users instead of the destination on any type of wireless or mobile network. The operation is almost same as the operator transparent caching.

In this paper, we use the pure play CDN model to compare with CCN since the mechanism of the operator IPTV CDN and wholesale CDN models are mostly similar to that of the pure play CDN model. The operator transparent caching and the mobile CDN can also be categorized as a very simplified version of the pure play CDN model with no re-direction functionality. For this reason, in the remainder of this paper, the term CDN refers to the pure play CDN model.

IV. Qualitative Comparison

With the current generation of CDN, cache hits occur at the object level with the typical size of 1~2GB download. But the network geography cannot be optimally utilized, since there is no direct support from the routers. With CCN, however, cache hits occur at the packet level which enables concurrent and fast content retrieval from multiple sources. Furthermore, since every network device can store the contents, the optimal network geography can be achieved [39]. TABLE I. summarizes the qualitative comparison results between CCN and CDN. It highlights that CDN operates at the application layer and thus it is slower, more complex, and more cumbersome to manage.

TABLE I. Qualitative Comparison

	CCN	CDN	Benefits of CCN
Operating layer	Layer 3	Layer 7	Less processing overhead with less number of layers
Overlay	No	Yes	Less delay with less number of layers
Universal layer	Yes	No	CDN has to handle more than 60 application protocols [2]
Protocol update	No	Yes	Many different types of CDNs need to be updated separately
Congestion /flow control	Yes	No	CCN's hop-by-hop control enables easier support.
Traffic engineering	Yes	No	CCN's hop-by-hop control enables easier support.
Security	Content-based	Session -based	Session-based security is temporary
Mobility	Yes	No	Communication using content name, not address

In the following, each comparison item in the table is discussed in more detail. We view that for the comparisons of congestion/flow control, traffic engineering, security and mobility, it is more reasonable to compare CCN with IP networks since those properties are related to the IP *path* where the CDN servers rely on.

A. Operating Layer, Overlay

Fig. 3. shows a schematic diagram of CCN and CDN, illustrating the main difference in how the clients' request is processed in CCN and CDN.

Akamai CDN service employs DNS redirection using DNS servers [3]. DNS servers translate client's request to the IP address of the nearest Akamai content cache server. The detailed process of DNS translation for Akamai network is as follows. First, a user's web browser sends a HTTP (HyperText Transfer Protocol) request to the content origin server. The user's DNS server redirects the request to the Akamai's DNS servers. Then the Akamai's DNS server responds to the DNS name-translation request. Finally, the customer sends the request to the Akamai edge server which can best serve the request.

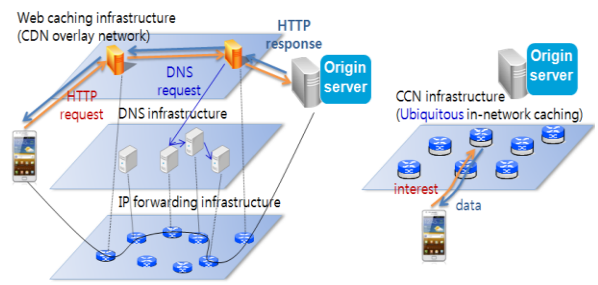

Fig. 3. CCN vs. CDN: Concept Comparison. CCN is operated in the network layer, it causes less processing delay.

For CCN networks, a user's request for contents is forwarded to the origin server using the routing tables at the routers. And the requests can be answered by any intermediate routers as long as the router has cached the corresponding data. Thus, the routing process for the CCN is much simpler than that of CDN since it is processed in the same layer and there is no additional procedure such as request routing or translation.

B. Universal Layer, Protocol Update

Network layer is the only layer that requires universal agreement. Since CDN is a network in the application layer which does not require universal agreement, almost every vendor of CDN operates their own protocol. It means it is very difficult to connect different CDNs even though there are new attempts to connect them [15].

Besides the compatibility issue between different types of CDNs, it causes a lot of maintenance cost. Wikipedia reports that there are more than 60 types of application protocols [2]. Thus, in terms of protocol update cost, it is obvious that CCN is quite a win.

C. Congestion/Flow Control

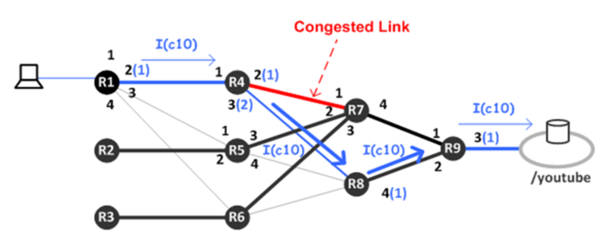

Fig. 4. Link congestion case in CCN.

CCN can employ both preventive and reactive congestion control mechanisms by utilizing multi-path and strategy layer capabilities [1]. When a node recognizes a high possibility of impending or existing congestion, the node can change the outbound faces for request transmission. For example, as the node R4 in Fig. 4. recognizes the congestion occurring on the link between R4 and R7, it can immediately reroute the content request packet to R8.

