



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)



제주대학교 중앙도서관  
JEJU NATIONAL UNIVERSITY LIBRARY

博士學位論文

사용자가 밀집된 그룹 통신을 위한  
효율적인 CCN 멀티캐스트 라우팅 및  
캐싱 기법에 관한 연구



濟州大學校 大學院

通信工學科

李 恩 辰

2015年 2月

# 사용자가 밀집된 그룹 통신을 위한 효율적인 CCN 멀티캐스트 라우팅 및 캐싱 기법에 관한 연구

指導教授 金 興 洙

李 恩 辰

이 論文을 工學 博士學位 論文으로 提出함

2014年 12月

제주대학교 중앙도서관  
李恩辰의 工學 博士學位 論文을 認准함

審査委員長 林 載 允

委 員 左 政 祐

委 員 尹 鍾 浩

委 員 金 度 縣

委 員 金 興 洙



濟州大學校 大學院

2014年 12月

# A Study on the Efficient CCN Multicast Routing and Caching Scheme for Dense User Group Communication

Eun-Jin Lee

(Supervised by professor Heung-Soo Kim)

A thesis submitted in partial fulfillment of the requirement for the degree of Doctor of Engineering

2014. 12. 15

This thesis has been examined and approved.

.....  
Thesis director, Jea-Yun Lim, Prof. of Telecommunication Eng.

.....  
Thesis director, Jeong-Woo Jwa, Prof. of Telecommunication Eng.

.....  
Thesis director, Yoon-Chong Ho, Prof. of Electronics and Information Eng.

.....  
Thesis director, Do-Hyeon Kim, Prof. of Computer Eng.

.....  
Thesis director, Heung-Soo Kim, Prof. of Telecommunication Eng.

.....  
(Name and signature)

.....  
2014. 12. 15

.....  
Date

Department of Telecommunication Engineering  
GRADUATE SCHOOL  
JEJU NATIONAL UNIVERSITY

## 목 차

요약 .....	1
I. 서론 .....	3
II. 관련연구 .....	6
1. 콘텐츠 중심 네트워킹(CCN,Content-Centric Networking) 프로토콜 .....	6
1.1 CCN 개요 .....	6
1.2 CCN에서 라우팅 알고리즘 .....	18
1.3 CCN에서 그룹 통신 방식 .....	25
2. IP 기반의 멀티캐스트 라우팅 프로토콜 .....	28
2.1 멀티캐스트 프로토콜 .....	28
2.2 코어 기반 멀티캐스트 라우팅 프로토콜 .....	29
III. 새로운 CMR-OSPFN(Core-based Multicast Routing algorithm using OSPFN) 알고리즘을 사용한 효율적인 데이터 전달 방식 .....	34
1. 멀티캐스트 전송 방식 .....	36
2. CMR-OSPFN .....	39
2.1 OSPFN 라우팅 알고리즘 .....	41
2.2 멀티캐스트 이름 접두사 .....	44
2.3 멀티캐스트 라우팅 경로 설정 .....	46
2.4 코어 라우터의 우선 순위 .....	53
3. CMR-OSPFN 알고리즘에 의한 데이터 전달 방식 .....	56
3.1 CMR-OSPFN 알고리즘 사용을 위한 메시지 패킷 구조 .....	56
3.2 코어 라우터와 라우터 간 동작 과정 .....	56

3.3 라우터 간 동작 과정 .....	59
4. 시뮬레이션 결과 및 고찰 .....	60
4.1 시뮬레이션 네트워크 토폴로지 .....	60
4.2 콘텐츠 제공자와 코어 라우터 사이의 홉 수에 따른 콘텐츠 전달 .....	65
4.3 콘텐츠 요청 수에 따른 콘텐츠 전달 시간 .....	67
4.4 콘텐츠 수에 따른 콘텐츠 전달 시간 .....	69
IV. 밀집된 사용자를 위한 CMR-OSPFN 알고리즘의 라이프타임을 사용한 캐싱 기 법과 데이터 전달 방식 .....	71
1. 밀집된 사용자를 위한 콘텐츠 공유 서비스 .....	71
2. 밀집된 사용자 환경에서 라이프타임을 사용한 캐싱 기법 .....	75
3. 콘텐츠 서비스 시간 단축을 위한 OROC 전달 방식 .....	78
4. 시뮬레이션 결과 및 고찰 .....	81
4.1 시뮬레이션 네트워크 토폴로지 .....	81
4.2 라이프타임을 사용한 캐시의 성능 분석 .....	82
4.3 OROC 전달 방식의 성능 분석 .....	84
V. CMR-OSPFN 검증을 위한 스포츠 모바일 서비스 구현 .....	87
1. 스포츠 모바일 서비스의 사용자 동기 방식 .....	87
2. 스포츠 모바일 서비스의 데이터베이스 구조 .....	88
3. 스포츠 모바일 서비스 서버 .....	90
4. 스포츠 모바일 서비스 어플리케이션 .....	94
5. 스포츠 모바일 서비스를 위한 멀티캐스트 라우팅 프로토콜 구현 .....	100
6. 고찰 .....	102
VI. 결론 .....	104
참고문헌 .....	106

## 표 목 차

Table 1. Comparative analysis of routing algorithms in CCN .....	24
Table 2. Tree scale comparison in CBT and DVMRP .....	30
Table 3. Name prefixes originated by each router .....	48
Table 4. Router configuration files .....	49
Table 5. RTR1's Name prefix table .....	51
Table 6. RTR1's routing table .....	52
Table 7. RTR1's FIB table .....	52
Table 8. Multipath priority for the /sport.com/jejuunite/musics .....	53
Table 9. Modified FIB entry at RTR3 .....	55
Table 10. Modified PIT Entry .....	57
Table 11. Simulation Parameters .....	61



## 그림 목 차

Fig. 1. Bottleneck due to the content concentration on a server .....	7
Fig. 2. Content transfer without bottleneck in the CCN .....	8
Fig. 3. IP and CCN protocol stack .....	9
Fig. 4. Content delivery in IP network and CCN .....	11
Fig. 5. CCN forwarding engine model .....	12
Fig. 6. The basic operation of a CCN router .....	15
Fig. 7. CCN message packet types .....	16
Fig. 8. Data name structure .....	17
Fig. 9. SoCCeR operation .....	18
Fig. 10. OSPFN operation .....	19
Fig. 11. Partitioning tree of the $\alpha$ Route .....	21
Fig. 12. RRG operation in IP network .....	25
Fig. 13. RRG operation in CCN .....	26
Fig. 14. Overview of ACT Design .....	27
Fig. 15. Conference Discovery .....	28
Fig. 16. Packet delivery in CBT .....	32
Fig. 17. Multicast route discovery in CBT .....	33
Fig. 18. An example of data forwarding in CCN .....	37
Fig. 19. An example of data forwarding in CBT .....	38
Fig. 20. An example of data forwarding in CMR-OSPFN .....	40
Fig. 21. Relationship between CCND, OSPFN, and OSPFD .....	42
Fig. 22. OLSA message format .....	43
Fig. 23. Name prefix structure .....	44

Fig. 24. Name tree traversal .....	45
Fig. 25. Multicast name prefix structure .....	45
Fig. 26. Network topology for route discovery .....	47
Fig. 27. OLSA for RTR3 .....	50
Fig. 28. Name prefix structure of the core router .....	55
Fig. 29. Extentsion of CCN Packets .....	56
Fig. 30. Flow chart of the Interest Packet to the CCN Core Router .....	57
Fig. 31. Flow chart of the Data Packet to the CCN Core Router .....	58
Fig. 32. Flow chart of the Interest Packet to the CCN Router .....	59
Fig. 33. Flow chart of the Data Packet to the CCN Router .....	60
Fig. 34. Network topology .....	61
Fig. 35. Multicast data delivery process in CBT .....	63
Fig. 36. Multicast data delivery process in CMR-OSPFN for the first request ..	64
Fig. 37. Multicast data delivery process in CMR-OSPFN for the repeat request ..	64
Fig. 38. Content delivery time relative to the number of hops .....	66
Fig. 39. Network bandwidth usage relative to the number of hops .....	67
Fig. 40. Content delivery time relative to the number of content request ...	68
Fig. 41. Content delivery time relative to the number of content .....	69
Fig. 42. Share contents in the classroom .....	72
Fig. 43. Share contents in the stadium .....	73
Fig. 44. Content delivery process .....	74
Fig. 45. Caching and data delivery in sport stadium .....	76
Fig. 46. Caching and Data delivery on the server .....	77
Fig. 47. Caching and Data delivery on the core router .....	78
Fig. 48. The comparison of the two transmission method .....	79
Fig. 49. Network topology of the user dense CCN environment .....	81

Fig. 50. Content service time relative to the number of content .....	83
Fig. 51. Content service time relative to the number of hops .....	84
Fig. 52. The size of Interest Packet relative to the content size .....	85
Fig. 53. Content service time relative to the content size .....	85
Fig. 54. The hierarchical structure of devices in the platform .....	88
Fig. 55. Relational database structure of the sports mobile services platform .....	89
Fig. 56. Relational database of the sports mobile services platform .....	90
Fig. 57. Login UML sequence diagrams .....	91
Fig. 58. Point UML sequence diagrams .....	91
Fig. 59. Board UML sequence diagrams .....	92
Fig. 60. Cheer story UML sequence diagrams .....	93
Fig. 61. Chatting UML sequence diagrams .....	93
Fig. 62. Initial setup and main screen .....	94
Fig. 63. Login Screen .....	95
Fig. 64. UML diagrams for cheer group formation .....	95
Fig. 65. Cheer group leader screen .....	96
Fig. 66. Cheer group user screen .....	97
Fig. 67. Board screen .....	97
Fig. 68. Cheer story screen .....	98
Fig. 69. Chatting screen .....	99
Fig. 70. UML sequence diagram of between server and App .....	100
Fig. 71. UML sequence diagram of between server and CCN core router ..	101
Fig. 72. UML sequence diagram for group communication .....	102

## Abstract

CCN is a content-centric networking architecture that routes and delivers contents using their names instead of IP addresses at the packet level of the network. In CCN, the name of the desired content in the request message called interest is used to calculate route and to find a match in the distributed cache. OSPFN has been proposed to distribute name prefixes and calculate routes to name prefixes.

In this thesis, we propose a new core-based multicast routing algorithm using OSPFN (CMR-OSPFN) for group communication in CCN. In the proposed CMR-OSPFN, a core-router and node flood content name prefixes and group name prefixes using the modified OLSA message in OSPFN, respectively. The group name and the core-router information are stored in the modified FIB table to provide the core-based multicast routing. We also propose a new data delivery scheme that uses the TOS field inserted in the message packet. The core router delivers data packet to all the members within the group for an interest message from a user. Therefore, the proposed CMR-OSPFN can reduce the content service time to  $T_I + N_u T_D$  relative to  $N_u T_I + N_u T_D$  in the dense user environment where  $N_u$  is the number of users and  $T_I$  and  $T_D$  are the transmission times of interest and data packets, respectively. We compare the performance of the proposed CMR-OSPFN with that of CBT using NS-2 simulator. We measure the content delivery time and the network bandwidth usage relative to the number of hops between the core router and the service server. Simulation results show that the average content delivery time and the average network bandwidth usage of the proposed CMR-OSPFN are reduced to 45% and 74% relative to those of CBT at 10 hops, respectively.

In CCN, the LRU (least recently used) and LFU (least frequently used) schemes are used in the distributed caches. We propose an effective cache scheme for a core-router with volatile content. In the proposed cache scheme, the content provider issues a volatile content to the core router and then the core-router serves for the duration of the lifetime in that content. To confirm the performance of the proposed cache scheme, we measure the content service time from the service server to the core router. The average content service time of the proposed core router based caching scheme can reduce by  $N_{SR}T_I$  where  $N_{SR}$  is the number of hops from the service server to the core router. In CCN,  $N_D$  interest messages should be used to receive a content consisted of  $N_D$  data packets based on the OROP (one request one packet) scheme. We propose the OROC (one request one content) scheme in which a core router transmits  $N_D$  data packets for an interest message from node. Therefore, the content service time of the proposed OROC scheme is reduced to  $T_I + N_D T_D$  relative to  $N_D T_I + N_D T_D$  of the OROP one.

The proposed CMR-OSPFN with the new cache and OROC schemes can be used in the dense user environments such as sport stadiums, classrooms, theaters, and conference rooms. We develop the sport mobile group communication service in stadiums using the proposed CMR-OSPFN. In this service, the sport mobile service server issues game contents with the lifetime filed and the core-router with a cache delivers game contents to users for the duration of the lifetime of contents. People come to cheer in the stadium can use this service using the cheer mobile application.

## I. 서론

현재 인터넷에 연결되는 디바이스가 다양해지면서 콘텐츠의 생산과 소비의 형태도 변화하고 있다. 콘텐츠 제공자가 제작한 멀티미디어 데이터를 사용자에게 전달하던 형태에서 사용자가 직접 데이터를 생산하고 이를 다수의 사용자가 사용할 수 있도록 하기 위해 서버로 데이터를 업로드하거나 데이터를 실시간으로 공유하기 위해 데이터 공유 그룹에게 직접 데이터를 전송하는 등의 방식으로 변화하고 있다 [1][2].

실시간 그룹 통신은 콘텐츠 공유를 위한 다양한 분야에서 기본 토대가 되고 있다. 그룹 통신은 그룹 멤버 중 누군가가 데이터나 제어 패킷을 발생시키면 모든 노드에게 동시에 전송되는 것을 의미한다. 그룹 멤버들 사이의 그룹 통신을 위해서는 대용량의 콘텐츠를 빠른 속도로 처리하여 사용자들에게 전달하여야 한다. 하지만 IP 기반의 인터넷 방식은 IP 주소 부족, 일대다 통신 지원 부재, 네트워크 관리의 용이성 부재 등으로 대용량의 콘텐츠를 실시간으로 그룹에게 전송하기에 효율적이지 못하다. 이를 해결하기 위해 미래 인터넷의 주요 핵심 기술로 주목받고 있는 콘텐츠 중심 네트워킹 기술인 CCN(content-centric networking)이 제시되었다<sup>[3]</sup>.

CCN은 IP 주소가 아닌 콘텐츠 이름으로 콘텐츠를 효율적으로 가져오거나 분산시킬 수 있는 새로운 네트워킹 기술이다. CCN에서는 콘텐츠 이름을 이용해 콘텐츠를 요청하고 가져올 수 있으며, 캐싱 기능과 CCN 라우팅 알고리즘의 사용으로 네트워크 용량을 증대시키고 콘텐츠 전달 시간을 줄인다. CCN 라우팅 알고리즘은 다중 경로를 지원하며, 동일한 시간에 여러 노드가 라우터에 동일한 콘텐츠를 요청하면 라우터는 멀티캐스트 전송 방식으로 콘텐츠를 전달한다. 이와 같이 CCN 라우팅 알고리즘은 하나의 노드가 콘텐츠를 요청하면 해당 그룹의 모든 노드에게 콘텐츠를 전달해야 하는 그룹 통신을 지원하지 못한다. 그래서 CCN에서 이루어지고 있는 그룹 통신을 위한 연구들은 CCN 네트워크 계층에서 멀티캐스트 프로토콜을 제안한 것이 아닌 응용 계층을 이용한 멀티캐스트 전송 방식으로 제안되고 있다.

CCN에서 그룹 통신을 위한 멀티캐스트 전송에 대한 연구로는 라우팅 기능을 응

용 프로그램으로 구축한 CCN 기반 SNS 플랫폼 구축<sup>[4]</sup>, 응용 계층의 멀티캐스트 기법인 가십 프로토콜(gossip protocol)<sup>[5][6]</sup>을 적용한 CCN에서 P2P 실시간 그룹 통신 설계<sup>[7]</sup>, 그리고 CCN의 상위 계층의 어플리케이션을 이용한 오디오 컨퍼런스 툴인 ACT(audio conference tool)<sup>[8]</sup> 등이 이루어지고 있다. 이러한 연구들은 응용 계층을 이용한 멀티캐스트 전송에 대한 연구들로 새로운 CCN 네트워크 계층의 멀티캐스트 라우팅 알고리즘이 필요하다. 또한 사용자가 밀집된 환경에서 같은 공간의 구성원들이 함께 콘텐츠를 공유하는 그룹 통신의 특징은 동일한 시간에 서버로 트래픽이 집중된다는 것과 동일한 콘텐츠를 반복적으로 서버에게 요청한다는 것이다. 이와 같이 서버로 동일한 시간에 트래픽이 집중되는 현상은 서버 주변에 심각한 인터넷 병목현상을 일으키며 콘텐츠 공유를 위해 콘텐츠를 전송하는데 많은 지연이 발생한다. 이에 사용자가 밀집된 네트워크 환경에서 콘텐츠를 공유하는 그룹 통신을 위한 효율적인 CCN 멀티캐스트 라우팅 알고리즘에 대한 연구가 필요하다.

본 논문에서는 CCN 네트워크 계층의 CCN 라우팅 알고리즘인 OSPFN<sup>[9]</sup>에 적합한 새로운 멀티캐스트 라우팅 알고리즘인 CMR-OSPFN(Core-based Multicast Routing algorithm using OSPFN)을 제안한다. 또한 제안하는 CMR-OSPFN 알고리즘을 사용한 새로운 데이터 전송 방식을 제안한다. CMR-OSPFN은 코어 라우터와 노드가 콘텐츠 이름 접두사와 그룹 이름 접두사를 네트워크로 플로딩하기 위해 OSPFN의 OLSA 메시지를 수정한다. 코어 기반 멀티캐스트 라우팅을 제공하기 위해 그룹 이름과 코어라우터의 정보는 FIB 테이블을 수정하여 저장한다. 또한 기존 CCN의 메시지 패킷 내에 삽입된 TOS(type of service) 필드를 사용하여 새로운 데이터 전송 방식을 제안한다.

사용자가 밀집된 네트워크 환경에서의 그룹 통신은 서버에 트래픽 집중으로 병목현상에 의해 전송 지연이 발생된다. 본 논문에서는 사용자가 밀집된 CCN 환경에서 그룹 통신을 위한 효율적인 캐싱 기법과 새로운 데이터 전달 방식을 제안한다. 기존 CCN의 캐싱 기법은 LRU(least recently used)와 LFU(least frequently used)를 사용한다<sup>[11]</sup>. 이 방식은 사용자가 밀집된 CCN 환경에서 그룹 통신을 위해서는 일정 시간 서비스를 제공하는 경우에는 적합하지 않다. 본 논문에서 제안하는 캐싱 기법은 콘텐츠 제공자로부터 라이프타임(lifetime) 필드가 추가된 콘텐츠를 전

달받아, 캐시에 저장하고 일정 시간 서비스한 후 라이프타임이 종료되면 삭제되는 휘발성 콘텐츠(volatile content)를 이용하는 방식이다. 제안하는 휘발성 콘텐츠를 이용한 캐싱 기법은 사용자가 밀집된 CCN 환경에서 콘텐츠를 더욱더 빠르게 사용자에게 제공할 수 있을 것이다.

기존 CCN에서는 N개의 데이터 패킷으로 구성된 콘텐츠를 전달하기 위해 N개의 요청 패킷이 필요한 OROP(One-Request-One-Packet)전달 방식을 사용한다. 본 논문에서는 사용자가 밀집된 CCN 환경에서 CMR-OSPFN 알고리즘을 사용하여 그룹 통신을 하므로 모든 멀티캐스트 데이터는 코어 라우터로부터 전달된다. 이때 OROP 방식을 이용할 경우 신뢰성 있는 전송을 보장할 수 있지만 불필요한 요청 패킷으로 데이터 전송 지연이 발생한다. 이에 하나의 요청 패킷을 코어 라우터로 전달하여 N개의 데이터 패킷을 전달받는 OROC(One-Request-One-Content)전달 방식을 제안한다. OROC를 이용할 경우 연속적인 시퀀스의 데이터 패킷에 대해 반복적으로 요청되는 요청 패킷을 줄여 콘텐츠 서비스 시간을 줄일 수 있을 것이다.

제안하는 CMR-OSPFN 알고리즘을 사용한 모바일 서비스 플랫폼은 경기장, 강의실, 공연장, 회의장 등 사용자가 밀집된 네트워크 환경에서 사용할 수 있다. 본 논문에서는 스포츠 경기장에서 응원 그룹 서비스를 제공하기 위해 스포츠 모바일 서비스 플랫폼과 모바일 응원 어플리케이션을 개발한다.