Compared with the functionality of CCN, with IP networks, routers keep pushing packets even to the already congested links since Open Shortest Path First (OSPF) routing algorithm [25] does not care about the network overload status.

D. Traffic Engineering

Internet Traffic Engineering (TE) is defined as the performance evaluation and optimization [20]. For the performance optimization, the control of Internet traffic is mandatory. The current IP networks overlaid by CDNs do not support those control such as multi-path forwarding, thus load balancing across multiple links are not feasible. Equal-Cost Multi-Path (ECMP) [40] has been proposed to address the lack of mutli-path capability, but it does not support unequal cost paths. Multi-Protocol Label Switching (MPLS) [41] protocol has been deployed to compensate for the weakness of IP protocol in supporting TE, however, it still incurs disadvantages as summarized in the below TABLE II.

TABLE II. SUMMARY OF TRAFFIC ENGINEERING COMPARISON

	CCN	MPLS
Coverage	end-to-end	Provider Edge-to-Provider Edge
Path provision	Automatic	manual tunnel setup (CSPF[42], RSVP-TE[43])
Scalability	O(1) due to hop by hop setup	$O(n^2)$ due to mesh-like tunneling, where n denotes # of nodes
Path optimality	Local optimal	Global optimal
Protocol complexity	No additional protocol required	Additional control protocols, e.g., OSPF-TE[44], CSPF, MPLS-TE, RSVP-TE
Re-routing capability	Yes	No

Thus, IP networks require MPLS to support TE but CCN can inherently support TE functionality.

E. Security

CCN architecture is based on the concept of content-based security [1][36]. Content-based security is fundamentally different from the channel-based security approach of IP networks overlaid by CDNs, since the protection and trust management goes with the content itself. In the channel-based security scheme, e.g., IPsec [37], we can only protect data flows between a pair of endpoints, not the data itself. It is not a complete protection since the channel protection may temporarily exist and protect the data only in a specific channel. So its protection is limited. The CCN approach enables persistent protection at the level of individual content due to the contents-based security mechanism.

Thus CCN inherently provides security features; IP networks do not support security. Even with IPSec, IP networks provide limited security features.

F. Mobility

CDN running on top of IP infrastructure does not yield easily to the support of mobility since with IP a sender should talk to a destination using the location 'address'. CCN operates only on the named data, not on the location of nodes, so it does not need to obtain or bind IP address to a layer 2 MAC address [1]. Even when either the sender or receiver, or both of them are rapidly moving, CCN can always exchange data without the need of updating their changing addresses. Hence, CCN inherently supports mobility.

V. QUANTITATIVE COMPARISON

The quantitative comparison between CCN and CDN has remained a non-trivial challenge, especially from the viewpoint of network deployment, H/W and S/W cost since CCN routers are not currently available. In this section, we first describe a feasible CCN router architecture. Then, the IP and CCN network deployment scenarios are compared quantitatively. For cost comparison, we use Korean Won as the unit. As of June 2014, the exchange rate of 1.00 US dollar is around 1,000 Korean Won.

A. A Reference Network Model

A reference network model for quantitative comparison is shown in Fig. 5. , which consists of a core network with edge routers (collocated with CDN edge servers) and access networks. The topology and various parameter values are determined based on the information collected from many talks [47] and papers [11][12]. Typical Internet Service Provider (ISP) sub-graph for the core and access networks tend to be configured in the mesh and tree topologies, respectively. The validity of our reference network model in faithfully representing the current ISP network topology has also been verified by many operator experts.

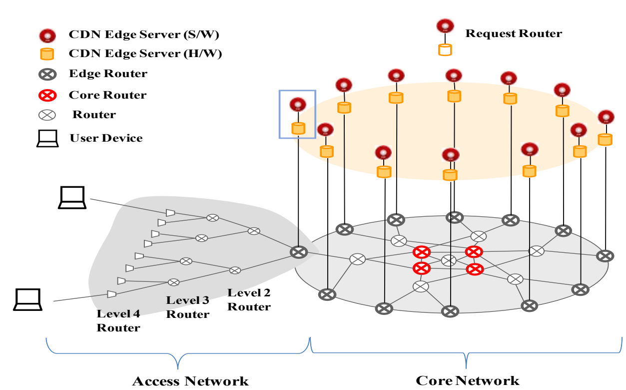

Fig. 5. A Reference Network Model.

The following network parameter values are used throughout the paper: the total number of subscribers – 3million [13], content encoding rate – 8Mbps (for High Definition TV), concurrent usage rate – 10%, number of CDN edge servers – 30. Each edge server holds 20TB of contents. The content popularity observes Zipf's law [35] with the exponent *s*. Zipf's law is a mathematical expression to describe a power law, meaning that the probability of attaining a certain size x is proportional to x^{-s}. These assumptions are valid throughout this paper.

B. Network H/W cost

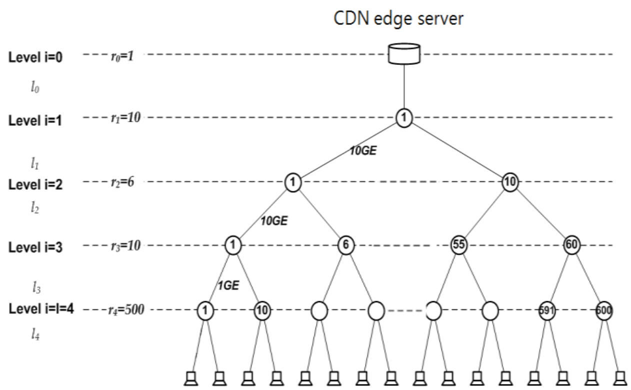

Fig. 6. A reference access network. The numbers in the circles represent the number of node, for example, there are 600 nodes at level 4.

Fig. 6. shows a detailed tree architecture of one access network shown in Fig. 5. It is assumed that one CDN edge server is placed over the level 1 router which connects to ten level 2 sibling routers. The number of sibling routers for levels 2 and 3 are 6 and 10, respectively, so that the total number of access routers at level 4 is 600. The uplink capacity of access routers at each level is also shown to be 10Gbps at level i=2 and 3, and 1Gbps at level i=4.

In Fig. 7. , the total amount of network bandwidth required for the given access network configuration shown in Fig. 6. is compared when there are 10,000 concurrent users. Note that in the CDN case there is no cache server between level 1 and level 4. In the CCN case, every router in each level (i = 1,2,3 and 4) is allocated pre-assigned amount of caches.

It shows that as the cache size allocated to the routers nearer to the users increase (8:4:2:1 → 1:2:4:8, the numbers represent the ratio of content amount cached at each level, *i* = 1,2,3 and 4), the total amount of network bandwidth required decreases. In other words, most of the content requests from the users can readily be served from the cached contents at the lower level routers. In this case, however, the total amount of allocated cache size should also increase since the number of routers increases at the lower level (i.e., i → 4).

As the Zipf's parameter *s* becomes bigger, the required bandwidth becomes even smaller with CCN since the popularity of contents becomes more skewed and more of the content requests are served from the low level router caches.

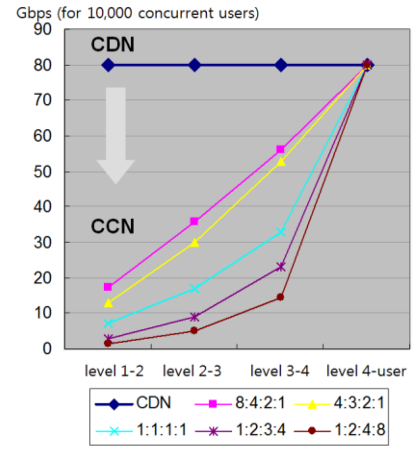

Fig. 7. Total amount of required network bandwidth. The ratio represents the caching capacity, for example, 4:3:2:1 means each level *i* (*i* = 1,2,3 and 4), node can store 40%, 30%, 20% and 10% of the total contents.

We have consulted multiple H/W vendors to get the typical price quotes for the routers and optical devices deployed for access networks in Korea. For the level 1 and 2 routers, CISCO 7600 and Juniper MX series [27][28] are mostly mentioned and we have chosen Junipher MX960 [29] high-end edge router for our cost analysis. For level 3, Cisco and Hitachi devices are used. For level 4 OLT (Optical Line Terminal), Huawei and Dasan are considered major vendors [30][31] and we selected Dasan V5724G [31] in our case study. For OADM (Optical Add-Drop Multiplexer), JDS Uniphase and SNH are referred to as big players and the SNH OADM 400G [33] is selected as a representative device in our study. The estimated price quotes are listed along the device configuration diagram shown in Fig. 8.

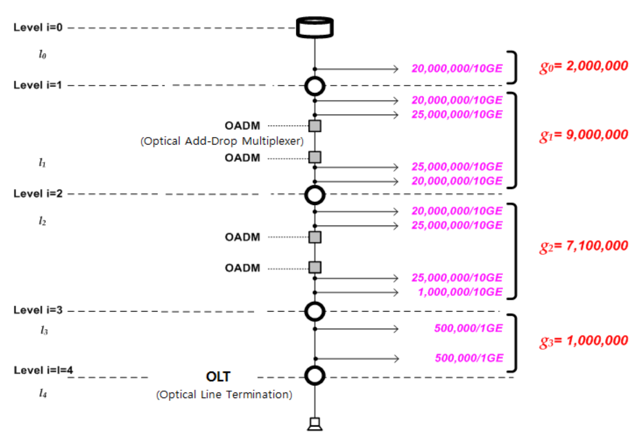

Fig. 8. Prices for the reference access network devices.

Now we are able to calculate the required bandwidth to provide the service. Since we know the network device cost per bandwidth, we can compute the overall network device cost to support the services by simply multiplying them.

Of course these data on equipment vendors and prices are from only one of the carriers, we guess the results are not far from the average conditions of most other carriers.