본 논문은 다음과 같이 구성되어 있다. II장에서는 CCN 개요, CCN에서 라우팅 알고리즘, 실시간 그룹 통신에 대한 관련 연구에 대한 내용과 IP 기반의 멀티캐스트 라우팅 프로토콜에 대한 관련 연구를 살펴본다. III장에서는 제안하는 CCN 기반의 CMR-OSPFN 알고리즘을 기술하고 이를 사용한 새로운 데이터 전송 방식에 대하여 설명하고 검증하며, IV장에서는 밀집된 사용자를 위한 CMR-OSPFN 알고리즘의 효율적인 캐싱 기법과 데이터 전달 방식을 설명한다. V장에서는 개발한 스포츠 모바일 서비스 플랫폼 및 모바일 응원 어플리케이션에 대해 기술하고, 개발된 모바일 응원 어플리케이션을 이용해 제안한 CCN에서 CMR-OSPFN 알고리즘을 사용한 캐싱 기법과 데이터 전달 방식을 검토 및 고찰하며, VI장에서는 본 논문의 결론을 맺는다.

## II. 관련연구

본 장에서는 CCN에서 그룹 통신을 위한 데이터 전달 방식을 제안하기 위해 기존 IP 기반의 인터넷의 콘텐츠 전송에서 발생하는 IP 주소 부족, 일대다 통신 지원 부재, 네트워크 관리의 용이성 부재 등의 문제를 해결하기 위한 콘텐츠 전송 방식인 CCN에 대해 기술하고 CCN에서의 그룹 통신을 위한 데이터 전달 방식 관련 연구에 대해 살펴본다. 또한 CCN에서의 그룹 통신 방식인 멀티캐스트 라우팅 프로토콜을 제안하기 위해 기존 IP 기반의 멀티캐스트 라우팅 프로토콜과 CCN의 기존 라우팅 알고리즘들을 기술한다.

### 1. 콘텐츠 중심 네트워킹(CCN, content-centric networking) 프로토콜



#### 1) CCN 개요

##### (1) 호스트 중심 네트워킹(Host-Centric Networking)

비디오, 오디오, 사진, 금융 등의 정보의 디지털화가 급속히 이루어지면서 매년 유선 인터넷 트래픽은 연평균 33% 이상의 증가가 예상되며, 전 세계적으로 2015년 매월 63EB(exabyte)에 이르게 될 것으로 전망되고 있다. 또한 국내 백본 트래픽은 연평균 31% 이상 증가가 예상되며, 2015년 매월 5.9EB에 도달할 것으로 예상된다. 무선 트래픽은 세계적으로 연평균 92% 증가하였으며, 국내의 경우는 연평균 103% 증가하여 현재 추세를 기준으로 2020년에는 2010년 대비 약 189배의 증가가 예상된다<sup>[12][13]</sup>. 하지만 현재의 인터넷은 과거에 이동이 어려운 부피가 큰 장비임을 감안하여 지역적으로 분산된 사용자들에게 서비스를 제공하기 위해 호스트 중심 네트워크를 통해 양단간의 데이터 통신을 위해 물리적 장비들의 트리 구조를

가진다<sup>[14]</sup>.

이는 통신에는 효율적일지는 몰라도 분산의 관점에서는 매우 비효율적인 구조이다. 따라서 이 관점은 스마트폰과 같은 이동 단말이 급속히 보급되고, 다양한 장소에서 콘텐츠를 이용하려는 환경에는 적합하지 않다. 따라서 위치가 동일한 정보에 대한 여러 단말들의 수요가 발생하게 되면 인터넷상의 병목 현상과 같은 문제가 발생할 수 있다.

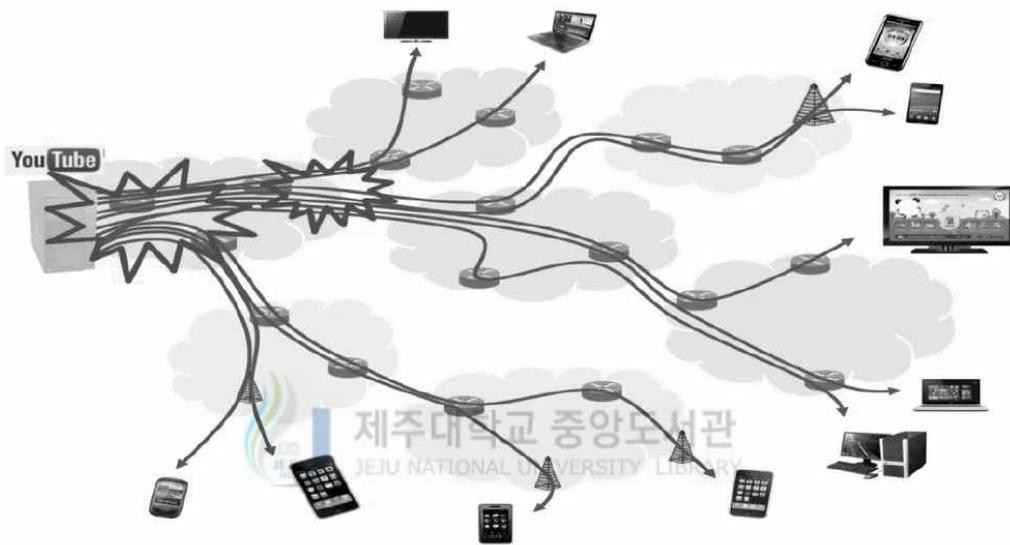


Fig. 1. Bottleneck due to the content concentration on a server

그림 1은 YouTube 서버로 다양한 단말들이 동영상 자료들을 요구하는 경우 발생할 수 있는 병목 현상을 보여준다. 이러한 문제는 사용자들의 정보사용 방식이 변화되었기 때문에 발생하게 된다. 과거의 인터넷 사용자들은 단말의 이동이 거의 불가능하기 때문에 단말의 위치가 곧 단말이 가진 정보 자체로 나타낼 수 있었다. 하지만 현재의 사용자들은 더 이상 원하는 정보의 위치가 아닌 정보 자체에 집중하는 방식으로 정보사용 방식이 변화되었다. 이러한 인터넷 사용자의 변화를 수용하기 위해 현재 전 세계적으로 다양한 연구가 진행되고 있다<sup>[14]</sup>.

## (2) 콘텐츠 중심 네트워킹

CCN은 현재 인터넷의 구조가 과거 고가의 제한된 네트워크 자원을 공유하며 소량의 데이터를 교환하는 것으로 설계되었다는 문제를 해결하기 위해 낮은 비용으로 제공 가능한 네트워크 자원을 이용하여 대량의 데이터 서비스를 효율적으로 지원하는 새로운 인터넷 구조로의 전이가 필요함을 주장하고 있다<sup>[15]</sup>. CCN은 IP 주소가 아닌 콘텐츠 이름으로 콘텐츠를 효율적으로 가져오거나 분배할 수 있는 새로운 네트워킹 프로토콜이다. 사용자는 콘텐츠를 제공하는 기기의 주소가 아닌 콘텐츠 이름으로 원하는 콘텐츠를 가져올 수 있으며, 네트워크 내 노드의 캐시를 활용하여 네트워크의 전송 용량 증설 없이 수많은 사용자를 지원할 수 있다. CCN에서 가장 큰 특징은 위치 정보를 나타내는 IP를 활용하지 않고 콘텐츠 자체에 부여된 콘텐츠 이름을 통해 기기가 콘텐츠를 획득할 수 있는 점이다. 두 번째 CCN의 특징은 콘텐츠의 집중을 막기 위해 각 노드가 경로를 통해 지나간 콘텐츠의 내용을 일정 시간 동안 보관해 그 사이 다른 단말이 동일한 콘텐츠를 요구하는 경우 콘텐츠 제공자에게로 요청 메시지를 보내는 것이 아니라 콘텐츠를 보관하고 있는 노드에서 응답을 보낸다는 것이다. 그림 2는 콘텐츠를 각 노드가 자신의 캐시에 콘텐츠를 보관함으로써 콘텐츠 제공자로 요청이 집중되는 병목 현상이 없는 콘텐츠 전송을 보여준다.

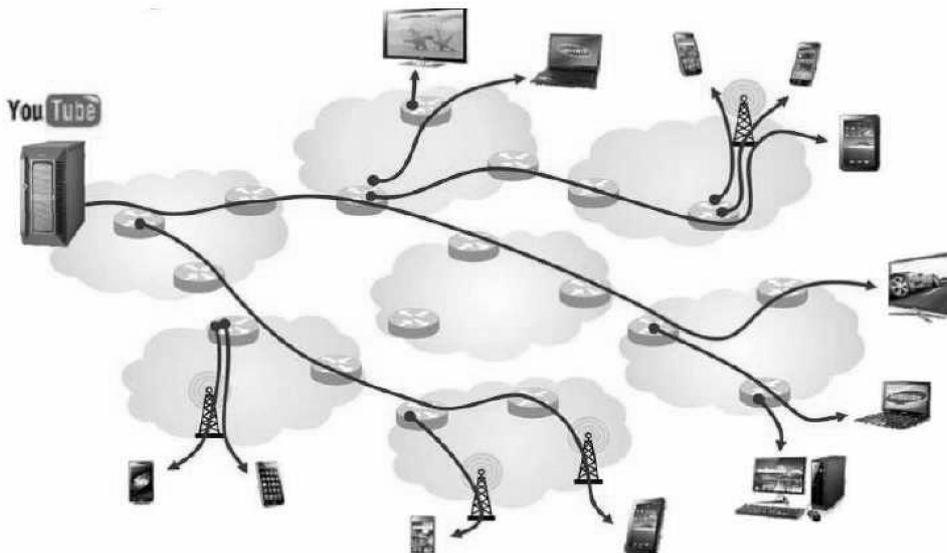
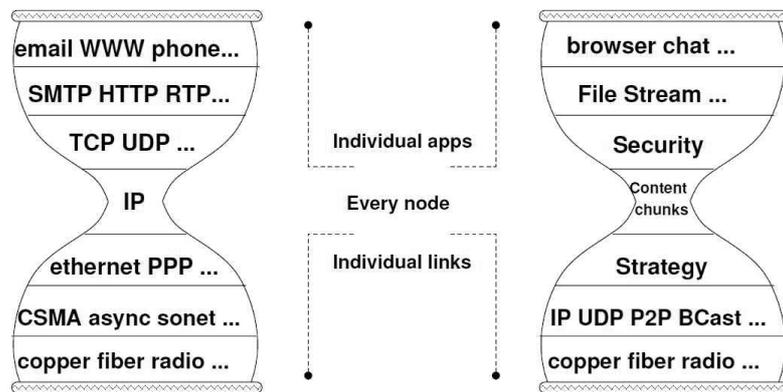


Fig. 2 Content transfer without bottleneck in the CCN

CCN 기술은 호스트 기기의 위치를 나타내는 인터넷 주소가 아닌, 콘텐츠 이름 자체를 네트워킹을 위한 주소로 사용하여 사용자가 원하는 콘텐츠를 가장 가까운 곳에서 가져올 수 있게 한다. 콘텐츠를 빠르게 가져오기 위해 네트워크 단에 콘텐츠 캐싱 기능을 제공한다. CCN은 콘텐츠 이름 자체를 네트워킹 주소로 사용하므로 콘텐츠 자체의 인증을 가능하게 한다. 이는 다양한 신뢰 모델에 기반한 네트워크 단에서 보안 기능을 제공할 수 있다. 또한 콘텐츠 자체가 네트워킹의 주체가 됨으로써 콘텐츠 별 QoS를 제공한다<sup>[14]</sup>.

그림 3은 IP와 CCN 프로토콜 구조<sup>[16]</sup>를 보여준다. 인터넷에서 IP프로토콜을 사용하는 반면, CCN 기술에서는 이름으로 식별되는 콘텐츠 청크(chunk), 정책(strategy), 보안(security) 계층을 사용한다는 차이점을 가진다. 인터넷 기술의 경우 송신 호스트가 패킷의 IP 헤더에 송수신 호스트의 IP 주소를 명시하여 데이터를 전송한다. 반면, CCN 기술은 데이터를 제작한 콘텐츠 제공자와 소비자가 IP 주소 대신 데이터 이름을 사용한다. 즉, 소비자는 데이터를 요구하는 패킷에 콘텐츠 이름을 담아 전달하고 데이터를 갖고 있는 노드가 콘텐츠 요구 패킷을 수신하면 콘텐츠 요구 한 노드에게 데이터를 전송한다. CCN 기술은 콘텐츠 생성자가 반드시 데이터를 전송한다는 제한을 두지 않고, 데이터를 갖고 있는 임의의 노드도 데이터를 배포할 수 있게 함으로써 데이터가 네트워크에서 반복되어 전송되는 횟수를 줄여 빠른 서비스가 가능하다는 장점을 가진다.



(a) IP protocol stack

(b) CCN protocol stack

Fig. 3. IP and CCN protocol stack

그림 3 (b)의 CCN 프로토콜 구조를 살펴보면, 그림 3 (a)의 TCP/IP 프로토콜 구조와 차이는 있지만 기존 TCP/IP 프로토콜 구조를 크게 바꾸지 않고도 설계할 수 있다. IP 프로토콜의 구조에서 대부분의 계층은 장치들 간의 서로 대등한 관계를 보인다. 즉, 2계층의 프레이밍 프로토콜은 물리적 링크의 양단 사이의 협의된 규칙이고, 4계층인 전송 프로토콜은 발신지와 목적지(소비자와 제공자) 사이의 협의된 규칙이다. 네트워크 계층인 3계층은 인터넷 프로토콜이라는 전송 메커니즘을 다루고 있으며 호스트의 주소 지정과 패킷 분할 및 조립 기능을 담당한다. 2계층에서는 최선 노력 전송(best-effort delivery), 비신뢰성, 비순차성, 불확실성 등을 3계층에 요구한다. CCN의 여러 가지 중요한 방법은 IP에 의해 시작된다. 프로토콜 스택에서 전략과 보안 두 개의 새로운 계층을 볼 수 있다. CCN은 2계층으로 인해 여러 개의 동시 연결을 최대한 활용할 수 있다(즉, 이더넷, 3G, 블루투스, 802.11). 전략 계층은 변화하는 네트워크에서 최상의 경로를 이용하여 다중 연결하는 방식인 동적 최적 선택(dynamic optimization choices)알고리즘이 필요 한다. CCN은 연결 경로보다 콘텐츠 자체를 보호하므로 IP 네트워킹의 문제인 호스트 기반의 많은 취약점들을 해결할 수 있다. CCN에서 제안하는 내용은 콘텐츠 이름을 기반으로 '무엇(what)'을 중심으로 콘텐츠를 찾는 것이며, 인터넷 IP 주소 기반의 '어디(when)'에 중점을 두어 콘텐츠의 종류와 상관없이 송신자와 수신자간의 1:1연결을 통해 다수의 수신자가 단일 송신자에게 동일한 데이터를 요청하는 것과는 다르다. CCN은 수신자의 요청 수만큼 반복 전송해야하는 비효율적인 IP 주소 기반을 개선한 구조이다.

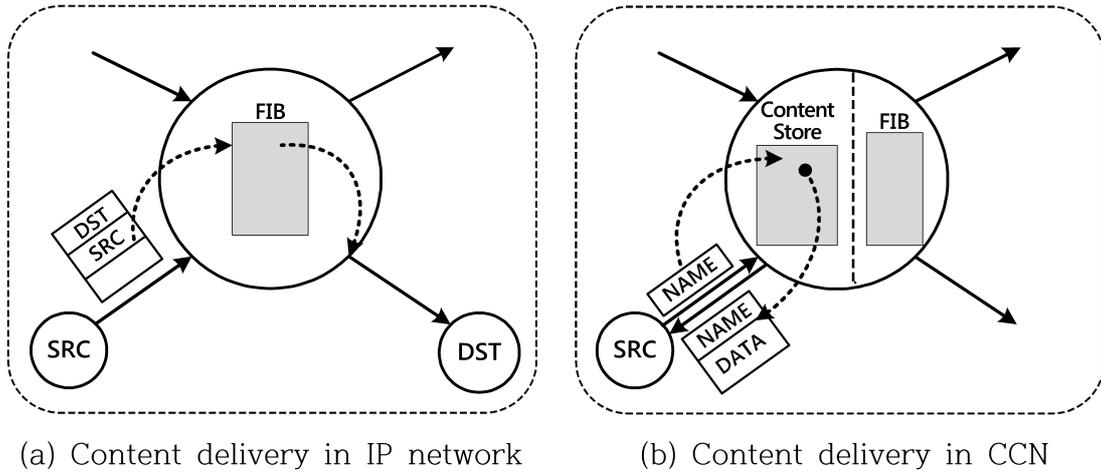


Fig. 4. Content delivery in IP network and CCN

그림 4 (a)에서 나타내듯이, 호스트 주소 기반으로 콘텐츠를 전달하는 시스템에서는 패킷을 보내는 소스 주소와 전달되는 목적지 주소가 패킷 자체에 포함된다. 이러한 주소를 가지는 패킷이 네트워크 스위치 혹은 라우터에 도착하면, 각각의 목적지 주소마다 전달 방향을 알려주는 주소록인 FIB(forwarding information base)를 이용하여 패킷 전달이 이루어진다. FIB는 네트워크 토폴로지 정보를 가지고 있는 테이블이며, 통상 라우팅 프로토콜을 통해 업데이트 된다. 이러한 동작 구조의 특징은 데이터 패킷이 소스와 목적지 주소를 지니고, 중간 노드들인 네트워크 스위치나 라우터들은 단순히 전달 방향을 바꾸어주는 역할만 하게 되는 것이다.

이와 달리 그림 4 (b)에서 표현하는 CCN에서는 데이터 요청 메시지나 데이터 패킷이 콘텐츠인지를 나타내는 이름을 가진다. 그 이름과 일치하는 데이터가 네트워크 스위치나 라우터의 로컬 캐시 메모리에 저장되어 있는 경우에는 그 데이터가 직접 요청자에게 보내지며, 그렇지 않은 경우에는 FIB를 이용하여 또 다른 콘텐츠 저장자에게 요청 메시지가 전달되게 된다. 이 경우 주소 기반 네트워킹의 경우와 달리, FIB는 콘텐츠 이름을 기반으로 전달 방향을 결정하는 일종의 콘텐츠 지도(content topology map)를 가진다.

또 하나 강조되어야 하는 점은, 호스트 기반의 메커니즘과 비교할 때 CCN에서는 네트워크 노드에 있는 저장 메모리는 효율적으로 대량의 콘텐츠를 다수의 사용자에게 분배하고 소비하는데 중요한 역할을 하게 된다는 것이다. 통신 회선과 메모

리 가격의 하락 추세를 보이며 가격의 하락 속도가 전용회선 대비 8배 이상이 되는 고용량 메모리의 사용을 기반으로 하는 새로운 네트워킹 기술이 비용 측면에서도 더욱 바람직하다<sup>[17][18]</sup>.

### (3) CCN의 구조 및 구성요소

#### ① CCN 라우팅

CCN 기본 라우팅 알고리즘은 각 노드의 포워딩 엔진을 이용해 실행되며, CCN 포워딩 엔진은 그림 5와 같이 FIB, PIT(pending interest table), CS(content store)로 구성된다.