C. Server/Router H/W cost

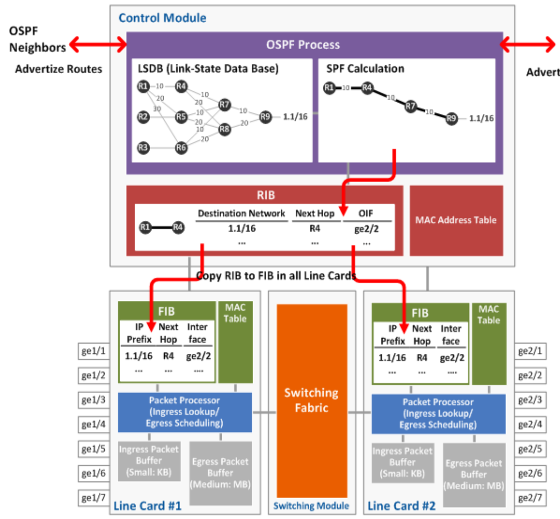

Fig. 9. IP Router Architecture.

Fig. 9. shows the architecture of IP routers. We investigate IP router architecture since it can be a base for CCN routers. For data planes, it has FIB, also known as forwarding table, packet processor and ingress/egress buffers. For control plane, to use the OSPF routing algorithm, it has Link-State Data Base (LSDB), a topology map to compute shortest path to a destination, Routing Information Base (RIB), routing table stored in a router that lists the routes to destinations and MAC address table for layer 2 connections. This is a typical architecture of an IP router.

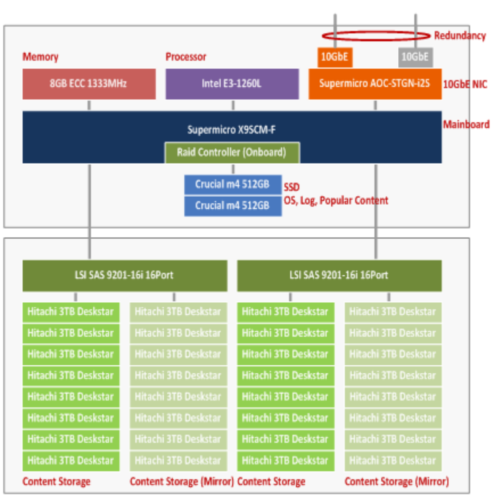

Fig. 10. CDN server architecture (Netflix Cache Server)

To propose an architecture for CCN routers, we benchmark cache servers from Netflix [21][22][23]. Netflix, Inc. is an American provider of on-demand Internet streaming media available to America and European countries [24]. Fig. 10. shows the high level diagram of its internal architecture. It contains Hard Disk Drive (HDD), Raid controller, processor, memory and LAN card.

For CCN routers, a referenceable architecture has never been proposed. In this section, we propose a feasible architecture of CCN router based on the required functionalities of CCN suggested to date [1][6][18][26][36][39]. The architecture should be similar to that of IP routers except that the CCN routers have forwarding strategy layer, PIT and CS. The functionality of PIT and CSs are already explained in the earlier section. The forwarding strategy layer makes the dynamic choices needed to best exploit multiple connectivities under changing conditions. It is used for path optimization and keeping track of dynamic network conditions [1][6]. One thing to remember is that the CS should be able to handle a great amount of contents. But fortunately, it doesn't need to be fast since it is enough to send data as fast as the CDN server that uses HDD. So we can use relatively cheap storage for CS. But index search for the CS needs to be fast enough to support real-time processing which means CS index should be stored on memory not disks. Based on the discussions for strategy plane, PIT and CS, we propose the architecture of a CCN router in Fig. 11.

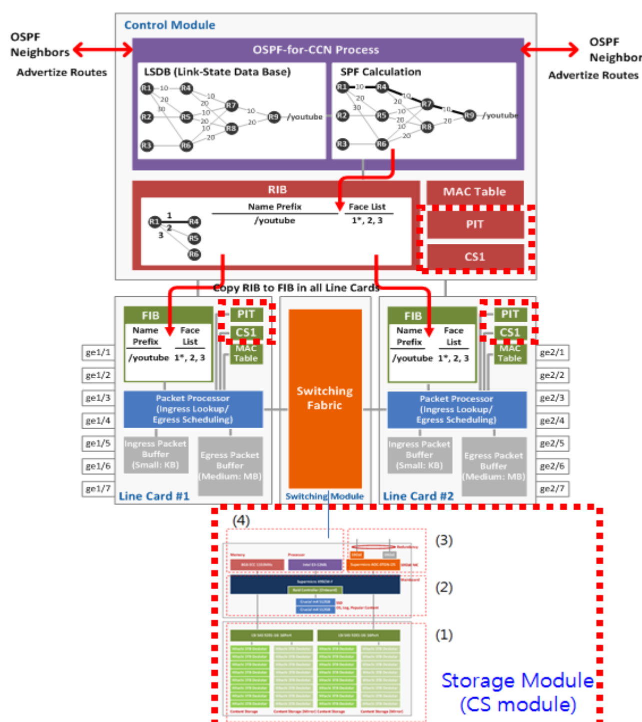

Fig. 11. CCN router architecture. Additional parts for CCN are marked with the red dotted boxes.