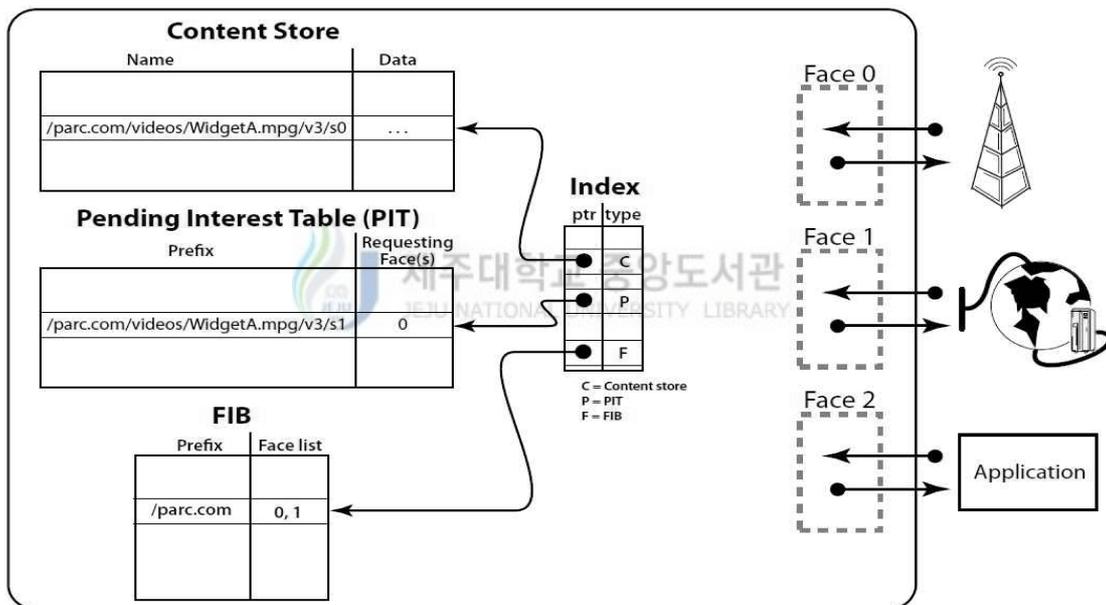


Fig. 5. CCN forwarding engine model

CCN 포워딩 엔진 안에 페이스(face)는 CCN과 IP 인터페이스의 차이를 표현하기 위해 사용된다. 즉, IP는 다중 연결을 지원하지 못하지만 CCN은 다중 연결을 지원할 수 있는 것을 의미한다. FIB는 콘텐츠를 요청하는 요청 패킷을 포워딩하는데 사용한다. CCN의 FIB는 IP의 FIB와 대부분 동일하나 IP의 경우 다중 연결 지원을 하지 못한다. 이는 CCN이 사용자와 콘텐츠 제공자 사이의 경로가 두 개 이상 존재할 때 한 개의 경로만을 남겨 두고 나머지 경로를 차단하는 IP의 스페닝

트리 알고리즘<sup>[19]</sup>을 이용하지 않았다는 것을 나타낸다. 콘텐츠에 대한 다중 소스를 허용하고 동시에 콘텐츠를 소유한 모든 소스에 대한 조회가 가능하다는 것이다.

CS는 IP 라우터의 버퍼 메모리와 같으나 CCN은 IP와 다른 메모리 교환 정책을 가진다. 각각의 IP 패킷은 하나의 단대단 통신을 하므로 다음 노드로 패킷을 전달한 후에 패킷을 저장하고 있지 않는다. IP는 패킷 전달을 완료한 후 패킷을 버퍼에서 삭제하고 즉시 버퍼를 재사용하는 MRU(most recently used) 교환 정책을 사용한다. CCN 패킷은 사용자들에게 각 패킷의 잠재적 유용성에 대한 자기식별과 자기인증을 가진다. 여기서 패킷의 잠재적 유용성이란 어떤 동일한 뉴스나 동일한 멀티미디어를 많은 사용자가 읽거나 볼 것이라는 것을 의미한다. 사용자가 패킷을 사용하는데 있어 대역폭의 수요와 대기 시간을 최소화하여 패킷의 공유 가능성을 높이기 위해 CCN은 LRU(least recently used)와 LFU(least frequently used) 교환 정책을 사용하여 요청 패킷 응답으로 도착한 데이터 패킷을 저장한다.

PIT는 콘텐츠를 요청한 사용자에게 데이터 패킷을 전송하기 위해 요청 패킷이 들어온 페이스를 기록한다. CCN은 오직 요청 패킷의 경로를 관리하여 라우팅을 수행한다. 즉, 콘텐츠 요청 노드가 요청 패킷을 전송하면 이를 수신한 노드들은 자신의 PIT에 요청 패킷이 들어온 페이스를 기록하였다가 요청된 데이터 패킷이 들어오면 자신의 PIT에 저장된 페이스를 이용해 콘텐츠 요청 노드로 전달한다. PIT에 저장된 페이스는 해당 데이터 패킷을 전달한 후 삭제된다. PIT는 일정 시간 안에 데이터 패킷이 도착하지 않으면 요청 패킷을 재전송해야한다. 재전송의 책임은 콘텐츠 요청 노드에게 있으며 재전송 후에도 일정 시간 동안 데이터가 도착하지 않으면 해당 요청 패킷에 대한 정보를 삭제한다.

CCN에서의 패킷 포워딩 과정을 살펴보면, 콘텐츠 제공자는 서비스하고자 하는 콘텐츠 이름을 자신이 연결된 네트워크에 공지함으로써 해당 정보가 네트워크로 알려진다. 네트워크의 CCN노드는 공지 받은 정보를 FIB 테이블에 추가한다. CCN노드가 요청 패킷을 수신하였을 때 CS와 PIT에서 먼저 콘텐츠 이름으로 검색해보고 없으면, PIT는 요청 패킷이 입력된 페이스를 기록하고 FIB의 정보에 따라 콘텐츠 제공자가 있는 방향으로 요청 패킷을 포워딩한다. FIB에 따라 요청 패킷을 전송 후에는 해당 요청 패킷 정보를 PIT에 일정시간 저장해 둔다. 이후 동일 콘텐츠 이름의 요청 패킷이 수신되었을 때 중복 포워딩하지 않고 해당 요청 패킷이 유입

된 인터페이스 정보만을 남겨두고, 요청된 콘텐츠를 포함한 응답으로 전송하는 데이터 패킷이 도착하면 해당 요청 패킷의 모든 인터페이스로 데이터 패킷을 포워딩한 후, 요청 패킷 정보는 PIT에서 제거한다. 이처럼 PIT는 요청 패킷을 일정시간 기록하여 요청 패킷이 지나온 역방향 경로를 따라 데이터 패킷을 포워딩할 수 있게 한다. 또한, 동일 콘텐츠 이름에 대한 요청 패킷은 외부로 한 번씩만 포워딩함으로써, 콘텐츠 제공자로 향하는 요청 패킷의 수를 줄이고, 데이터 패킷이 도착하면 요청 받았던 모든 인터페이스로 전달한다. 이는 IP 멀티캐스트와 같이 다수의 콘텐츠 요청자에게 효율적인 데이터 전송을 가능하게 한다. CS는 라우터의 버퍼 메모리와 같이 데이터 콘텐츠와 콘텐츠 리스트를 저장하고 있다. CCN 노드는 데이터 패킷을 받으면 자신의 PIT에 남겨진 정보에 따라 데이터 패킷을 전달하고 해당 데이터 패킷을 삭제하는 것이 아니라 CS에 저장한다. 이는 In-Network Caching 개념으로 이후 동일한 콘텐츠 이름에 대한 요청 패킷이 수신되었을 때 CS에서 먼저 검색하여 즉시 응답할 수 있다<sup>[20]</sup>.

CCN 포워딩 엔진을 통한 기본 라우팅 알고리즘의 동작은 그림 6과 같다. 요청 패킷이 페이스로 노드에 들어오고 요청된 콘텐츠의 이름을 CS에서 확인한다. CS에 저장된 콘텐츠 이름과 수신된 요청 패킷의 콘텐츠 이름이 일치하는 경우 데이터 패킷을 요청 패킷이 입력된 페이스로 전달되고 요청 패킷은 삭제된다.

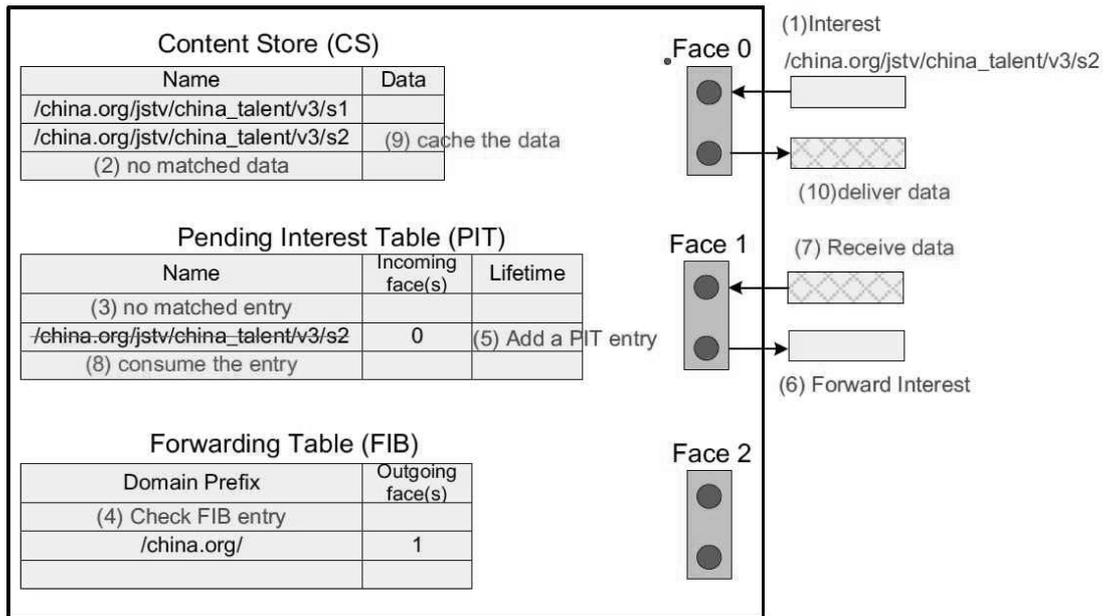


Fig. 6. The basic operation of a CCN router

그림 6의 '(2)no matched data'에서와 같이 CS에 저장된 콘텐츠 이름과 수신된 요청 패킷의 콘텐츠 이름이 일치하지 않은 경우 '(4)Check FIB entry'와 같이 요청 패킷은 FIB에 따라 콘텐츠 제공자 방향으로 라우팅되며 전달한 Interest 정보를 '(5)Add a PIT entry'와 같이 PIT에 저장한다. 콘텐츠를 보유한 노드를 만날 때까지 계속 라우팅되며 콘텐츠를 소유한 노드나 콘텐츠 제공자를 발견하면 더 이상 요청 패킷을 전달할 필요 없이 즉시 데이터 패킷 전달을 수행한다. '(8)consume the entry'와 같이 데이터 패킷은 노드의 PIT에 남겨진 정보에 의해 요청 패킷이 지나온 역방향으로 콘텐츠 요청자에게 전달되며 노드는 데이터 패킷을 전달 후 '(9)cache the data'와 같이 CS에 저장하여 동일 콘텐츠 요청 시 즉시 응답할 수 있게 한다<sup>[21]</sup>.

## ② CCN 메시지 패킷

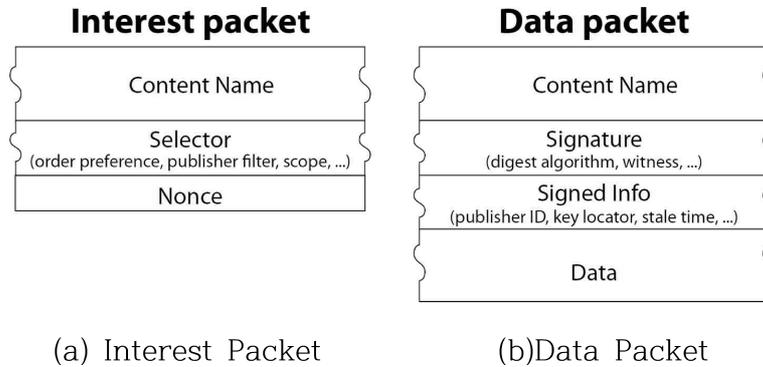


Fig. 7. CCN message packet types

그림 7은 CCN에서 사용되는 메시지 패킷인 요청 패킷(Interest Packet)과 데이터 패킷(Data Packet)의 구조를 나타낸 것이다<sup>[21]</sup>. 첫 번째는 콘텐츠 소비자가 콘텐츠를 요청하는 요청 패킷이고, 두 번째는 요청된 콘텐츠를 갖고 있는 임의 노드가 요구된 콘텐츠를 응답으로 전송하는 데이터 패킷이다. 그림 7 (a)의 요청 패킷은 콘텐츠 이름 항목에 필요한 콘텐츠 이름을 넣어 네트워크로 전송되며, IP패킷과 다르게 콘텐츠 요청자 및 제공자의 주소가 없고 모든 콘텐츠는 콘텐츠의 이름을 식별자로 사용한다. 데이터 패킷 역시 콘텐츠 이름 항목이 존재하며, 콘텐츠를 요청한 요청 패킷의 콘텐츠 이름을 그대로 복사하여 사용한다. 요청 패킷은 “Content name”, “Selector”, “Nonce”로 구성되고, 그림 7 (b)의 데이터 패킷은 “Content name”, “Signature”, “Signed Info”, “Data”로 구성된다. 요청 패킷의 Selector는 요청 패킷의 콘텐츠 이름에 기술되지 않은 이름 항목을 자세히 기술하는 항목이다. 요청 패킷을 수신한 CCN 노드는 콘텐츠 이름을 확인하여 해당하는 데이터가 CS에 존재하는 경우 데이터 패킷에 담아 응답한다. 데이터 패킷의 라우팅은 요청 패킷이 지나온 역방향 경로(RPF, reverse path forwarding)를 따라 요청자에게 콘텐츠를 전송한다. 요청 패킷의 Nonce는 랜덤하게 생성된 바이트 문자로, 요청 패킷이 브로드캐스트 되므로 서로 다른 경로를 통해 루프를 형성할 수 있어 요청 패킷을 식별하여 루프 형성을 방지하는데 사용한다. 데이터 패킷의 Signature와 Signed Info 항목은 데이터 자체의 인증과 보안성을 제공할 수 있으

며, 기존의 인터넷은 보안성을 높이기 위해 종단 간 연결 시 마다 보안 절차를 수행하지만, CCN에서는 콘텐츠를 처음 배포할 때만 수행하면 된다<sup>[22]</sup>.

그림 8은 콘텐츠 및 콘텐츠 제공자에 대한 정보를 제공하는 계층화된 콘텐츠 이름의 구조이다. CCN 메시지 패킷의 콘텐츠 이름 부분에서는 '/'를 사용해서 계층화된 이름을 사용하게 된다. 사용자는 원하는 콘텐츠의 이름을 요청 패킷에 담아 전송하고, 해당 콘텐츠에 도달할 때까지 전달되게 한다. 전달되는 중에 CCN 노드에서 데이터를 찾으면 데이터 패킷을 통해 요청 패킷의 경로를 역으로 따라가면서 사용자에게 콘텐츠를 공급하게 된다. CCN 네트워크 노드들이 콘텐츠의 이름을 참조하여 요청 패킷을 콘텐츠 제공자 호스트로 전송하기 위하여 콘텐츠 이름은 계층화된 요소들로 구성되며, 각각의 요소들은 콘텐츠 제공자 도메인 또는 호스트와 의미론적으로 연관성을 갖게 된다. 요청 패킷을 수신한 네트워크 노드들이 수신된 요청 패킷과 임시 저장되어 있는 콘텐츠와의 일치 여부를 효율적으로 판단할 수 있도록 CCN은 네트워크 패킷 헤더를 IP주소가 아닌 콘텐츠 이름으로 구성한다. 또한, 네트워크 노드가 요청 패킷을 다음 노드로 전달할 때 참조할 수 있도록 계층화된 요소들로 콘텐츠 이름을 생성 관리 하도록 제안하고 있다. 그림 8은 CCN에서 제안하는 콘텐츠 이름의 예를 보여준다<sup>[16][20]</sup>.

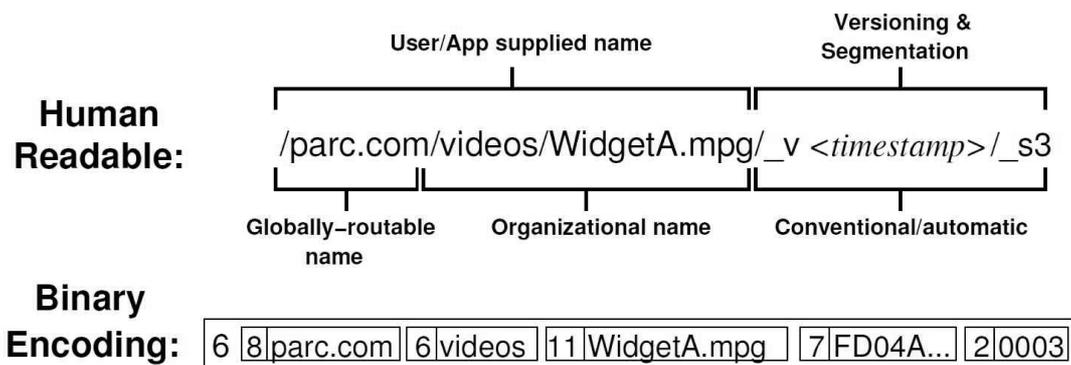


Fig. 8. Data name structure

## 2) CCN에서의 기존 라우팅 알고리즘

CCN에서의 기존 라우팅 알고리즘은 기존 호스트 중심의 라우팅과 다르게 각 콘텐츠에 이름을 붙여 사용하는 CCN의 특성을 고려한 라우팅 알고리즘으로, 사용자의 요청 콘텐츠에 따라 해당 콘텐츠를 빠르게 전달하는 것을 목적으로 라우팅을 수행한다. 따라서 CCN에서의 기존 라우팅 알고리즘에서는 사용자가 원하는 이름의 콘텐츠까지 얼마나 빠르게 도달할 수 있는지가 라우팅 알고리즘의 성능을 좌우하며, 콘텐츠에 대한 접근 시간을 향상시키기 위한 최단 경로 설정에 초점을 맞추어 연구되고 있다.

### (1) SoCCeR(services over content-centric routing)알고리즘<sup>[23]</sup>

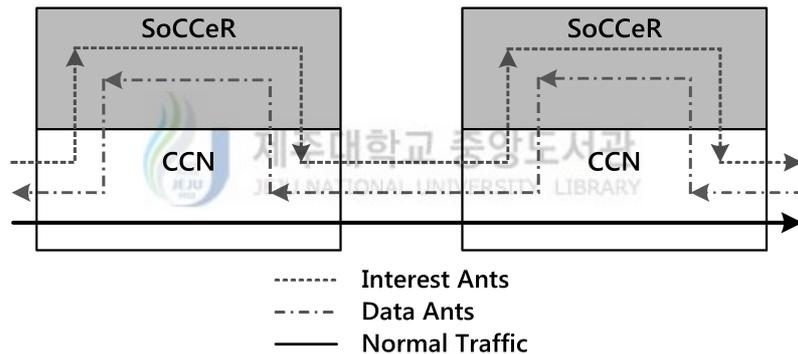


Fig. 9. The SoCCeR operation

S. Shanbhag는 데이터의 수직적인 증가를 고려하여 그림 9와 같이 SoCCeR 알고리즘을 제안하였다. 그림 9는 SoCCeR 알고리즘에서 제안하는 네트워크 흐름을 나타낸 것이다. 최상위 계층에는 정보의 효율적인 관리를 위하여 SoCCeR 계층을 정의한다. 이 계층에서는 개미집단 최적화(ACO: ant colony optimization)알고리즘<sup>[24]</sup>을 이용한다. 각 라우터의 Ant들이 ACO에 의해 경로를 최적화시키기 위해 Interest Ants와 Data Ants를 이용하여 최적 경로 테이블을 설정한다. Interest Ants는 각 노드로 전달될 때 전달 시간을 기록하는 시간 스택을 가진다. SoCCeR 노드들은 시간 스택에 자신의 시스템 시간을 담아 콘텐츠 서비스가 가능

한 노드로 향하는 확률을 갖는 임의의 여러 페이스로 Interest Ants를 전달한다. 이는 이전에 검출되지 않은 더 나은 경로 탐색을 가능하게 한다. 서비스 노드가 Interest Ant를 수신하면, 노드의 SoCCeR 계층은 시간 스택과 서비스 상태 정보를 담은 Data Ant를 Interest Ant가 전달되어 온 역경로를 이용해 전달한다. SoCCeR 알고리즘에서는 각각의 라우터가 수신한 Data Ant 내의 시간 스택과 서비스 상태 정보를 이용하여 경로를 최적화한다. 이러한 최적화된 경로로 인하여 혼잡을 피하는 장점을 가지고 있으며 또한 사용자들이 원하는 서비스에 대한 접근 시간을 향상시켜 적절한 혼잡제어가 가능하다.