Fig. 11. shows how we can place forwarding strategy layer, CS module and PIT in the CCN router. To compare the H/W cost, we need to compute the cost for the three parts. For the CS module design, we can refer to the Netflix server architecture in Fig. 10. We have made some changes needed to the architecture. The storage (1) and storage control part (2) can be same. But the LAN card (3) is not needed since the CS part can be embedded as a router module. So we replace it with a connector to the backplane and an interface chip. For the processor (4), we do not need a high-end processor since there is such overhead as the CDN server (e.g., no TCP overhead). So we downsize it to Intel Zeon 5060 (3.2GHz, Dual core). For memory (4), since the size of Linux Operating System (OS) is less than 400MB, we downsize it to 8GB from 32GB.

For strategy layer and PIT, we consider that relatively small amount of memory and computing power is needed since compared with contents, entries and policies handled by PIT and strategy layer should be much smaller.

For control plan, OSPF in IP routers can be extended to OSPF-N [26] for CCN, so we can assume both architectures have almost same control plane functionality. TABLE III. shows the results of total cost comparison for CDN edge server and CCN CS module.

TABLE III. CS MODULE COST (UNITS: KOREAN THOUSAND WON)

	CDN edge server		CCN (CS module)	
CPU	350	Intel E3-1260L	105	Intel Zeon 5060(3.2GHz, Dual Core)
Memory	320	4 x 8GB ECC 1333MHz	80	8GB ECC 1333MHz
Main board	250	Supermicro X9SCM-F	250	Supermicro X9SCM-F
LAN Card	550	Super micro AOC-STGN-i2S	100	Broadcom (Connector, I/F Chip)
HDD	3,600	Hitachi Deskstar 7K3000 1TB * 40	3,600	Hitachi Deskstar 7K3000 1TB * 40
Raid	1,000	LSI SAS 9201-16i 16 port	1,000	LSI SAS 9201-16i 16 port
SSD	600	Crucial m4 512GB	600	Crucial m4 512GB
Chassis	200	TST custom	0	-
Power	1,000	Zippy MRW-5600V4V/DMRW-5600V4V	0	-
Total	**7,870**		**5,735**	

Of course, we don't believe that the costs of CCN routers can simply be estimated based on CPU, memory, main board and other units discussed above. However, we believe that the numbers suggested here should be reasonable guess at this point.

CS index should be processed very fast so the index should reside in memory. CS index cost can be quite expensive as the CS size increases.

CS index consists of contents name and index for the location of the contents. Contents name is a human readable plain text [1] so it should be hashed to save memory space. In [18], three hashing schemes for CS index, HC-basic, HC-log and HC-log+LRU have been suggested. The bits/packet for those schemes is 40, 72 and 136, respectively. We will choose 136 bits/packet. We assume the corresponding packet payload size is 1.5KB. Then we can compute the total CS index size when the total contents size stored at CS is given.

Memory type should be determined since to support high enough index searching speed, relatively expensive memory should be used, e.g., SRAM. Fig. 12. shows that the packets coming in less than 3 million per second can be processed with the indices stored in DRAM [18]. In our reference network applications, we can use DRAM for CS index since when the Zipf's law exponent s of user request is equal to 0.5, the highest packet rate is around 370 K packets/sec at the level 1 router. Routers at level $i > 1$ have slower incoming rate. And it is known that the Zipf's law exponent for user traffic is between 0.5 and 1 in general [16][17]. The packet rate increases as the exponent becomes smaller, we can consider that 370 K packets/sec would be the maximum rate in the access network.

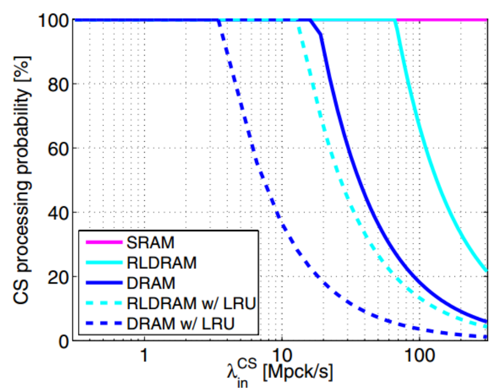

Fig. 12. Memory processing speed [18]

D. S/W cost

To provide contents services such as IPTV services, specific software such as streaming are needed. We have checked the CDN solution prices of CISCO [34] as shown in TABLE IV. Per stream cost is given. Service routers and content acquirers reside only in the center server thus their costs can be ignored.

TABLE IV. S/W PRICE FOR CISCO CDN SOLUTION

Product	Product Description	Discount Price*
CDN Edge Server (CDS-TV)	CDS-TV (RTSP Streaming, RTSP Session Management, Content Placement, Server Heath-Report, etc.)	$30.4 per Stream
Content Acquirer (CDS-CA)	CDS-CA (Reverse Proxy, Content Placement, etc.)	-

E. Total cost

In this section, we compute the total cost to provide CDN and CCN services to the entire subscribers. We converted all the network, H/W and S/W cost estimation results to the total sum for the nationwide 3 million subscribers.

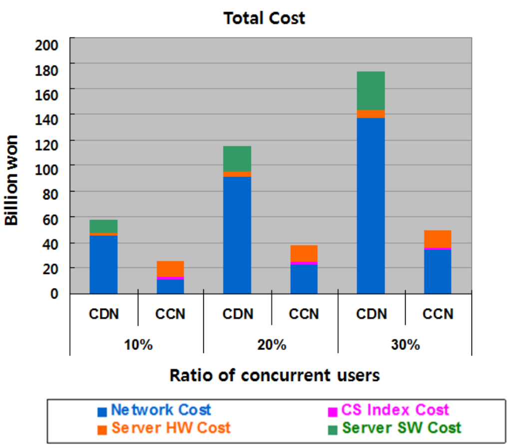

Fig. 13. Total cost comparison corresponding to the ratio of the concurrent users (s = 1, caching rate = 4:3:2:1)

Fig. 13. shows the total costs for both CDN and CCN cases. As the ratio of the concurrent user increases, the cost for the CDN increases linearly. On the contrary, with CCN, the increase rates are only 45% and 91%. When the ratio of the concurrent user is 30%, the cost to construct CDN environment is over 3.5 times bigger than that of CCN.

Compared with CDN, CCN requires much less network bandwidth thus we can save lots of costs. But H/W cost for CCN increases due to the storage cost. The S/W cost for the CDN increases linearly as the number of concurrent users increases.

VI. Performance: Protocol Overhead

In this section, we compute protocol overhead when a CDN server and CCN router process a packet. The protocol overhead is related to the performance of a network since whenever a packet passes by network equipment it should be processed by them.

Fig. 14. shows the CPU processing overhead comparison results. Firstly, for the CDN case, we listed all the instructions needed to handle CDN packets. Then we calculated the number of instructions need as shown in [19]. To compute the CCN processing overhead, we did similar computations for each step as in Fig. 14. For example, TCP process such as connection setup, it requires 1,372 instructions to run. But with CCN, since CCN does not need any connection setup process, we don't need any instruction to run. For network and Ethernet layer packet processing and Network Interface Card (NIC) driver call, we assume that the same number of instructions would be needed. Steps for request parsing and locating the contents would incur same overhead for both CCN and CDN. But the number of instructions for generating HTTP response header will be more than generating CCN response data packet header since CCN data packet format is very simple (it contains contents name and data payload).

CDN (HTTP/TCP/IP/Ethernet)		**CCN** (CCN/Ethernet)	
Processing Steps	**# of Instr.s**	**Processing Steps**	**# of Instr.s**
[Kernel TCP/IP] TCP Connection Setup		No need	
[Kernel TCP/IP] TCB (TCP Control Block) Read		No need	
[Kernel TCP/IP] Read Application Data from Application Buffer		No need	
[Kernel TCP/IP] Write Socket Buffer		No need	
[Kernel TCP/IP] TCP/IP Checksum Calculation	1372	No need	0
[Kernel TCP/IP] Write TCP Header		No need	
[Kernel TCP/IP] Update TCB		No need	
[Kernel TCP/IP] Congestion Control (Flow Control)		No need	
[Kernel TCP/IP] TCP Connection Termination		No need	
[Kernel TCP/IP] Write IP Header	380	[Kernel CCN] Write CCN header	380
[Kernel TCP/IP] Write Ethernet Header	411	same	411
[Kernel TCP/IP] NIC Driver Call	530	same	530
[Kernel TCP/IP] System Call (Receive)	499	No need	0
[Web Server] System Call (Send)		No need	
[Web Server] Parse the Request → Cache Hit		[Kernel CCN] Parse the Request → Cache Hit	
[Web Server] Locate the Requested File	574	[Kernel CCN] Locate the Requested Chunk	<574
[Web Server] Generate HTTP Response Header		[Kernel CCN] Generate CCN Response Header	
No need	0	PIT handling	<574
# of Instruction per packet	**3,776**	**# of Instruction per packet**	**<2,469**

Fig. 14. CPU processing overhead comparison.

For PIT handling, we assumed that the steps of parsing the request and locating the chunk would be the same to the steps of parsing the content name of the data reply packet and locating it in the PIT.

The results show the CCN routers require less instructions than the CDN servers by around 35%, which means CCN routers are faster than the CDN servers in processing packets in its performance.

VII. Future Forms of CDN

This section explores the hypothesis that the CDN networks evolve in response to the technology evolution of competitive networks such as CCN. This is important since CCN and CDN architectures are continually evolving so it should be fair to compare CCN with the future form of CDN configurations.

A. CDN+ & CDN++

As shown in Fig. 13. the network cost takes the biggest share in the total cost for CDN services. To reduce the network cost, CDNs can provide their services at much nearer positions to the subscribers as in Fig. 15. That is to say, CDN servers are placed at the all places of CCN routers. We call this configuration, *CDN+*.

In addition, we can embed the CDN server to the line-card of IP routers [48]. With this type, we can additionally save the costs of chassis, power and LAN card. With this configuration the network cost of CDN should be equal to that of CCN. We call this type of CDN, *CDN++*.