(2) OSPFN(OSPF for named-data networking)라우팅 알고리즘 <sup>[9]</sup>

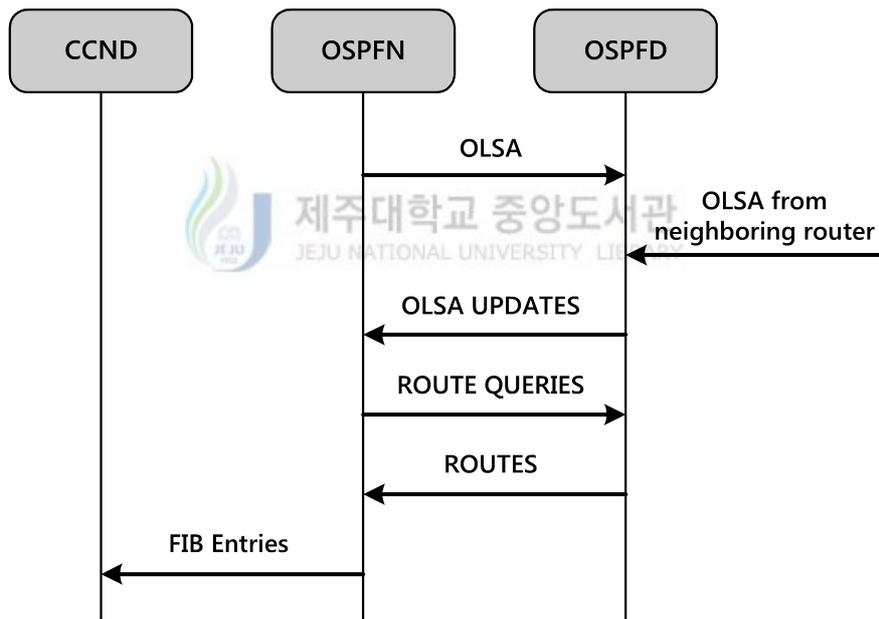


Fig. 10. The OSPFN operation

L. Wang은 OSPFN 라우팅 알고리즘을 제안하였으며, 제안된 알고리즘은 현재 IP에서 사용 중인 OSPF(open shortest path first) 프로토콜<sup>[25]</sup>을 CCN에 맞게 변형하여 이름 기반으로 라우팅하는데 있어 빠른 경로 설정과 트래픽을 최소화한다. OSPF의 방식을 그대로 사용하여 각 라우터의 LSDB(link state database)를 통해 각 라우터들은 OLSA를 플러딩하여 자신의 경로를 교환하고 최적의 경로를 찾는

다. OSPF의 다중 경로 지원하지 못하는 문제를 해결하기 위해 그림 10과 같이 CCND(content-centric networking daemon)와 OSPFD(open shortest path first daemon)를 CCN 라우터에 추가하여 정보의 효율적인 관리와 다중경로 지원에 대한 문제점을 해결하였다. CCND는 Interest와 Data 메시지 전달을 처리하고, OSPFN은 이름 정보에 기반한 OLSA를 전체 네트워크에 플로딩한다. 라우터의 OSPFD가 OLSA를 수신하면 수신한 OLSA를 OSPFN에 전달한다. OSPFN은 OLSA가 자신의 것인지 아닌지를 판별하여 자신의 OLSA가 아닌 경우 OSPFD에게 목적지로 향하는 다음 홉에 대한 정보를 요청한다. OSPFD는 자신의 라우팅 테이블에서 다음 홉 목록과 관련된 경로 비용에 대한 정보를 OSPFN에게 전달한다. OSPFN은 이 정보를 이용하여 이름에 기반한 FIB 항목을 생성하고 CCND에 삽입한다. 이에 OSPFD는 정기적으로 OLSA를 플로딩하여 오버레이 토폴로지 기반의 최단 경로 트리를 계산하여, 최적의 경로를 찾는다.

### (3) GACF(greedy ant colony forwarding)알고리즘 <sup>[26]</sup>

C. Li는 사용자가 정보를 전송받는 QoS 향상을 목적으로 GACF알고리즘을 제안하였으며, 경로 최적화 및 효율적인 전송을 위해 HA(hello ant)와 NA(normal ant)의 지능화된 두 종류의 개미를 사용하여 경로를 탐색하고 최적화한다. HA는 라우터에 의해 생성되며, 라우팅 및 포워딩 정보를 수집하는데 사용된다. NA는 데이터를 요청하는 소비자에 의해 발생되며 데이터를 검색하는데 사용된다. NA는 경로에 대한 오버헤더만을 포함하는 대신 HA는 라우팅 정보를 수집하기 위해 오버헤드, 최소 대역폭, 전체 경로의 왕복 지연, 홉의 정보 등 NA보다 더 많은 정보를 포함한다. HA는 일정 시간 간격으로 발생되며 자신의 FIB에서 랜덤하게 콘텐츠 이름을 선택하여 네트워크로 포워딩한다. 이를 수신한 노드들은 자신의 NLS(neighbour link states) 테이블에 삽입하고 이를 이용하여 최적의 경로를 찾아낸다. 따라서 주변 가능한 경로의 검색 최적화 및 재탐색을 통한 QoS 향상의 장점을 가지고 있다. 그러나 QoS를 향상하는 과정에서 사용하는 많은 정보를 포함한 HA는 지속적인 경로 재탐색에 대한 매우 높은 트래픽이 발생하는 단점을 가지고 있다.

(4)  $\alpha$ Route 라우팅 알고리즘 [27]

R. Ahmed는 인터넷에서 제공하는 정보가 증가하는 만큼 라우팅 테이블의 확장성에 중점을 두어 제안하였으며, 이름 기반의 DHT(distributed hash table)은 분산 방식으로 키 값을 매핑할 수 있다. DHT는 두 가지 구성 요소를 포함한다. 첫째로 전체 키 공간을 부분 공간으로 분리하여 물리적 노드에게 각 부분 공간을 할당하는 partitioning과 두 번째로는 홉의 제한된 수에 대한 임의의 키를 찾기 위한 메커니즘인 routing의 두 가지 구성요소를 갖는  $\alpha$ Route를 제안하였다.  $\alpha$ Route는 라우팅 테이블의 확장성을 보장하고 정보들을 효율적으로 관리하기 위해 콘텐츠의 이름으로 해시되는 Partitioning 트리로 정보에 대한 라우팅과 관리를 병행한다.

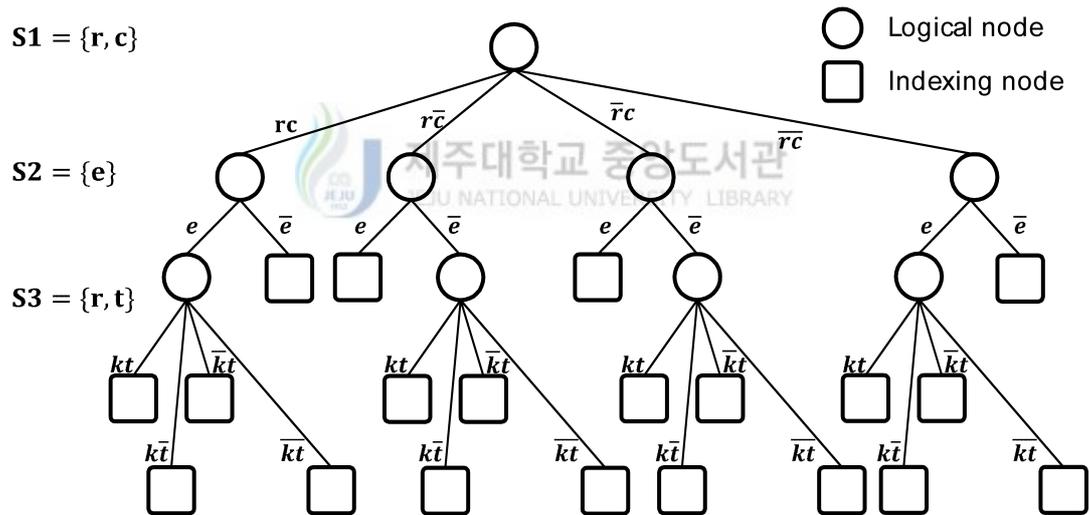


Fig. 11. Partitioning tree of the  $\alpha$ Route

그림 11은 Partitioning 트리 예제를 통해 partitioning 과정을 설명한 것이다. 트리를 통하여 정보에 대한 직접 인덱싱하므로 정보에 대한 접근 시간을 보장하고 현재의 IP기반의 라우팅 테이블에 비해 적은 규모의 라우팅 테이블 사이즈를 갖게 된다.

## (5) LANES(logical address space network extensions) 프로토콜 <sup>[28]</sup>

K. Visala는 네트워크 내부의 혼잡제어와 사용자 요청 정보를 전송받는데 있어 지연시간의 최소화를 위하여 제안하였으며, 도메인 간 라우팅 계층과 시스템의 구성 요소와의 상호작용을 제시한다. LANES 프로토콜은 저장 공간과 대역폭을 독립적으로 네트워크의 요구에 맞게 확장할 수 있다. 또한 혼잡 제어를 위하여 전달 트리를 이용하며 전달 트리를 통해 정보를 멀티캐스트 함으로써 각 도메인에서 경로를 구성한다. 따라서 효율적인 혼잡 제어를 통한 병목현상을 최소화하고 네트워크의 로드 밸런싱을 유지시키는 장점을 가지고 있다. LANES 프로토콜에서는 사용자의 최소화된 평균 지연을 위하여 라우터 캐싱 용량을 늘리는 방법을 사용하기 때문에 각각의 라우터가 최대한의 용량을 네트워크 정보에 할당하여 사용자가 정보에 접근하는 평균 접속 시간을 감소할 수 있다. 그러나 트리 구조를 사용하기 때문에 네트워크 토폴로지의 변화에 민감하지 못하다. 따라서 잦은 라우터의 추가나 변경에 대처하지 못하거나 물리적인 스토리지의 증가에 대한 한계 문제가 발생한다.



## (6) CCN에서 라우팅 알고리즘 비교 분석

CCN의 효율적인 사용을 위해 라우팅 알고리즘 설계 시 다음과 같은 요소들을 고려해야한다.

### ① 캐싱(Caching)

CCN은 네트워크 내부에서 활용되는 캐시를 사용하는데, 이는 캐시가 별도의 서버가 될 수도 있고 라우터나 CCN 라우터 자체가 캐시 역할을 수행할 수도 있다. 이에 사용자는 서버까지 도달하지 않고 라우터에서 정보를 가져와 네트워크 트래픽을 감소시킬 수 있다. 따라서 CCN에서는 네트워크 트래픽 감소에 있어 캐싱이 중요한 역할을 하며, 잘못된 캐싱 정책 사용 시, 정보가 자주 교체되는 등 비효율적인 저장을 수행할 수 있으므로 효율적인 라우팅을 위해선 실제 사용을 고려한 캐싱 정책과 그에 합당한 캐시 용량이 필요하다.

## ② 접근 시간(Information Access Time-Lookup Delay)

CCN 라우터에 정보들이 개별적인 정책에 의해 각 캐시에 저장되는데, 이는 룩업 지연 문제를 발생시킬 수 있다. 이에 사용자가 특정 정보를 요청했을 경우, 사용자의 접근 시간을 단축하기 위해 각 노드에서 효율적으로 정보를 관리하는 방법과 단시간 내에 사용자가 원하는 정보를 찾는 방법이 필요하다.

## ③ 분산(Information Distribution)

현재 인터넷은 CDN(Content Delivery Network)과 같은 방법을 사용하여 전송 효율과 데이터 접근 시간을 향상 시키려 하고 있다. CCN은 CDN의 특성을 더욱 확장시켜 모든 CCN 라우터에서 CDN의 역할을 하는 캐시를 갖고 있어 트래픽 분산에 있어 매우 높은 효율성을 가지고 있다. 그러나 정보의 업데이트 관리에 대한 문제를 발생시킬 수 있으므로, 정보에 대한 적절한 분산 정책이 필요하다.

## ④ 이동성(Mobility)

클라이언트의 이동, 콘텐츠의 이동, 네트워크의 이동으로 나눌 수 있으며, 클라이언트의 이동은 일반적인 클라이언트가 직접적으로 위치를 이동하는 것이고, 콘텐츠의 이동은 콘텐츠가 서버에서 클라이언트로 이동할 때 이동 경로 상에 있는 라우터의 캐시에 저장되어 사용되는 것을 의미한다. 네트워크 이동은 서로 다른 종류의 네트워크 간의 이동성을 말한다. 이동성은 세 가지를 모두 고려해야 한다.

## ⑤ 확장성(Scalability)

확장성은 정보 관점과 라우터 관점으로 나뉜다. 정보 관점은 정보의 증가로 인한 CCN 라우터의 캐시에 대한 확장성을 말하고, 라우터 관점은 각 라우터들을 연결하는 경로를 저장하는 라우팅 테이블에 대한 확장성을 말한다. 확장성은 이 두 가지 모두가 고려된 것을 의미한다.

## ⑥ 배포(Information Dissemination)

CCN 라우터의 캐시에 정보를 저장하는 것도 중요하지만 사용자에게 의해 정보 요

청 시 정보의 배포도 중요하다. 이에 정보가 최초 배포될 때 사용자에게 대한 유니캐스트가 아닌 특정 그룹에 멀티캐스트 형태로 배포하는 등 정보 배포를 위한 여러 가지 상황을 고려해야한다.

Table 1. Comparative analysis of routing algorithms in CCN

프로토콜	캐싱	접근시간	분산	이동성	확장성	배포
SoCCeR		○		○		
OSPFN		○		○	○	
GACF		○		○		○
$\alpha$ Route		○	○		○	
LANES	○	○				

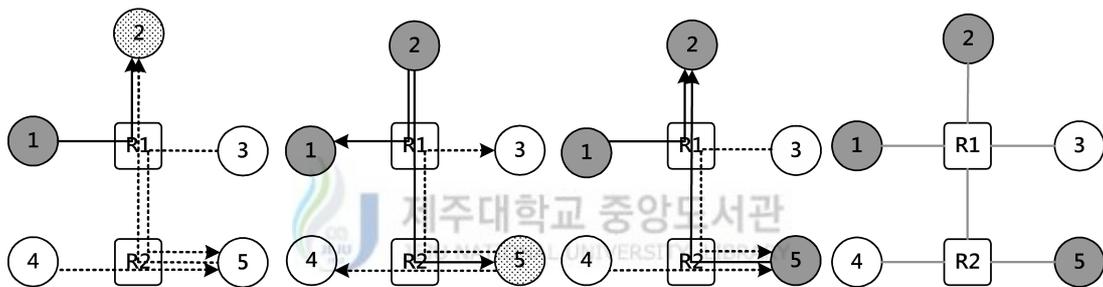
표 1은 CCN 라우팅 알고리즘들을 6가지 라우팅 설계 시 고려사항에 대해 비교 분석한 것이다. 표 1을 살펴보면 대부분의 라우팅 알고리즘이 정보를 빠르게 전달할 수 있는 접근 시간 향상에 초점을 맞춰 개발되고 있다. SoCCeR은 사용자에게 제공되는 서비스에 초점을 맞춰 정보를 빠르게 전달할 수 있는 접근 시간 향상을 목적으로 하고 있고, OSPFN은 OSPF의 짧은 경로에 대한 장점으로 접근시간을 향상시키고 여러 경로를 저장하기 위해 라우팅 테이블의 확장성에 집중하고 있다. GACF는 다중 경로를 이용해 정보의 배포를 통한 접근 시간을 향상시키고, 이들은 브로드캐스트 형식을 띄고 있어 이동성에 강점을 가지고 있다.  $\alpha$ Route는 정보의 분산과 테이블의 확장성, 접근 시간을 향상시키는 데 주목하고 있고, LANES는 내부의 전송 트리를 이용하여 정보를 전송하고 라우터 자체의 캐싱을 향상시켜 정보에 대한 접근시간을 향상시킨다.

CCN에서 멀티캐스트 라우팅 알고리즘을 제안하기 위해서는 멀티캐스트 라우팅 정보에 대한 라우팅 테이블 확장성과 멀티캐스트 그룹 관리에 대한 클라이언트의 이동성이 고려되어야 한다. 표 1을 보면 라우팅 테이블의 확장성과 클라이언트의 이동성이 고려는 OSPFN 알고리즘만이 하고 있다. 이에 본 논문에서는 OSPFN 알고리즘을 이용하여 CCN에서의 멀티캐스트 라우팅 알고리즘을 제안하고자 한다.

### 3) CCN에서 그룹 통신 알고리즘

#### (1) CCN에서의 실시간 그룹 통신<sup>[7]</sup>

IP 네트워크의 실시간 N-to-N 동적 그룹 통신에 효율적인 가십 프로토콜<sup>[5][6]</sup>인 RRG(redundancy reduction gossip)는 메시지 패킷을 전송하기 위해 자신의 이웃하는 노드 중 임의의 수만큼의 노드를 선택하여 메시지를 전송하는 프로토콜이다. 주기적으로 네트워크 내에 이웃하는 노드를 임의의 수만큼 선택하여 메시지를 전달하므로 네트워크 내에 불필요한 중복 메시지의 수가 많다. 이러한 트래픽 과부하 문제를 해결하기 위해 CCN의 캐시 기능을 활용하여 CCN에서 RRG 프로토콜을 적용한다.

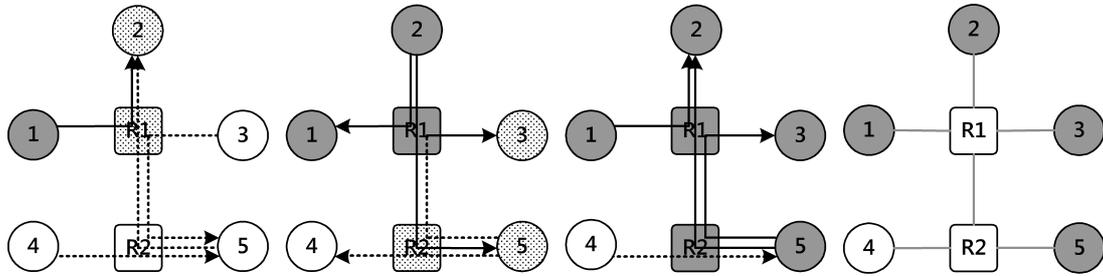


(a) Phase 1: Greeting (b) Phase 2: Response (c) Phase 3: Closure (d) After all phases

Fig. 12. The RRG operation in IP network

그림 12는 IP 네트워크에서 주기적으로 메시지를 전달하는 RRG의 동작 과정을 나타낸 것으로 노드 1의 정보 프레임을 하나의 주기에 3 단계로 메시지가 확산 것을 볼 수 있다. 그림 12 (a)에서 1단계인 Greeting 단계에서는 노드들은 자신의 임의로 선택한 노드들에게 자신이 보유한 정보를 포함하여 Greeting 메시지를 전달한다. 노드 1이 노드 2로 데이터가 포함된 Greeting 메시지를 전달하는 것을 볼 수 있다. 나머지 노드들은 메시지 정보가 없는 Greeting 메시지를 전달한다. 그림 12 (b)의 2단계는 Response 메시지 전달로 자신의 임의로 선택한 노드들에게 역경로로 수신된 메시지 정보를 포함하여 전달한다. 노드 2가 수신한 메시지가 포함된 응답 메시지를 1과 5에게 전달한다. 마지막 3단계는 그림 12 (c)에서와 같이 Closure 메시지에 1단계와 2단계에서 수신한 메시지를 포함시켜 원래의 경로로

전달한다. 1번 노드와 5번 노드가 2번 노드에게로 메시지가 포함된 Closure 메시지를 전달한다. 모든 단계가 끝나고 메시지를 전달받은 노드는 그림 12 (d)와 같다.