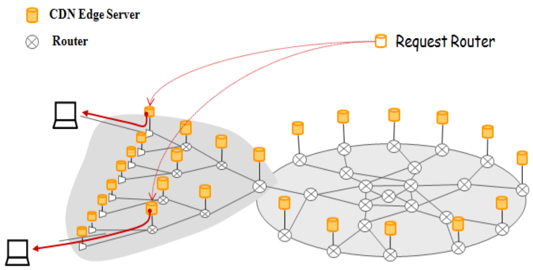

Fig. 15. CDN+

	CDN+	CDN++	CCN
Cisco CD	$30.4 / stream	$30.4 / stream	1 none
CPU	350,000	350,000	2 105,000
Memory	320,000	320,000	80,000
Main board	250,000	250,000	250,000
LAN Card	550,000	3 100,000 (Connector, I/F Chip)	100,000 (Connector, I/F Chip)
HDD	3,600,000	3,600,000	3,600,000
Raid	1,000,000	1,000,000	1,000,000
SSD	600,000	600,000	600,000
Chassis	200,000	3 0	0
Power	1,000,000	0	0
Total cost	7,870,000	6,220,000	5,735,000

1. SW for CDN service removal
2. Downsizing from TCP overhead removal
3. Line-card embedding

Fig. 16. Cost comparisons for CDN+, CDN++ and CCN.

We compared costs for CDN+, CDN++ services and CCN. For CDN+, we use the same CDN server since the only change from CDN is the CDN server placement. For CDN++, the server cost departs from CDN+. Fig. 16. shows the H/W cost comparison results.

Fig. 17. shows the total costs for CDN, CDN+, CDN++ and CCN for their nationwide service. The network cost for CCN and CDN++ are same since the placement of CDN++ servers and CCN routers is same. But the network cost for the CDN+ is a little more than that of CDN++ due to the IP port price between the IP router and the CDN server. Note that CDN++ server is embedded in IP routers as a linecard but CDN+ server is connected to IP routers.

We can see that the costs for CDN+ and CDN++ are still higher than CCN by 37% and 61%. In addition, as we discussed in the previous section, the processing speed would be much faster with CCN since it requires less number of instructions that CDN.

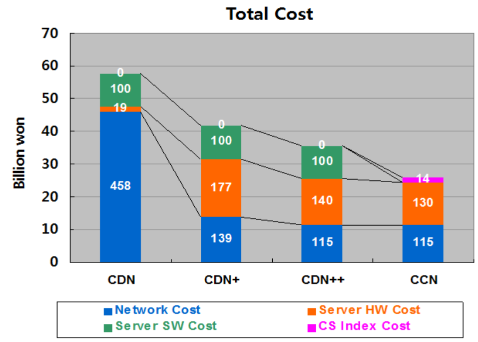

Fig. 17. Total cost for entire subscribers comparison among CDN, CDN+, CDN++ and CCN.

B. Others

Application-Layer Traffic Optimization (ALTO) service [14] that will provide applications with information to perform better by obtaining network information dynamically or measuring link performance with respect to particular peers (or servers). ALTO was developed to help reduce the redandancy caused by peer-to-peer (P2P) traffic but it can be used to improve the performance of CDN. It can be used to help improve the traffic engineering functionality of CDN.

CDNi is proposed to allow the interconnection of separately administered CDNs. With CDNi, end-to-end delivery of content through multiple CDNs will be possible [15].

ALTO and CDNi are good candidates to provide traffic engineering and inter-connection functionality to application-specific CDN networks. In addition, MPLS is one the best candidate to help provide Quality of Service (QoS) to IP-based CDN.

VIII. Conclusion

In this paper, we have compared CCN with CDN in both quantitative and qualitative manner. We proposed the first feasible CCN router implementation and typical CDN deployment topologies. We show CCN is better than CDN from many viewpoints. Qualitatively, CCN provides many advantages such as in maintenance, flow control, traffic engineering, and supporting security and mobility. Quantitatively, we estimated the total costs for network, H/W and S/W for a virtual nationwide service provider in Korea. Overall cost can be reduced by 3.5 times at maximum assuming 900 thousand concurrent users. For performance, CCN can outperform CDN by 35% due to its simpler architecture.

We couldn't deal with every aspect of CCN and CDN since many parts of CCN are still under discussion and the many details of proprietary CDNs have not been known yet. However, we believe that the discussion covered here should be a good start.

REFERENCES

[1] Van Jacobson et al. , Networking Named Content, ACM CoNEXT, 2009.

[2] http://en.wikipedia.org/wiki/Application_protocol.

[3] Ao-Jan Su, Choffnes, D.R., Kuzmanovic, A. and Bustamante, F.E., Drafting Behind Akamai: Inferring Network Conditions Based on CDN Redirections, IEEE/ACM Trans. on Net., 2009.

[4] http://www.cisco.com/web/about/ac79/docs/sp/CDN-PoV_IBSG.pdf.

[5] http://www.akamai.com.

[6] E. Paik, P. Mahadevan, M. Jang and E. Cho, Benefits and Research Challenges of Content-Centric Networking, ICN Research Group, Internet-draft, 2012.