(a) Phase 1: Greeting (b) Phase 2: Response (c) Phase 3: Closure (d) After all phases

Fig. 13. The RRG operation in CCN

그림 13은 CCN에서 RRG의 동작 과정으로 1단계인 Greeting 단계에서는 그림 13 (a)와 같이 노드 1이 노드 2로 데이터가 포함된 Greeting 메시지를 전달하는 것을 볼 수 있다. 이때 CCN은 라우터의 캐시를 이용하여 R1에도 메시지가 저장된다. 그림 13 (b)의 2단계는 Response 메시지 전달로 노드 2가 수신한 메시지를 포함한 응답 메시지를 1과 5에게 전달하는 과정에서 CCN 라우터 R2에게도 메시지가 저장된다. 그림 13 (c)의 3단계에서 Closure 메시지를 1번 노드와 5번 노드가 2번 노드에게로 메시지가 포함된 Closure 메시지를 전달한다. 여기까지는 기존 IP 네트워크에서와 동일하게 동작한다. 하지만 CCN의 경우 5번 노드가 3번 노드에게로 보내는 정보 메시지가 포함되지 않은 Closure 메시지를 전달하는 과정에서 CCN 라우터 R1이 보유하고 있는 정보 메시지를 담아 3번 노드에게 전달하므로 하나의 주기 동안 기존 IP 네트워크 보다 더 많은 노드에게 메시지를 전달한다.

CCN에서 RRG는 IP 네트워크에서 RRG 보다 메시지를 전달하는데 중복 메시지의 수가 줄어들고 대기 지연 시간을 개선하며 대규모의 네트워크 환경에서 확장성을 제공한다. 하지만 네트워크 내에서 라우팅 경로 없이 임의로 선택한 노드에게로 대용량의 메시지를 전달하는 방식인 RRG는 네트워크 트래픽 과부하 문제를 발생시키며 복잡한 네트워크 구조에서 대용량의 데이터를 실시간 전송한다는 것은 어렵다.

## (2) CCN에서 오디오 컨퍼런스 톨<sup>[8]</sup>

CCN에서 오디오 컨퍼런스 톨인 ACT는 CCN의 상위 계층인 서비스 계층의 어플리케이션 설계에 대한 연구 중 하나이다. ACT는 각 컨퍼런스에서 음성 데이터를 제공하는 화자뿐만 아니라 진행 중인 컨퍼런스를 발견하고 각 화자로부터 음성 데이터를 가져오기 위한 도구이다. ACT는 세 가지 기본 작업을 수행하는데 첫 번째 작업은 모든 회의에 대한 스케줄과 진행 중인 회의에 대한 최신 정보와 사용자들이 접속하기를 원하는 각 진행 중인 회의의 화자를 찾는 것이다. 두 번째 작업은 컨퍼런스 정보에 대한 각 요청 패킷의 요구를 네트워크의 모든 곳으로 플로딩하는 것이며, 마지막 세 번째는 미디어 데이터 처리와 사용자 인터페이스를 설계하는 것이다. 그림 14는 ACT 설계에 대한 구성도를 나타낸 것이며 회의 검색에 대한 별도의 모듈로 구성한 이유는 회의 참가 작업을 용이하게 하기 위해서이다.

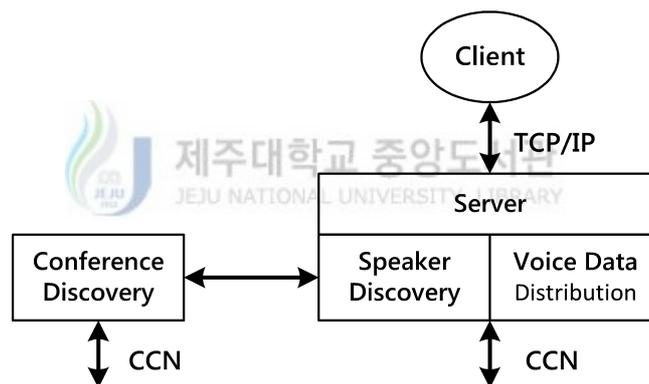


Fig. 14. An Overview of ACT Design

ACT의 각 사용자가 진행 중인 회의나 스케줄 목록을 검색 위해 사용자는 회의 데이터에 대한 이름의 구성은 /ccn/broadcast/conference/conference-list로 첫 번째 /의 이름 구성요소는 네트워크의 범위, 두 번째 /는 브로드캐스트 이름, 세 번째는 어플리케이션 유형 식별, 마지막의 이름 구성 요소는 참가자가 관심 있는 회의의 데이터에 대한 정보이다.

 /ccn/broadcast/conference/conference-list/ccn2014

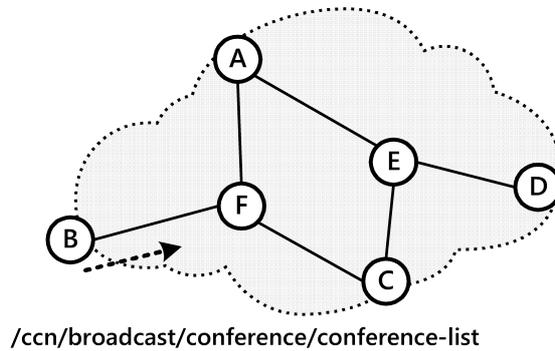


Fig. 15. Conference Discovery

그림 15는 회의를 검색하는 과정을 나타낸 것으로 노드 A가 'ccn2014'라는 이름의 회의를 발표했다고 가정하자. 노드 A는 conference-list 접두사를 갖는 요청 패킷을 수신하면 회의의 설명을 포함한 데이터 패킷을 응답할 것이다. 데이터 패킷을 요청자에게 전달할 때 데이터 패킷이 지나가는 경로상의 라우터들은 CS에 데이터 패킷을 저장한 후 conference-list에 해당하는 요청 패킷을 수신하면 저장된 회의 설명 데이터를 전달한다. 경로상의 라우터에 저장된 데이터 패킷은 회의의 데이터 정보로 회의가 개최되는 동안 사용되는 데이터로 유효시간을 주어 일정 시간 동안만 사용할 수 있게 한다.

## 2. IP 기반의 멀티캐스트 라우팅 프로토콜

### 1) 멀티캐스트 프로토콜

멀티캐스트란 데이터를 전송하는 노드가 연결된 출발지 라우터에서 해당 멀티캐스트 그룹 멤버가 연결된 모든 목적지 라우터까지 데이터 패킷을 가장 효율적으로 전달하는 것을 말한다. 멀티캐스트 프로토콜은 최소 비용의 경로를 거쳐 모든 목적지 라우터에 전달되어야 하고 가장 작은 수의 멀티캐스트 데이터 복사본을 생성할

수 있어야 한다. 또한 N개의 멀티캐스트 그룹이 존재하면 N개의 최소 비용 트리가 필요하다.

일반적으로 멀티캐스트 라우팅 알고리즘에 의해 만들어진 멀티캐스트 트리는 소스 기반 트리(Source Based Tree)와 코어 기반 트리(Core Based Tree)의 두 가지로 분류할 수 있다. 소스 기반 트리를 구성하는 알고리즘은 멀티캐스트 그룹에서 각 소스 노드 당 하나의 트리를 구성하게 된다. 반면에 코어 기반 트리와 같은 공유 트리를 구성하는 알고리즘은 멀티캐스트 그룹의 모든 노드를 위해 공유되는 단지 하나의 트리, 즉 공유 트리를 구성하게 된다<sup>[29]</sup>.

소스 기반 멀티캐스트 알고리즘으로는 Distance Vector Multicast Routing Protocol(DVMRP)<sup>[30]</sup>, Multicast Open Shortest Path First(MOSPF)<sup>[31]</sup>, 그리고 Protocol Independent Multicast-Dense Mode(PIM-DM)<sup>[32]</sup> 등이 있다. 소스 기반 트리에서 각 라우터는 각 그룹을 위한 하나의 최단 경로를 갖는 것이 필요하다. 멀티캐스트 그룹마다 서로 다른 다음 홉 라우터들에 대한 정보를 유지해야 한다.

공유 트리를 사용하는 코어 기반 멀티캐스트 알고리즘으로는 Core Based Tree(CBT)<sup>[10]</sup>와 Protocol Independent Multicast Sparse Mode(PIM-SM)<sup>[33]</sup>이 있다. 이런 코어 기반 멀티캐스트 라우팅 알고리즘은 코어 혹은 센터라 불리는 노드를 루트로 하여 공유 트리를 구성하고, 트리의 코어에서 멀티캐스트 그룹의 각 멤버로 최단 경로가 형성된다.

## 2) 코어 기반 멀티캐스트 라우팅 프로토콜(CBT, Core Based Tree)

코어 기반 멀티캐스트 라우팅 알고리즘을 설계하기 위해서는 트리 구성 및 관리를 위한 코어 위치 알고리즘과 데이터 패킷 전달과 제어 패킷 전달에 대한 포워딩 알고리즘이 필요하다.

### (1) 코어 위치 알고리즘

CBT 프로토콜은 멀티캐스트 서비스에 직접적으로 관련되는 라우터 혹은 네트워크들만을 포함하는 공유 멀티캐스트 트리를 구성하고 유지하기 위해 제안된 방식

이다<sup>[10]</sup>. 각 소스마다 최단 경로 트리를 유지하는 멀티캐스트 프로토콜인 DVMRP와 MOSPF는 트리를 구성하는 라우터들이 유지해야 하는 트리 상태 정보가 증가하므로 멀티캐스트 그룹이 대규모인 경우 요구를 충족시키기에는 부적합한 특성을 가지고 있다. 이를 해결하기 위해 그룹 당 하나의 멀티캐스트 트리를 유지하는 공유 트리 멀티캐스트 프로토콜들이 개발되었으며 그 중 하나가 CBT 알고리즘이다.

Table 2. scalability comparison in CBT and DVMRP

그룹 수	그룹당 멤버수(G)	그룹당 소스수(S)	DVMRP 라우터수(O)	CBT 라우터수(O)
10	20	10%	20	10
		50%	100	
		100%	200	
100	20	10%	200	100
		50%	1,000	
		100%	2,000	
1000	20	10%	2,000	1,000
		50%	30,000	
		100%	40,000	

멀티캐스트 트래픽을 전송하는 소스의 수를 S라 하고 그룹 멤버의 수를 G라 하면, 소스 기반 멀티캐스트 트리는  $O(S \cdot G)$ 의 규모의 확장성(scalability)을 갖는다. 표 2에서 보는 것과 같이 DVMRP의 경우 소스의 수가 증가함에 따라 라우터의 수가 증가하는 것을 볼 수 있다. 하지만 CBT의 경우 소스수와 상관없이 그룹의 수만큼만 멀티캐스트 트래픽을 전송하는 라우터가 존재한다. 이를 통해 CBT가 규모의 확장성에 있어 기존의 소스 기반의 멀티캐스트 프로토콜의 방식에 비하여 우수한 특성을 갖는다는 것을 알 수 있다. 소스 기반의 멀티캐스트 트리의 경우 각 소스 그룹 쌍에 대하여 트리 정보를 유지해야 하지만 CBT 방식의 경우 소스에 상관없이 수가 늘어난다고 해서 네트워크 정보량이 급격히 증가하지는 않는다. 또한 CBT 방식은 모든 라우터에 주기적으로 IGMP 호스트 멤버십 정보를 전송하지 않으므로 링크 대역폭 등의 네트워크 자원 측면에서도 우수하다.

공유 트리를 사용하는 멀티캐스트에서 코어를 선정하기 위해서는 멀티캐스트 수신자의 위치, 코어를 통해 흐르는 트래픽의 양, 멀티캐스트 송신자의 위치, 트래픽 지연 감소 및 대역폭 요구 등의 고려해야하는 여러 사항들이 있다. 코어 선택 방법은 크게 세 가지 분류로 나눌 수 있다. 이는 임의 코어 선택 방법, 네트워크 위상 기반 코어 선택 방법, 그룹 기반 코어 선택 방법이다<sup>[34][35]</sup>.

임의의 코어 선택 방법은 멀티캐스트 그룹, 네트워크 위상, 이전 코어에 대한 정보 등 선택 기준 없이 코어를 선택하는 방법이다. 이는 노드의 멀티캐스트 그룹 멤버와 상관없이 임의로 코어를 선택하거나 또는 그룹 멤버 중에서 임의로 노드를 코어로 선택하는 방식이다. 이 방법은 코어의 위치가 코어 기반 트리의 성능에 영향을 미치지 않는 경우에 적합하다. 임의의 선택 방법은 그룹 생성에 대한 어떤 정보도 필요로 하지 않으므로 코어 선택이 간단하고 코어 선택에 있어 멀티캐스트 그룹 멤버 정보, 네트워크 위상 등을 계산하지 않으므로 계산 복잡도를 줄일 수 있다. 하지만 경로 비용이 고려되지 않은 코어의 선택은 전송 지연을 발생시킬 것이다.

위상 기반 코어 선택 방법은 임의의 코어 선택 방법보다 향상된 코어 선택을 하기 위하여 네트워크 위상 정보를 기반으로 코어를 선택하는 방법이다. 이 방법은 전체 위상 정보를 필요로 하는데, 이 정보는 실제 변화를 정밀하게 나타내지 못하므로 위상 재계산하는데 걸리는 시간이 알고리즘의 성능을 좌우한다. 그러나 멀티캐스트 그룹 멤버 및 송신 노드에 대한 추가 정보를 필요로 하지 않는 것이 이 방법의 장점이다. 위상 기반 코어 선택 방법은 위상 정보만을 기반으로 코어를 선택하기 때문에 멀티캐스트 그룹 또는 그룹 멤버의 변화에 상관없이 코어가 선택된다. 따라서 네트워크의 모든 노드에 대한 최대 지연 또는 최대 흡이 최소인 노드, 즉 네트워크의 중심에 위치한 노드가 코어로 선택될 확률이 높다. 이 방법의 장점은 그룹 기반 코어 선택 방법에 비해 계산량이 작고, 코어의 이동이 거의 없어서 견고성을 가진다는 점이다. 단점은 각 그룹의 코어가 네트워크의 중심에 분포하게 되어 부하가 집중되고 멀티캐스트 그룹 멤버가 네트워크의 특정 위치에 밀집되어 형성되는 경우는 성능이 저하된다는 점이다.

임의 코어 선택 방법과 위상 기반 코어 선택 방법은 멀티캐스트 그룹 멤버들의 분포 형태에 따른 적합한 코어를 선택하지는 못한다. 따라서 멀티캐스트 그룹 내의

송신 노드와 수신 노드의 위치 정보를 바탕으로 코어를 선택하는 알고리즘이 필요하다. 그룹 기반 코어 선택 방법은 위상과 그룹 멤버에 대한 정보를 바탕으로 코어를 결정하므로 멤버의 분포 변화에 효율적이고 신뢰성 있게 코어를 선택할 수 있다. 이 방법은 코어 선택 방법 중에 가장 복잡하지만 최적의 코어를 선택할 수 있다.

그룹 기반 코어 선택 방법 중 멀티캐스트 그룹 멤버의 중앙에 위치한 멤버를 코어로 선택하는 방법은 위상 정보와 멀티캐스트 그룹 멤버에 대한 정보를 바탕으로 코어를 선택하는 방법과 모든 수신 노드에서의 지연과 대역폭 성능을 계산한 후에 최적의 수신 노드를 코어로 선택하는 방법이 있다. 이 방법은 여러 코어 선택을 위한 알고리즘 중 가장 최적의 성능을 가지며 특히 멀티캐스트 그룹 멤버들이 밀집되어 있는 경우에 매우 효율적이다.

## (2) 포워딩 알고리즘

CBT 알고리즘은 각 그룹에 대해 다른 CBT를 허가하는 것을 제외하고는 스패닝 트리 알고리즘<sup>[19]</sup>과 비슷하다. 각 그룹을 위한 멀티캐스트 트래픽은 발신지를 고려하지 않고 같은 배달 트리를 통해 패킷을 송·수신한다. 그림 16은 멀티캐스트 트래픽이 어떻게 그룹의 모든 구성원에게 전송되는지를 보여준다. CBT는 코어 라우터(core router)를 중심으로 해서 다수의 CBT 라우터를 연결하는 공유 트리를 형성하는 방법을 택하고 있다.

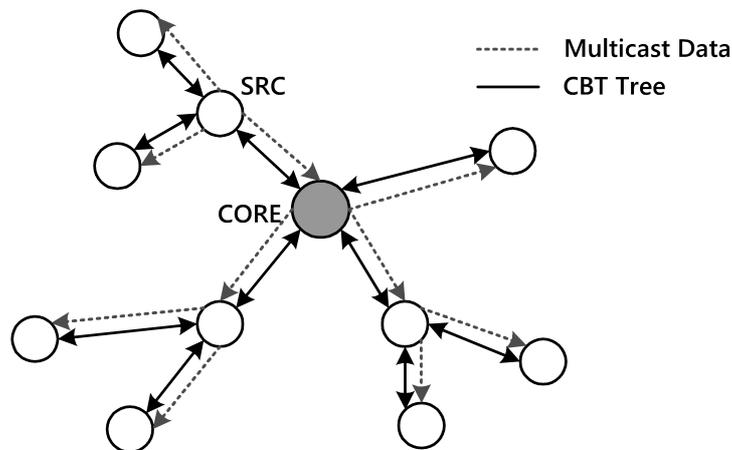


Fig. 16. Packet delivery in CBT

그림 17은 멀티캐스트 경로 설정 과정을 나타낸 것이다. 자신의 서브넷에 그룹 멤버가 존재함을 확인한 지정 라우터는 그룹과 코어 리스트를 담고 있는 JOIN\_REQUEST 메시지를 코어 라우터를 향하여 보내게 됨으로써 트리 참여 프로세스를 시작한다. 메시지를 중계하는 CBT 라우터들은 가입 상태를 기억하고 있다가 지정 라우터로 되돌아오는 JOIN\_ACK 메시지를 릴레이하며 트리를 완전히 구성하게 된다. 또한 JOIN\_REQUEST 메시지에 대한 정보가 경로상의 CBT 라우터에 이미 있는 경우 코어 라우터까지 JOIN\_REQUEST 메시지를 전달하지 않고 CBT 트리 상에 위치한 다른 라우터에 의해 JOIN\_REQUEST 메시지를 보내온 역 경로를 통해 지정 라우터에게 JOIN\_ACK 메시지를 보내게 된다. 멤버의 참여 과정이 이루어지고 나면 모든 멀티캐스트 데이터는 송·수신은 코어 라우터를 경유하여 일어난다.

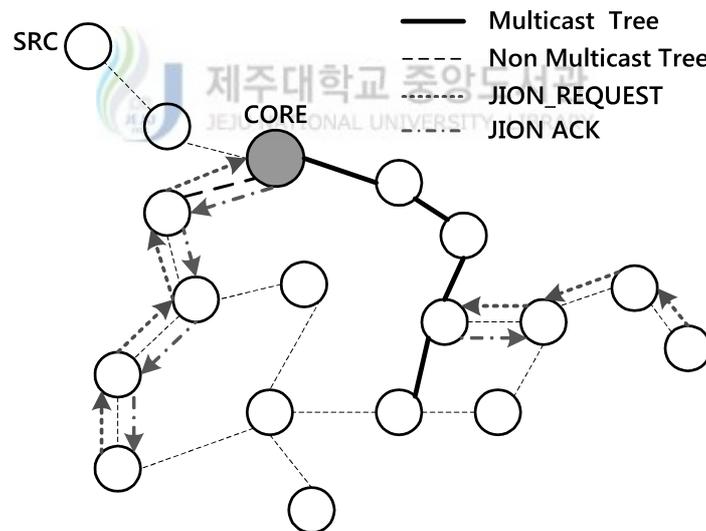


Fig. 17. Multicast route discovery in CBT

### III. 새로운 CMR-OSPFN(Core-based Multicast Routing algorithm using OSPFN) 알고리즘을 사용한 효율적인 데이터 전달 방식

멀티캐스트 그룹 통신은 양방향 미디어 어플리케이션, N-스크린 기반의 1:N 미러링 플랫폼, 그룹채팅, 음성 및 화상회의, 웹 기반 수업, 텔레프레젠틸, 온라인 분산 협업 환경, 가상 현실, 멀티플레이어 게임, 소셜네트워킹 프로그램, 소셜 게임 및 네트워킹 기기간 콘텐츠 공유 등 다양한 분야에서 기본 토대가 되고 있다. 그룹 통신은 그룹 멤버 중 누군가가 데이터나 제어 패킷을 발생시키면 모든 노드에게 동시에 전송되는 것을 의미한다.