[7] http://www3.alcatel-lucent.com/enrich/v2i32008/article_c4a3.html.

[8] NetManias Technical Reports, 2013 Content Networking Trends - OTT, Global CDN and Operator CDN, http://www.netmanias.com/en/?m=view&id=reports&no=6015.

[9] http://www.peerapp.com/products/transparentcaching.aspx.

[10] http://en.wikipedia.org/wiki/Mobile_CDN.

[11] Neil Spring, Ratul Mahajan and David Wetherall, Measuring ISP topologies with rocketfuel, IEEE/ACM Transactions on Networking, 2004.

[12] R. Govindan, and H. Tangmunarunkit, Heuristics for Internet map discovery, IEEE INFOCOM, 2000.

[13] http://www.telegeography.com/products/commsupdate/articles/2011/11/07/kts-third-quarter-net-profit-falls-amid-lower-interconnection-revenues-new-discount-plans/.

[14] http://datatracker.ietf.org/wg/alto/charter/.

[15] http://datatracker.ietf.org/wg/cdni/charter/.

[16] Lee Breslau, et al., Web Caching and Zipf-like Distributions: Evidence and Implications, IEEE INFOCOM, 1999.

[17] Lada A. Adamic and Bernardo A. Huberman, Zipf's law and the Internet, Glottometrics 3, 2002.

[18] Diego Perino and Matteo Varvello, A Reality Check for Content Centric Networking, Information-Centric Networking , 2011.

[19] Hyong-youb Kim and Scott Rixner, Performance Characterization of the FreeBSD Network Stack, Rice University, Technical Report, TR05-450, June 2005.

[20] D. Awduche, et al., Overview and Principles of Internet Traffic Engineering, IETF RFC 3272, 2002.

[21] http://blog.netflix.com/2012/06/announcing-netflix-open-connect-network.html.

[22] Netflix Open Connect CDN Deployment Guide (https://netflix.hs.llnwd.net/e1/us/layout/signup/deviceinfo/OpenConnectDeploymentGuide-v2.4a.pdf).

[23] http://www.uknof.org.uk/uknof21/Temkin-Netflix.pdf.

[24] https://en.wikipedia.org/wiki/Netflix.

[25] J. Moy, OSPF version 2, IETF RFC 2328, 1998.

[26] Lan Wang, et al., OSPFN: An OSPF Based Routing Protocol for Named Data Networking, 2012. http://www.named-data.net/techreport/TR003-OSPFN.pdf.

[27] http://www.cisco.com/en/US/products/hw/routers/ps368/prod_technical_reference09186a0080092246.html.

[28] http://enfopoint.com/products/routing.

[29] http://www.juniper.net/us/en/products-services/routing/mx-series/mx960/.

[30] http://www.alibaba.com/showroom/huawei-olt.html.

[31] http://dasannetworks.eu/en/products/gepon-olt---5724g/27/

[32] http://www.lightreading.com/cir-oadm-to-grow-52-annually/240037522.

[33] http://www.gobizkorea.com/blog/ProductView.do?blogId=mysnh&id=936646.

[34] http://www.globalpricelists.com/globalpricelistcisco.php.

[35] George K. Zipf, Human Behavior and the Principle of Least Effort, Addison-Wesley, 1949.

[36] Diana Smetters and Van Jacobson, Securing Network Content, PARC Technical Paper, 2009.

[37] RFC4301, Security Architecture for the Internet Protocol, Network Working Group of the IETF. December 2005.

[38] Nan Zhang, Internet Content Delivery as a Two-Sided Market, Aalto University, Master's thesis, 2010.

[39] KyoungSoo Park, Smart Content Delivery Technology: CCN or CDN?, TTA presentation, 2012.

[40] D. Thaler and C. Hopps, Multipath Issues in Unicast and Multicast Next-Hop Selection, IETF, RFC 2991, 2000.

[41] Rosen, E., Viswanathan, A. and R. Callon, Multiprotocol Label Switching Architecture, IETF, RFC 3031, 2001.

[42] Ziegelmann, Mark, Constrained Shortest Path and Related Problems. Constrained Network Optimization, 2007.

[43] D. Awduche, et al., RSVP-TE: Extensions to RSVP for LSP Tunnels, IETF, RFC 3209, 2001.

[44] D. Katz, K. Kompella and D. Yeung, Traffic Engineering (TE) Extensions to OSPF Version 2, IETF, RFC 3630, 2003.

[45] Lucian Popa, et al, HTTP: An Evolvable Narrow Waist for the Future Internet, TR. No. UCB/EECS-2012-5, 2012.

[46] Patrick K. Agyapong and Marvin Sirbu, Economic Incentives in Information-Centric Networking: Implications for Protocol Design and Public Policy, IEEE Comm. Magazine, 2012.

[47] KT, IPTV technical seminar, May 18, 2010 (Korean).

[48] http://www.cisco.com/en/US/docs/routers/asr9000/software/asr9k_r4.2/cda/is/release_notes/reln_a9k_ISM_251iosxr421.html#wp482412.