CCN에서 그룹 통신을 위한 연구인 CCN 기반 SNS 플랫폼 구축에서는 CCN의 라우팅 기능을 응용프로그램으로 구축하였다<sup>[4]</sup>. 이것은 별도의 콘텐츠 서버 없이 사용자 기기들 간에 직접 콘텐츠를 제공하는 방식이다. 이 방식은 스마트 폰과 같은 이동 기기가 한 번의 콘텐츠 전송으로 동일 콘텐츠를 원하는 많은 기기들에게 동시에 전달할 수 있다. 또한 그룹 통신 방식인 채팅 기능 구축을 위해 콘텐츠 발생에 대한 제어 메시지를 전송한다. 이것은 채팅 그룹에 참여 하였으나 콘텐츠를 요청하지 않은 그룹 내 클라이언트에게 콘텐츠를 전송하기 위한 방식이다. CCN은 콘텐츠를 요청한 클라이언트로만 콘텐츠를 전송하므로, 콘텐츠를 생성한 클라이언트는 콘텐츠 발생에 대한 제어 메시지를 전송하도록 설계하였다. 제어 메시지를 수신한 클라이언트는 콘텐츠 발생 여부를 인지하고, 그에 해당하는 콘텐츠를 요청하여 콘텐츠를 제공받는다. 이는 그룹 통신을 위해 그룹 참여 메시지 외에도 그룹 내 콘텐츠가 발생할 때마다 제어 메시지를 전송하므로 오버헤드 문제를 가져온다.

CCN에서 P2P 실시간 그룹 통신은 응용 계층의 멀티캐스트 기법인 가십 프로토콜(gossip protocol)<sup>[5][6]</sup>을 적용하여 설계되었다<sup>[7]</sup>. 가십 프로토콜은 멀티캐스트 라우팅 테이블 없이 자신의 이웃하는 노드 중 임의의 수만큼 노드를 선택하여 메시지를 전송하는 방식이다. 이는 캐시 기능을 가지지 않는 IP 기반 네트워크에 비해 중복 메시지의 수를 줄일 수 있어 대기 지연 시간을 개선하였다. 또한 라우팅 테이블

블 형성 및 관리가 필요하지 않으므로 대규모의 네트워크 환경에서 확장성을 제공한다. 하지만 네트워크 내에서 라우팅 경로 없이 임의로 선택한 노드에게 대용량의 메시지를 전달하므로 네트워크 트래픽의 과부하 문제를 발생시킨다.

CCN에서 오디오 컨퍼런스 툴(ACT, audio conference tool)은 CCN의 상위 계층인 어플리케이션 계층을 이용하여 개발되었다<sup>[8]</sup>. 이 툴은 회의 검색에 대한 별도의 모듈을 설계하여 실시간 그룹 통신을 제공한다. 하지만 별도의 모듈을 설계해야 하는 ACT의 경우 다양한 그룹 통신에 적용하기에 적합하지 않다.

이와 같이 CCN에서 이루어지고 있는 그룹 통신을 위한 멀티캐스트 전송 방식의 제안들은 네트워크 계층이 아닌 응용 계층을 이용하여 개발되고 있으며, CCN에서 네트워크 계층을 이용한 멀티캐스트 라우팅 알고리즘에 대한 연구가 이루어지지 않고 있다. 이에 CCN에서 멀티캐스트 라우팅 알고리즘을 사용한 새로운 데이터 전송 방식에 대한 연구가 필요하다.

본 논문에서는 CCN 기반의 효율적인 멀티캐스트 라우팅 알고리즘을 제안하고, 이를 사용한 새로운 데이터 전송 방식을 제안하고자 한다. 제안하는 CCN에서 코어 기반 멀티캐스트 라우팅 알고리즘(CMR-OSPFN, Core-based Multicast Routing algorithm using OSPFN)에서 기존 CCN 라우팅 알고리즘 OSPFN에 멀티캐스트 이름 접두사 메시지를 추가하여 그룹 관리 기능을 구현한다. 또한 이는 그룹 관리와 코어 라우터의 우선순위 설정을 위해 OSPFN의 라우팅 테이블을 수정한다. 제안된 CMR-OSPFN 알고리즘은 IP 멀티캐스트 라우팅 알고리즘인 CBT와 같은 공유 트리 구조를 갖고 코어에서 멀티캐스트 트리를 관리하는 방법을 사용한다.

기존 CCN 라우터는 콘텐츠를 요청한 사용자의 역경로로만 데이터 패킷을 전달한다. 하지만 제안하는 CMR-OSPFN 알고리즘에서는 하나의 요청으로 전체 그룹 멤버에게 데이터 패킷을 전달해야 하므로, TOS(type of service) 필드를 갖는 기존 CCN의 메시지 패킷을 확장한 새로운 형태를 제안한다.

## 1. 멀티캐스트 전송 방식

CCN 데이터 전송 방식은 하나의 데이터 패킷에 대해 하나의 요청 패킷을 전달한다. 또한 이는 멀티패스로 요청된 동일한 데이터 패킷에 대해 멀티캐스트 전송이 가능하다. 여기서 멀티캐스트 전송은 여러 사용자가 동일한 데이터 패킷에 대한 요청 패킷을 하나의 라우터로 전송하였을 경우, 라우터가 하나의 데이터를 이용하여 여러 사용자에게 전달하는 것을 의미한다.

그림 18은 CCN에서 데이터 전달 과정을 나타낸 것이다. 클라이언트 A, B가 동일한 콘텐츠를 요청하였을 경우 CCN 라우터 1은 콘텐츠 제공자로 콘텐츠를 요청한다. 또한 이는 자신의 PIT에 A, B의 요청 패킷이 수신된 페이스를 콘텐츠 이름과 함께 저장한다. CCN 라우터 1이 콘텐츠 제공자로부터 데이터 패킷을 수신하면, PIT 정보를 이용해 데이터 패킷을 A, B에게 멀티캐스트한다. 이 멀티캐스트 전송은 CCN 라우터 1이 A가 요청한 콘텐츠 요청 패킷을 콘텐츠 제공자에게 전송하고 데이터 패킷을 기다리는 동안, B로부터 동일한 콘텐츠에 대한 요청 패킷이 수신되는 과정에서 이루어진 것이다. 이는 하나의 라우터에 동일한 콘텐츠 요청에 대한 멀티 패스가 형성 가능한 CCN의 특성 때문이다.

클라이언트 E, G가 동일한 콘텐츠를 CCN 라우터 2에게 요청한다. 이때 CCN 라우터 2는 CCN 라우터 1과 달리 CS에 요청에 해당하는 데이터가 저장되어 있다. CCN 라우터 2는 콘텐츠 제공자로 데이터를 요청하지 않고 바로 CS의 데이터 패킷을 전달한다. 그러므로 E의 요청 패킷을 수신한 후 G의 요청 패킷을 수신하였을 경우라면, E의 요청에 대한 CS의 데이터 패킷을 전송한 후 다시 G의 요청에 대한 데이터 전송이 이루어진다. 이 경우에는 A,B의 요청과 같이 멀티캐스트 전송이 이루어지지 않는다.

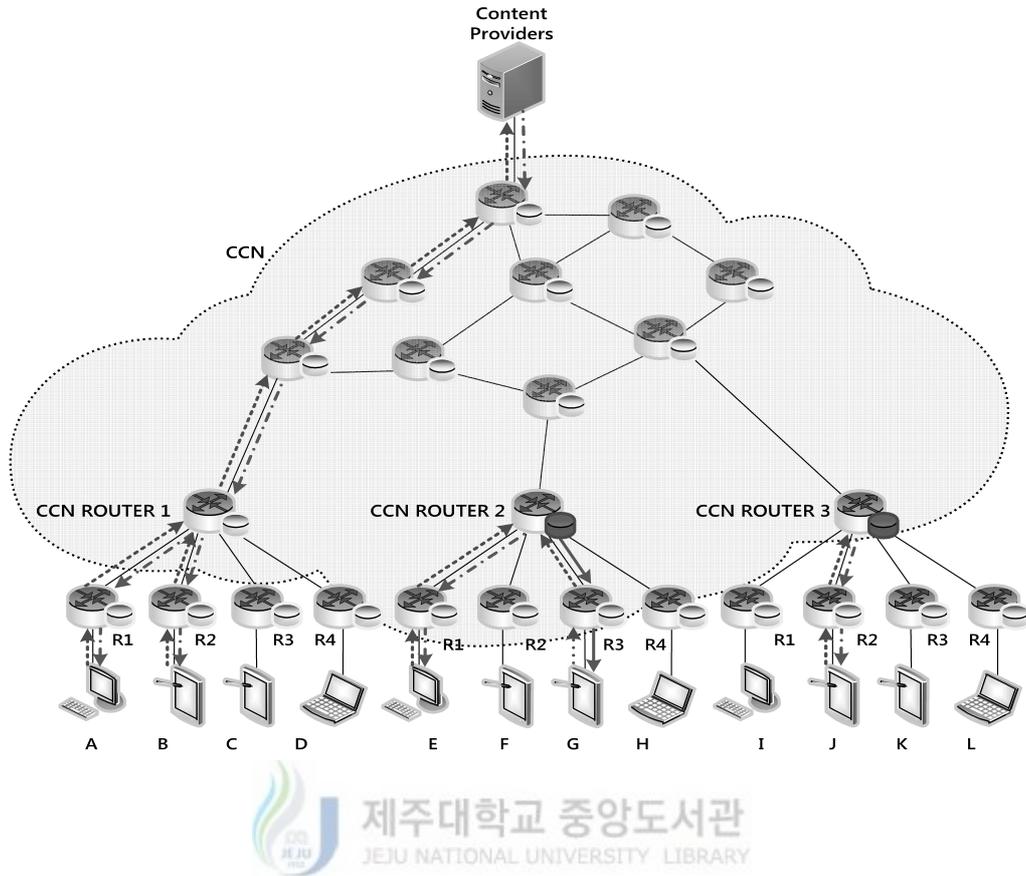


Fig. 18. An example of data forwarding in CCN

IP 멀티캐스트 라우팅 프로토콜인 CBT는 대표적인 공유 트리 구조를 갖는다<sup>[10]</sup>. 이는 양방향 트리를 구성하며, 이를 통해 수신자들은 수신자인 동시에 송신자의 역할을 수행한다. 또한 CBT는 멀티캐스트 그룹 당 하나의 멀티캐스트 트리를 유지한다. 이는 하나의 그룹 당 하나의 코어 라우터를 중심으로 다수의 라우터를 연결하여 트리를 형성한다. 이러한 CBT는 광범위한 지역에 분포된 사용자들 간 멀티캐스트 그룹 통신보다는 특정 영역에 사용자가 밀집된 지역에서 멀티캐스트 그룹 통신으로 더욱 적합하다.

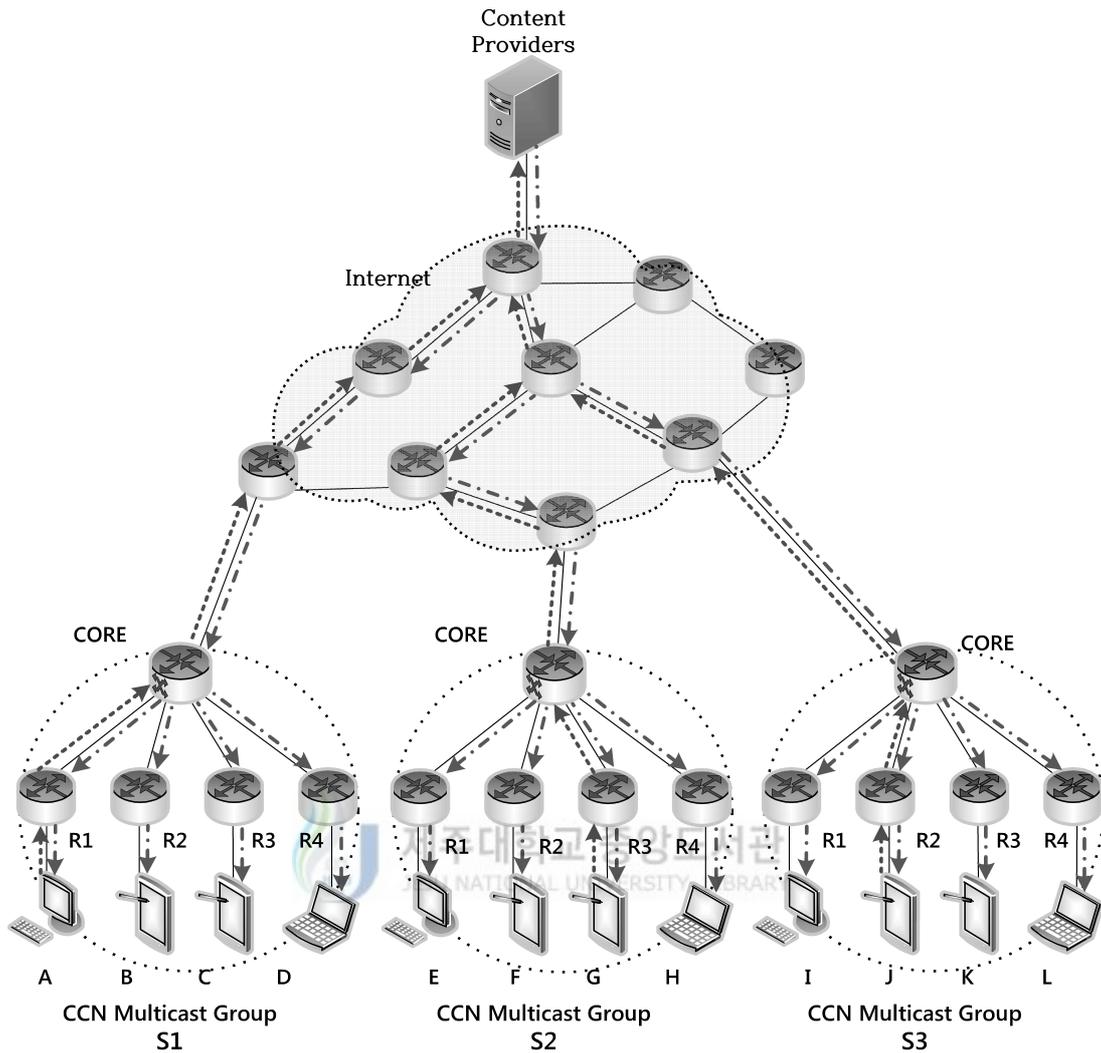


Fig. 19. An example of data forwarding in CBT

그림 19는 CBT에서 멀티캐스트 데이터 전달 과정을 나타낸 것이다. 이는 3개의 멀티캐스트 그룹인 S1, S2, S3가 형성되어 있으며, 각 그룹을 관리하는 코어라우터를 중심으로 라우터, 클라이언트, 콘텐츠 제공자인 서버와 연결되어 있다. 여기서 데이터 요청은 클라이언트로부터 서버까지 유니캐스트 방식으로 이루어지며 이를 수신한 서버가 콘텐츠를 멀티캐스트 그룹 내로 전송할 때 멀티캐스트 한다.

그림 19에서 S1 영역을 살펴보면 클라이언트 A가 콘텐츠를 요청하면 콘텐츠 제공자는 콘텐츠를 코어 라우터에게 전달한다. 콘텐츠를 제공 받은 코어 라우터는 멀티캐스트 라우팅 테이블을 이용해 멀티캐스트 그룹에게 콘텐츠를 전달한다. 콘텐츠를 전

달받은 라우터 R1, R2, R3, R4은 클라이언트들에게 콘텐츠를 전달한다.

하지만 S1, S2, S3 영역의 클라이언트들이 동일한 콘텐츠를 계속적으로 요청할 경우, 콘텐츠 요청시 마다 콘텐츠 제공자로 콘텐츠를 요청한다. 이는 콘텐츠 제공자의 IP 주소를 이용하여 콘텐츠를 요청하므로, 클라이언트가 요청하는 콘텐츠가 어떤 콘텐츠인지 상관없이 콘텐츠의 요청 메시지는 IP 주소에 해당하는 콘텐츠 제공자로 전송된다. CBT는 특정 영역내의 많은 사용자가 동일한 콘텐츠를 계속적으로 요청한다면, 콘텐츠 제공자로 트래픽 집중 현상을 일으켜 심각한 병목현상을 발생한다.

## 2. CMR-OSPFN(Core-based Multicast Routing algorithm using OSPFN)

본 논문에서는 CCN에서 특정 영역에 사용자가 밀집된 지역에서 효율적인 그룹 통신을 위한 코어 라우터를 중심으로 멀티캐스트 그룹인 CBT 트리 구조를 형성한다. 그림 20은 본 논문에서 제안하는 CMR-OSPFN 알고리즘을 사용한 새로운 멀티캐스트 데이터 전송 과정을 나타낸 것이다.

코어 기반 멀티캐스트 라우팅 알고리즘을 설계하기 위해 트리 구성 및 관리를 위한 코어 위치 알고리즘과 데이터 패킷 전달과 제어 패킷 전달에 대한 포워딩 알고리즘이 필요하다. 이에 본 논문에서는 CCN에서 그룹 통신을 위한 코어 기반 멀티캐스트 라우팅 알고리즘을 제안하기 위해 트리 구성 및 관리를 위한 코어 위치 알고리즘은 IP 멀티캐스트 알고리즘인 CBT을 사용한다. 또한 데이터 패킷 전달과 제어 패킷 전달에 대한 포워딩 알고리즘은 CCN 라우팅 알고리즘인 OSPFN을 수정하여 사용한다.

공유 트리를 사용하는 멀티캐스트에서 코어를 선정하기 위해서는 멀티캐스트 수신자의 위치, 코어를 통해 흐르는 트래픽의 양, 멀티캐스트 송신자의 위치, 트래픽 지연 및 대역폭 사용량 등을 고려해야한다. 본 논문에서는 특정 영역에 사용자가 밀집된 지역에서 효율적인 그룹 통신을 위해 그림 20과 같은 코어 중심의 트리 구조를 갖는 토폴로지를 선정한다. 여기서 멀티캐스트의 송신자의 위치는 서버이다. 하지만 CCN은 네트워크 내 캐시를 이용하므로 멀티캐스트 송신자의 위치는 코어 라우터가

될 수도 있다. 또한 이 구조는 여러 멀티캐스트 그룹에 대한 트래픽이 서버로 집중되는 현상을 CCN 코어 라우터가 분산시켜준다. 이를 통해 네트워크의 데이터 전송 지연과 네트워크 대역폭 사용량을 줄일 수 있다.

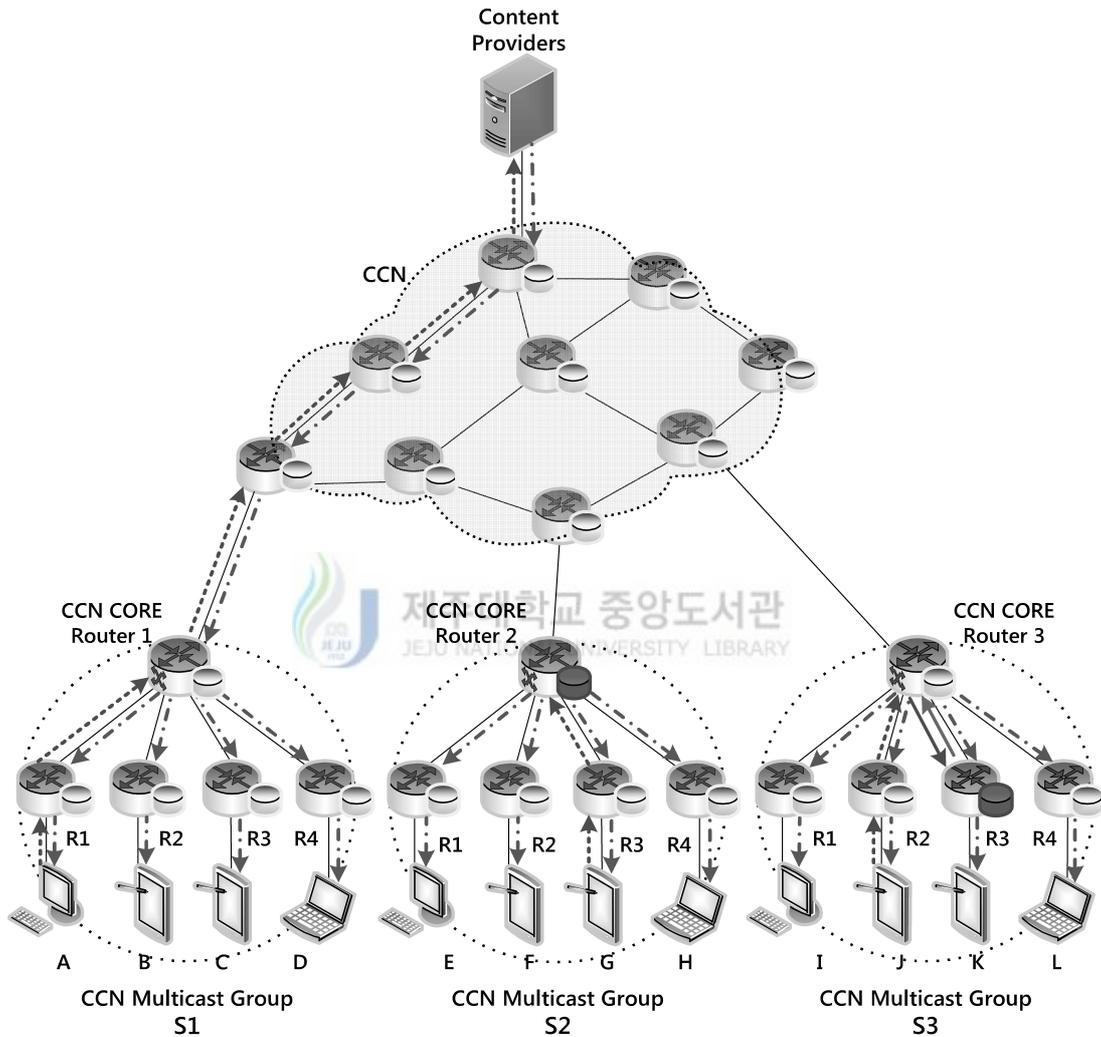


Fig. 20. An example of data forwarding in CMR-OSPFN

그림 20에서 CCN 멀티캐스트 그룹인 S1의 영역에서 동작 과정을 살펴보면, 클라이언트 A가 CCN 코어 라우터로 멀티캐스트 콘텐츠를 요청한다. 이때 CCN 코어 라우터 1의 CS에 데이터 패킷이 없다면 CCN 코어 라우터 1은 콘텐츠 제공자로 데이터 요청 패킷을 전송한다. CCN 코어 라우터 1이 콘텐츠 제공자로부터 데이터 패킷

를 수신하면 S1의 CCN 멀티캐스트 그룹 내의 CCN 라우터로 데이터 패킷을 멀티캐스트한다. 이 과정은 CBT의 멀티캐스트 데이터 전달 과정과 동일하다.

그림 20에서 S2 영역은 CCN 코어 라우터 2에 콘텐츠 정보가 있는 경우이다. 클라이언트 G가 콘텐츠를 요청하면 이를 수신한 CCN 코어 라우터 2는 자신의 CS를 검색한다. CCN 코어 라우터 2는 콘텐츠 제공자로 콘텐츠를 요청하지 않고 CS에 저장된 요청한 데이터 패킷을 S2의 멀티캐스트 그룹에게 데이터 패킷을 멀티캐스트한다.

그림 20의 CCN 멀티캐스트 그룹 S3의 영역을 살펴보면 코어 라우터에는 콘텐츠가 없으나 S3의 멀티캐스트 그룹 내의 중간 라우터인 CCN 라우터 R3에 콘텐츠가 있는 경우이다. 클라이언트 J가 멀티캐스트 데이터 요청 패킷을 CCN 코어 라우터 3에 요청한다. 이를 수신한 CCN 코어 라우터 3은 콘텐츠를 찾기 위해 FIB 테이블을 이용하여 콘텐츠 제공자보다 가까운 곳에 위치한 R3로 콘텐츠를 요청한다. CCN 코어 라우터 3은 R3로부터 제공받은 데이터 패킷을 S3의 멀티캐스트 그룹 내 CCN 라우터들에게 멀티캐스트한다.



## 1) OSPFN 라우팅 알고리즘

OSPFN은 CCN 내의 이름을 기반으로 한 라우팅을 제공하기 위해, 이름 접두사를 이용해 라우팅 경로를 설정한다. 이름 접두사를 네트워크의 라우터들에게 플로딩하기 위한 IP 라우팅 알고리즘 OSPF를 확장한 알고리즘이다.

OSPF는 자율시스템 내에서 링크 상태 라우팅 알고리즘이다. 네트워크 내의 각 라우터는 LSDB에 네트워크의 링크 상태 정보를 수집한다. LSDB는 링크 상태 광고(LSA, link state advertisement) 메시지를 네트워크로 퍼블리싱(publish)하여 업데이트한다. 네트워크 내의 모든 라우터들은 LSDB에 같은 복사본을 가지며, 각 라우터는 LSDB에 대한 네트워크 토폴로지를 구축하고, 라우팅 경로 계산을 위한 최단 경로 우선(shortest path first) 알고리즘을 운영한다. OSPF는 네트워크 토폴로지가 변화할 때 마다 LSA 메시지를 네트워크로 퍼블리싱하는 방법으로 네트워크 내 모든 라우터에게 플로딩(flooding)을 사용한다. OSPF는 네트워크 토폴로지의 변화에 따라

라우팅 테이블을 다시 계산해야 하지만 링크 상태 정보의 잦은 플로딩으로 인해 라우팅 경로에 대한 반복 루프가 발생하지 않는다.

CCN 라우팅 알고리즘인 OSPFN은 IP 라우팅 알고리즘과 두 가지 다른 차이점을 갖는다. 하나는 콘텐츠 이름을 이용한 라우팅으로 콘텐츠 제공자는 주소가 아닌 콘텐츠를 찾을 수 있는 정보만을 담고 있는 이름 접두사를 이용한다는 것이다. 두 번째는 다중 경로 설정으로 각 이름 접두사를 위한 여러 개의 경로를 제공한다는 것이다. 이름 접두사를 네트워크 내에 있는 라우터들에게 플로딩하기 위해 OSPFN은 링크 상태 광고 메시지로 OLSA(opaque LSA)을 사용한다. 네트워크 내의 OLSA의 플로딩은 OLSA 헤더의 Opaque 타입 필드에 의해 설정되며 플로딩 범위를 제한할 수 있다.



Fig. 21. Relationship between CCND, OSPFN, and OSPFD

OSPFN은 CCND<sup>[36]</sup> 내에 설치되고 이름 접두사를 이용해 라우팅 경로를 생성하여, 요청 패킷과 데이터 패킷의 전달을 조절한다. 그림 21과 같이 각 CCN 라우터는 CCND, OSPFN 그리고 OSPFD로 구성된다. OSPFN은 OLSA를 만들고 라우터가 보유한 콘텐츠의 이름 접두사를 전체 네트워크에 플로딩하여, OLSA에 설정된 플로딩 영역 내에 있는 라우터의 OSPFD에 OLSA를 전달한다. 라우터의 OSPFD가 OLSA를 수신하면 자신의 OSPFN에게 OLSA를 전달한다. CCN에서도 각 라우터들은 여전히 기존의 LSA를 전달하며 이를 이용해 네트워크의 최단 경로 트리를 형성한다. 그러므로 OSPFD는 각 라우터에 대한 다음 홉 정보를 가지고 있다. 이에 OSPFN은 OLSA의 다음 홉 정보를 얻기 위해 OSPFD에 쿼리를 전송한다. OSPFD는 쿼리에 대한 라우팅 정보를 OSPFN에게 전달한다. OLSA의 정보와 다음 홉에 대한 라우팅 정보를 수신한 OSPFN은 CCND의 FIB 안에 이름 접두사와 다음 홉 정보를 저장한다.

0	8	16	24	31				
LS Age		Options		LS Type				
Opaque Type	Opaque ID							
Advertising Router								
LS Sequence Number								
LS Checksum			Length					
Opaque Information (variable size)								
<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">Name LSA Opaque Information</td> </tr> <tr> <td style="text-align: center;">Size in Bytes of Name Prefix(32bits)</td> </tr> <tr> <td style="text-align: center;">Name Type(8 bits)</td> </tr> <tr> <td style="text-align: center;">Name Prefix(Variable size)</td> </tr> </table>					Name LSA Opaque Information	Size in Bytes of Name Prefix(32bits)	Name Type(8 bits)	Name Prefix(Variable size)
Name LSA Opaque Information								
Size in Bytes of Name Prefix(32bits)								
Name Type(8 bits)								
Name Prefix(Variable size)								

Fig. 22. OLSA message format

OLSA 메시지는 하나의 이름 접두사를 포함한다. 그림 22는 OLSA 메시지 형태를 나타낸 것이다. OLSA 메시지 필드들의 대부분은 라우터 내의 OSPFD에 의해 할당되어진다. OLSA는 네트워크 플로딩 영역을 위해 세 가지 범위를 가진다. 플로딩 영역 범위에 대해 LS 타입 필드는 10으로 사용한다. 이것은 로컬 영역 내에 OLSA를 플로딩할 것이라는 것을 의미한다. Opaque ID는 이름 접두사를 식별하기 위해 사용자에게 할당되어진 고유의 값이다. Opaque 정보 필드는 실제 이름 접두사와 마찬가지로 이름 접두사의 크기에 대해 32 bit 필드를 할당하고 이름 접두사 타입에 대해 8 bit 필드를 할당한다. 이름 접두사에 대한 필드의 크기는 이름 접두사의 길이에 따라 달라질 수 있다.

CCN에서 콘텐츠 제공자는 콘텐츠를 네트워크 내 라우터에게 알리기 위해 콘텐츠 이름을 명명한 후 이 콘텐츠 이름에 해당하는 콘텐츠 이름 접두사를 라우터에게 전송한다. 이 과정 후 CCN 라우터는 OSPFN 라우팅 알고리즘을 이용하여 이름 접두사를 네트워크 내로 플로딩하여 라우팅 테이블을 형성한다. CCN을 이용한 그룹 통신을 위해 제안하는 CMR-OSPFN 알고리즘의 제어 패킷 전달에 대한 포워딩 알고리즘

은 OSPFN을 수정하여 구현하고자 한다. 제어 패킷 전달은 그룹 가입, 탈퇴, 유지 관리에 관련된 패킷을 네트워크로 포워딩하여 멀티캐스트 라우팅 테이블을 형성하고 관리하는 것이다. 제안하는 CMR-OSPFN 알고리즘은 멀티캐스트 그룹 가입과 탈퇴는 CCN에서 콘텐츠 제공자가 최초 생성된 콘텐츠나 삭제된 콘텐츠를 네트워크로 이름 접두사를 이용해 알리는 것과 같이 멀티캐스트 이름 접두사를 이용해 네트워크로 알릴 수 있다. CCN 라우터는 멀티캐스트 이름 접두사 메시지를 네트워크로 플로딩하여 CCN 코어 라우터로 전달하여 멀티캐스트 라우팅 테이블을 형성하는 방식을 사용한다. 또한 라우팅 테이블을 유지 관리하기 위해서는 OSPFN의 라우팅 테이블의 정보가 변할 때 마다 네트워크로 플로딩한다.

## 2) 멀티캐스트 이름 접두사

CMR-OSPFN은 그룹 관리 기능을 구현하기 위해, 그룹 관리 메시지인 멀티캐스트 이름 접두사를 포함한 메시지를 추가한다. 각 라우터들은 멀티캐스트 이름 접두사를 주기적으로 네트워크에 플로딩하여 멀티캐스트 그룹 경로를 설정한다.

OSPFN의 이름 접두사는 콘텐츠 제공자의 정보를 포함하며 그 구조는 그림 23과 같다. 이름 접두사는 데이터 이름 전체가 아닌 콘텐츠를 찾을 수 있는 정보만을 담고 있다.

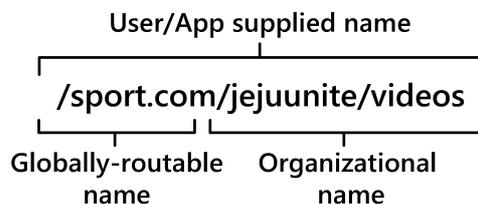


Fig. 23. Name prefix structure

CCN의 이름 접두사는 콘텐츠 제공자로부터 시작되는 반면 멀티캐스트 라우팅 경로의 형성을 위한 멀티캐스트 이름 접두사는 콘텐츠 제공자가 아닌 그룹에 참여

를 원하는 클라이언트로부터 시작된다. CCN의 라우터는 이름 접두사의 Globally-routable name을 이용해 콘텐츠 제공자를 알 수 있으며 콘텐츠 제공자에 도착했을 때 해당하는 콘텐츠를 찾기 위해 그림 24와 같은 이름 트리를 탐색하여 콘텐츠를 검색한다. CCN에서 멀티캐스트 라우팅을 위해서는 어떤 콘텐츠 제공자의 멀티캐스트 그룹에 참여할 것인지에 대한 정보를 포함한 그룹 가입 메시지를 클라이언트가 CCN의 코어 라우터에게 전달하여야 한다.



Fig. 24. Name tree traversal

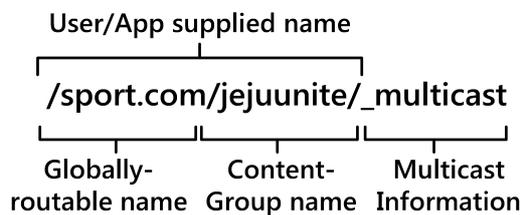


Fig. 25 Multicast name prefix structure

그림 25는 Globally-routable name, Content-Group name, Multicast

Information으로 구성되고 멀티캐스트 그룹 가입 메시지의 이름 접두사 마지막에 \_multicast가 있으면 그룹에 참여를 원하는 클라이언트로부터 전송된 것이고 \_multicast가 없는 경우라면 콘텐츠 제공자로부터 전송된 것이다.

### 3) 멀티캐스트 라우팅 경로 설정

#### (1) 다중 경로 설정

다중 라우팅 경로를 형성하기 위해 OSPFN은 최단 경로 계산을 수행하지 않고, 이름 접두사를 전송한 라우터에 대한 다음 홉의 정보를 OSPFD에 조회한다. OSPFN은 조회 결과를 수신했을 때, 이름 접두사와 다음 홉이 포함된 FIB 항목을 CCND에게 추가한다. 기존 IP 기반의 OSPF 프로토콜은 각 목적지를 위한 오직 하나의 다음 홉을 제공하기 때문에 동일 비용의 최단경로를 제외하고는 기본적으로 OSPFN은 하나의 이름 접두사에 대한 하나의 경로만을 가진다. 그러나 CCN의 고유의 특징 중 하나는 포워딩 전략으로 이름에 기반한 데이터의 조각을 검색하기 위해 여러 경로를 탐색할 수 있다는 것이다.

OSPFN은 FIB 내에 동일한 이름 접두사를 갖는 최선의 경로 라우터들의 다음 홉에 대한 여러 개의 항목을 저장하는 것을 허용한다. 각 라우터의 인터페이스는 경로 비용이 낮은 경로를 제일 먼저 연결할 것이다. 이 기능을 '구성 다중 경로 라우팅(configured multipath routing)'이라 한다.

CCND에 대한 초기 경로 검색에 대해 기술하면 CCND의 포워딩 전략은 데이터를 가장 빠르게 검색하는 경로를 최선의 경로로 선택하는 방법을 사용한다. 다중 경로 구성이 사용될 때, OSPFN은 각 이름 접두사에 대해 FIB 항목의 리스트를 발생한다. 가장 선호되는 다음 홉은 출발지로의 다중 경로, 경로 비용과 연관된 랭킹에 의한 다중 경로, 그리고 내림차순 우선순위에 의한 다중 경로 등에 의해 선택될 것이다. OSPFN은 역순으로 FIB에 삽입되어 CCND는 우선순위가 가장 높은 FIB 항목이 마지막에 삽입된다.

(2) 멀티캐스트 라우팅 경로 설정

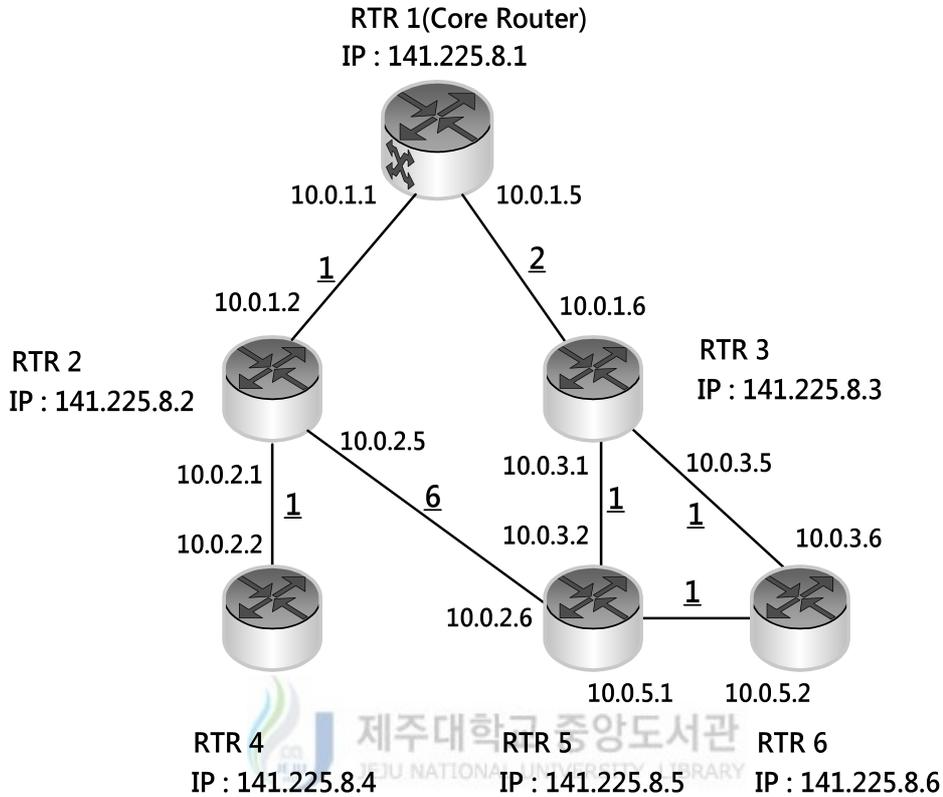


Fig. 26. Network topology for route discovery

CMR-OSPFN은 멀티캐스트 라우팅 경로 설정에 대한 설명하기 위해, 그림 26의 네트워크 토폴로지를 사용한다. 네트워크는 1개의 CCN 코어라우터와 5개의 CCN 라우터로 구성되며 각 라우터의 터널 주소, IP, 링크 비용은 그림 26과 같다. 각 CCN 라우터는 이름 접두사에 직접 액세스하며 라우터는 자신이 보유한 콘텐츠의 각 이름 접두사와 멀티캐스트 이름 접두사는 표 3과 같다.

Table 3. Name prefixes originated by each router

Router	Name Prefixes
RTR1	/sport.com/jejuunite/videos /sport.com/jejuunite/images /sport.com/jejuunite/musics
RTR2	/sport.com/fcseoul/videos /sport.com/fcseoul/images /sport.com/jejuunite/_multicast
RTR3	/sport.com/fcseongnam/videos /sport.com/fcseongnam/musics /sport.com/jejuunite/_multicast
RTR4	/sport.com/hyundai/videos /sport.com/hyundai/musics /sport.com/jejuunite/_multicast
RTR5	/sport.com/bluewings/videos /sport.com/bluewings/images /sport.com/jejuunite/_multicast
RTR6	/sport.com/dragons/videos /sport.com/dragons/images /sport.com/jejuunite/musics /sport.com/jejuunite/_multicast

각 라우터는 처음에 ccname 명령을 사용해 광고하기를 원하는 이름 접두사와 멀티캐스트 이름 접두사를 구성한다. 또한 라우터는 multipath-order 명령을 사용해 다른 경로를 명시할 수 있다. RTR1, RTR3 그리고 RTR6에 대한 구성 파일은 표 4와 같다. 각 구성 파일은 라우터가 발생한 이름 접두사, 멀티캐스트 이름 접두사로 구성되고 RTR1, RTR3의 경우 다중 경로 구성을 설정한다.

Table 4. Router configuration files

RTR1 Configuration File
<pre>ccnname /sport.com/jejuunite/videos 1 ccnname /sport.com/jejuunite/images 2 ccnname /sport.com/jejuunite/musics 3 multipath-order 10.0.1.2 10 multipath-order 10.0.1.6 20</pre>
RTR3 Configuration File
<pre>ccnname /sport.com/fcseoul/videos 1 ccnname /sport.com/fcseoul/images 2 ccnname /sport.com/jejuunite/_multicast 0 multipath-order 10.0.1.5 10 multipath-order 10.0.3.6 15 multipath-order 10.0.3.2 20</pre>
RTR6 Configuration File
<pre>ccnname /sport.com/dragons/videos 1 ccnname /sport.com/dragons/images 2 ccnname /sport.com/jejuunite/musics 3 ccnname /sport.com/jejuunite/_multicast 0</pre>

각 라우터는 자신의 구성 파일을 네트워크에 플로딩하기 위해 이름 접두사와 멀티캐스트 이름 접두사에 대한 OLSA를 생성한다. OSPFD에 의해 제공되는 API를 통해 다른 라우터로부터 OLSA의 도착/삭제/업데이트를 조정하며 OSPFD는 자신의 OSPFN에게 OLSA를 삽입할 것이다. 각 라우터에는 네트워크 내에 플로딩된 모든 콘텐츠의 이름 접두사와 멀티캐스트 이름 접두사를 포함한 이름 접두사 테이블이 구축되어 있다. RTR3에 대한 OLSA는 그림 30과 같다. 표 4의 RTR3의 라우터의 구성 파일 내의 플로딩을 원하는 2개의 이름 접두사와 1개의 멀티캐스트 이름 접두사를 네트워크로 플로딩하기 위해 OLSA 메시지에 이름 접두사와 멀티캐스트 이름 접두사의 정보를 삽입한다.

RTR3 Opaque Name LSA	
LS Type	10
Opaque Type	236
Advertising Router	141.255.8.3
Opaque Information	
20 (Size in byte of prefix name)	
0 (0/1 0-indicates uri, 1-indicates ccnb)	
/sport.com/fcseoul/videos	

RTR3 Opaque Name LSA	
LS Type	10
Opaque Type	236
Advertising Router	141.255.8.3
Opaque Information	
23 (Size in byte of prefix name)	
0 (0/1 0-indicates uri, 1-indicates ccnb)	
/sport.com/fcseoul/images	

RTR3 Opaque Name LSA	
LS Type	10
Opaque Type	235
Advertising Router	141.255.8.3
Opaque Information	
25 (Size in byte of prefix name)	
0 (0/1 0-indicates uri, 1-indicates ccnb)	
/sport.com/jejuunite/_multicast	

Fig. 27. OLSA for RTR3

코어 라우터인 RTR1의 이름 접두사 테이블은 표 5와 같으며 각 이름 접두사와 멀티캐스트 이름 접두사에 대해 발생한 라우터가 그림 27의 네트워크 토폴로지에 의해 일치하는 것을 볼 수 있다. 각 CCN 라우터는 표5의 RTR1의 이름 접두사 테이블에서 이름 접두사에 대한 CCN 경로 설정 테이블인 FIB 필드를 구성하기 위

해, CMR-OSPFN은 이름 접두사 테이블의 이름 접두사 레코드의 값이 이름 접두사인지 멀티캐스트 이름 접두사인지를 구별한다.

Table 5. RTR1's Name prefix table

RTR1 Name Prefix Table	
Name Prefix	Advertising Router
/sport.com/fcseoul/videos	141.225.8.2 (RTR2)
/sport.com/fcseoul/images	141.225.8.2 (RTR2)
/sport.com/jejuunite/_multicast	141.225.8.2 (RTR2)
/sport.com/fcseoul/videos	141.225.8.3 (RTR3)
/sport.com/fcseoul/images	141.225.8.3 (RTR3)
/sport.com/jejuunite/_multicast	141.225.8.3 (RTR3)
/sport.com/hyundai/videos	141.225.8.4 (RTR4)
/sport.com/hyundai/musics	141.225.8.4 (RTR4)
/sport.com/jejuunite/_multicast	141.225.8.4 (RTR4)
/sport.com/bluwings/videos	141.225.8.5 (RTR5)
/sport.com/bluwings/images	141.225.8.5 (RTR5)
/sport.com/jejuunite/_multicast	141.225.8.5 (RTR5)
/sport.com/dragons/videos	141.225.8.6 (RTR6)
/sport.com/dragons/images	141.225.8.6 (RTR6)
/sport.com/jejuunite/musics	141.225.8.6 (RTR6)
/sport.com/jejuunite/_multicast	141.225.8.6 (RTR6)

이름 접두사인 경우 RTR 1에 대한 FIB에 레코드 필드를 저장하는 과정을 예를 들어 살펴보면 이름 접두사 /sport.com/jejuunite/musics에 대한 콘텐츠를 가진 라우터를 표 5의 이름 접두사 테이블에서 찾는다. 그리고 OSPFD는 표 6과 같이 라우팅 테이블로부터 콘텐츠를 가진 라우터로의 다음 홉 정보를 검색한다. FIB 레코드의 필드는 표 7과 같이 이름 접두사와 콘텐츠를 가진 라우터로 향하는 다음 홉으로 구성된다. FIB 항목을 구성한 다음에 CCND의 FIB안에 삽입한다.

Table 6. RTR1's routing table

RTR1 OSPF Routing Table	
Destination	Advertising Router
RTR2	10.0.1.2 (RTR2)
RTR3	10.0.1.6 (RTR3)
RTR4	10.0.1.2 (RTR2)
RTR5	10.0.1.6 (RTR3)
RTR6	10.0.1.6 (RTR3)

Table 7. RTR1's FIB table

RTR3 FIB		
Name Prefix	Next Hop	Multicast Group
/sport.com/fcseoul/videos	10.0.1.6 (RTR3)	
/sport.com/fcseoul/videos	10.0.1.2 (RTR2)	
...	...	
/sport.com/fcseoul/images	10.0.1.6 (RTR3)	
/sport.com/fcseoul/images	10.0.1.2 (RTR2)	
...	...	
/sport.com/jejuunite/_multicast	10.0.1.6 (RTR3)	jejuunite
/sport.com/jejuunite/_multicast	10.0.1.2 (RTR2)	jejuunite

이름 접두사 테이블의 이름 접두사가 멀티캐스트 이름 접두사인 경우 OSPF는 멀티캐스트 이름 접두사의 마지막 / 다음의 \_multicast를 삭제하고 RTR 1에 대한 FIB 테이블의 이름 접두사 레코드의 필드에 저장한다. 또한 RTR 1의 멀티캐스트 그룹 레코드에 대한 필드는 멀티캐스트 이름 접두사에서 콘텐츠 그룹 이름과 같다. 다음 흡의 레코드 필드는 표 6을 이용해 멀티캐스트 이름 접두사를 발생한 라우터로 가는 다음 흡의 정보를 FIB 테이블에 담는다.

#### 4) 코어 라우터의 우선순위

##### (1) 다중 경로 우선순위

이름 접두사와 멀티캐스트 이름 접두사가 여러 라우터에서 플로딩되었다면, FIB 항목은 각 발생 라우터로 향하는 다음 홉에 대한 다중 경로를 구성한다. 다중 경로에 대한 우선순위는 경로 비용을 낮추기 위해 제공되어 진다. FIB 테이블에 다중 경로를 구성하는 경우 경로 비용이 낮은 이름 접두사의 라우터에 대한 다음 홉을 따르며 우선순위 증가에 의한 이름 접두사에 대해 구성된 각 다음 홉에 대한 값을 FIB 테이블에 삽입하여 구성한다.

Table 8. Multipath ording for the /sport.com/jejuunite/musics

RTR3 FIB		
Name Prefix	Next Hop	Multicast Group
/sport.com/jejuunite/musics	10.0.5.1 (RTR5)	
/sport.com/jejuunite/musics	10.0.1.5 (RTR1)	
/sport.com/jejuunite/musics	10.0.3.6 (RTR6)	

CCND는 처음에 삽입된 FIB 필드를 마지막에 사용하기 때문에, OSPFN은 다른 다음 홉 보다 높은 우선순위를 갖는 라우터의 다음 홉이라는 것을 알 수 있다. 표 8과 같이 RTR3는 FIB로 다음 홉은 낮은 경로 비용에 높은 우선순위를 부여한다. /sport.com/jejuunite/musics에 대한 RTR3의 다음 홉을 찾기 위해, 첫 번째 이름 접두사의 발생 라우터(즉, RTR1, 6)은 OSPF 라우팅 테이블을 검색한다. 각각 10.0.1.5(RTR1)와 10.0.3.6 (RTR6)의 라우터에 연관된 다음 홉의 경로 비용을 산출한다. 경로 비용을 찾으며, 낮은 값이 우선순위를 부여 받는다. RTR1이 RTR3로부터 2의 경로 비용을 가지는 반면, RTR6은 RTR3으로부터 1의 경로 비용을 가진다. 그러므로 RTR6은 RTR1에 최선의 경로인 다음 홉이 된다. 그러므로 최선의 경로는 다음 홉에 대한 비용이 2인 RTR6이 된다. 또한 RTR3은 자신의 다중 경로 구성 때문에 백업 경로를 가진다.

멀티캐스트 데이터 패킷은 패킷을 요청한 클라이언트에게만 전달되는 것이 아니라 멀티캐스트 그룹에 해당하는 모든 라우터를 통해 데이터를 전달하기 때문에 멀티캐스트 이름 접두사는 우선순위를 요구하지 않는다. 멀티캐스트 데이터 패킷을 수신한 라우터들은 자신의 FIB의 이름 접두사와 멀티캐스트 그룹 필드와 일치하는 모든 다음 홉으로 데이터 패킷을 전달한다.

## (2) 코어 라우터의 우선순위

제안하는 CMR-OSPFN 알고리즘에서 코어 라우터의 우선순위가 필요하다. 코어 라우터의 우선순위가 설정되면, 클라이언트의 멀티캐스트 요청 패킷을 CCN 라우터가 수신하였을 때, FIB를 이용해 다중 경로 우선순위를 무시하고 코어 라우터의 우선순위에 의해 요청 패킷을 코어 라우터로 전달한다. 하지만 코어 라우터의 우선순위를 설정하지 않은 경우, CCN 라우터는 멀티캐스트 요청 패킷을 코어 라우터로 전달하지 않고 우선순위가 높은 CCN 라우터로 요청 패킷을 전달하게 될 것이다. 예를 들어, /sport.com/jejuunite/musics/B.mpg/v3/s0의 멀티캐스트 요청 패킷을 수신한 RTR3은 수신한 요청 패킷을 다음 홉으로 전달하기 위해 표 7의 FIB 테이블을 검색한다. 이때 FIB 테이블에 코어 라우터의 우선순위가 설정되어 있지 않다면, 다중 경로의 우선순위에 따라 코어 라우터인 RTR1(10.0.1.5)가 아닌 RTR6(10.0.3.6)로 요청 패킷을 전달하게 된다. 이를 막기 위해 코어 라우터에 의해 발생된 이름 접두사에 대한 FIB 필드에 우선순위를 부여해야 한다.

코어 라우터에게 우선순위를 부여하기 위해 콘텐츠 제공자에게서 부여받은 이름 접두사의 끝에 코어 라우터 정보 /\_CORE를 그림 28과 같이 추가한다. 코어 라우터는 이름 접두사를 네트워크 내의 라우터들에게 플로딩한다. CCN 라우터들은 코어 라우터가 전달한 이름 접두사를 수신하면 수신한 이름 접두사에 코어 라우터의 정보가 있는지를 확인한다. 코어 정보가 있을 경우 자신의 FIB에 이름 접두사의 코어 정보를 삭제한 후 FIB 테이블의 이름 접두사 필드에 이름 접두사를 저장한다. 또한 멀티캐스트 그룹 필드에 코어 라우터의 정보를 표 9와 같이 저장한다.

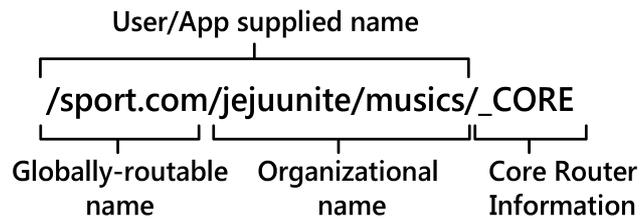


Fig. 28. Name prefix structure of the core router

Table 9. Modified FIB entry at RTR3

RTR3 FIB		
Name Prefix	Next Hop	Multicast Group
/sport.com/jejuunite/musics	10.0.5.1 (RTR5)	
/sport.com/jejuunite/musics	10.0.1.5 (RTR1)	Core
/sport.com/jejuunite/musics	10.0.3.6 (RTR6)	
...	...	
/sport.com/fcseoul/videos	10.0.3.6 (RTR6)	
/sport.com/fcseoul/videos	10.0.5.1 (RTR5)	
/sport.com/fcseoul/videos	10.0.1.5 (RTR1)	
...	...	
/sport.com/jejuunite	10.0.1.5 (RTR1)	jejuunite
/sport.com/jejuunite	10.0.5.1 (RTR5)	jejuunite
/sport.com/jejuunite	10.0.3.6 (RTR6)	jejuunite

### 3. CMR-OSPFN 알고리즘에 의한 데이터 전달 방식

#### 1) CMR-OSPFN 알고리즘 사용을 위한 메시지 패킷 구조

기존 CCN 라우터는 콘텐츠를 요청한 사용자의 역경로로만 데이터 패킷을 전달한다. 하지만 제안된 CMR-OSPFN 알고리즘에서는 하나의 요청으로 전체 그룹멤버에게 데이터 패킷을 전달해야 하므로, CCN 메시지 패킷인 요청 패킷과 데이터 패킷을 수정해야한다. 멀티캐스트 패킷인지를 구별하기 위하여 요청 패킷과 데이터 패킷을 포함한 기본 CCN 패킷에 TOS(type of service)를 그림 39와 같이 확장하였다. TOS 필드는 16bit 길이이고 멀티캐스트 패킷인지를 나타내기 위해 사용된다. 멀티캐스트 패킷인 경우 TOS의 0번째 비트에 1을 나타내고 멀티캐스트 패킷이 아닌 경우 0번째 비트를 0으로 나타낸다. 나머지 bit는 멀티캐스트 그룹명을 저장한다.

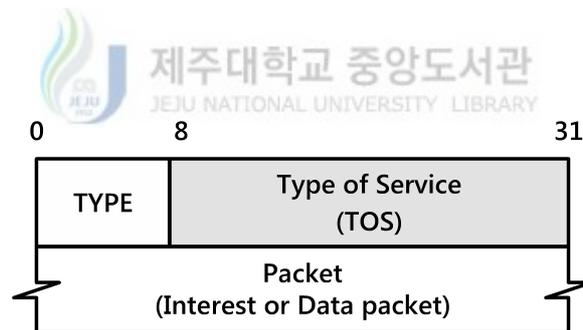


Fig. 29. Extentsion of CCN Packets

#### 2) 코어 라우터와 라우터 간 동작 과정

CMR-OSPFM 알고리즘은 라우터는 CCN 코어 라우터와 CCN 라우터로 구성된다. 우선 CCN 코어 라우터의 동작 과정을 살펴보면 CCN 코어 라우터에 그림 30과 같이 CCN 라우터로부터 요청 패킷이 들어오면 TOS의 0번째 비트를 확인하여 멀티캐스트 패킷인지를 판단한다. 멀티캐스트 패킷이 아니면 기존의 CCN과 동일

한 방법으로 동작한다. 반면 멀티캐스트 패킷인 경우 TOS의 0번째 비트를 제외한 나머지 비트에 저장된 멀티캐스트 그룹 정보를 이용하여, FIB에 그룹 정보에 해당하는 모든 CCN 라우터의 페이스 정보를 찾는다. 코어 라우터는 표 10과 같이 자신의 PIT에 이름, 멀티캐스트 그룹, 페이스 레코드에 해당하는 필드를 저장한다.

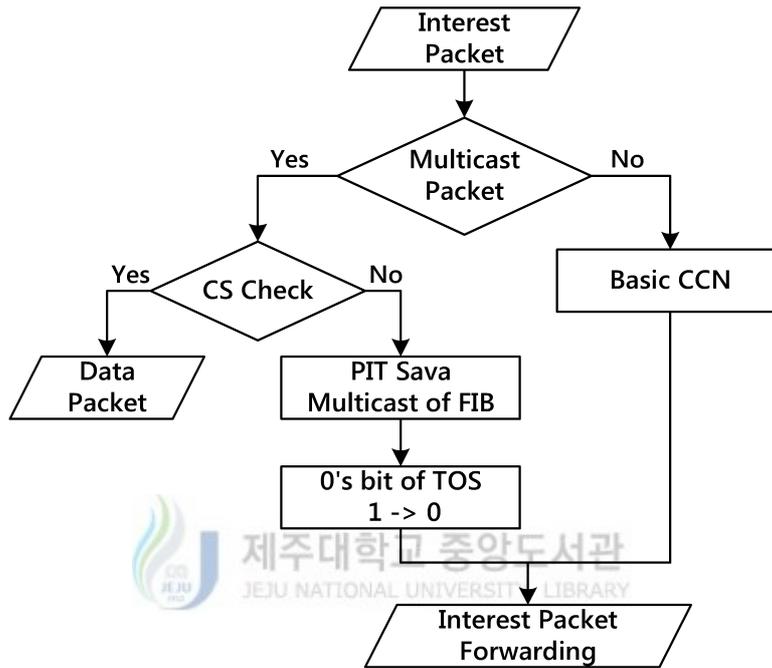


Fig. 30. Flow chart of the Interest Packet to the CCN Core Router

Table 10. Modified PIT Entry

Name	Multicast Group	face
/sport.com/jejuunite/videos/B.mpg/v3/s0	jejuunite	0,2
/sport.com/FC_Seoul/videos/A.mpg/v3/s0	FC_Seoul	5,1

코어 라우터는 요청 패킷이 요청한 데이터 패킷이 자신의 CS에 있는지 확인한 후 자신의 CS에 있으면 그림 29의 데이터 패킷에 TOS 0번째 비트에는 1을 담고 나머지 비트에 멀티캐스트 그룹 정보를 저장한다. CCN 코어 라우터는 PIT에 저장

된 페이스 정보를 이용하여 그룹 내 CCN 라우터들에게 데이터 패킷을 멀티캐스트한다. 만약 자신의 CS에 데이터 패킷이 존재 하지 않는다면 TOS의 0번째 비트를 1에서 0으로 변경한 후 FIB의 라우팅 정보를 이용해 다음 홉으로 전달한다. 0번째 비트를 1에서 0으로 변경하는 이유는 FIB 테이블에 설정된 코어 라우터에 대해 우선순위가 높은 경로에 영향을 받지 않고 가장 빠른 속도로 데이터 패킷을 찾기 위해서이다.

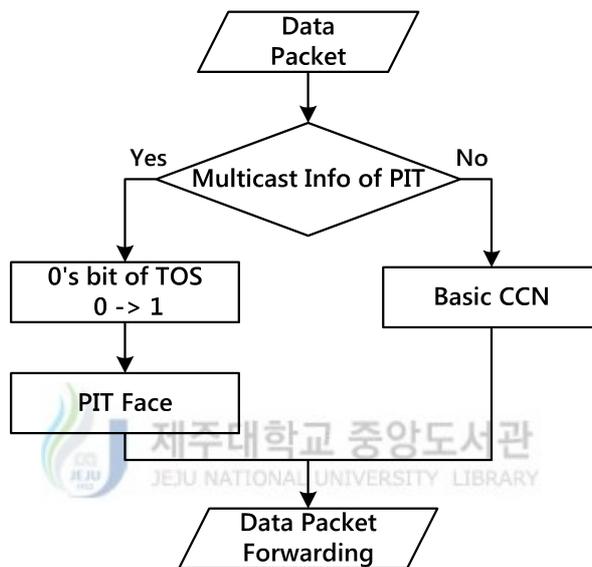


Fig. 31. Flow chart of the Data Packet to the CCN Core Router

CCN 코어 라우터에 그림31과 같이 데이터 패킷이 들어오면 자신의 PIT를 확인하여 데이터 패킷의 이름 접두사와 동일한 필드를 찾는다. 또한 동일한 필드의 멀티캐스트 그룹 레코드에 해당하는 필드의 값을 검사한다. 멀티캐스트 그룹 필드에 값이 존재한다면 데이터 패킷의 TOS의 0번째 비트를 0에서 1로 변경해 데이터 패킷을 CCN 라우터에게 전달한다. 0번째 비트의 변경은 멀티캐스트 패킷이라는 것을 멀티캐스트 그룹에 알리기 위해서이다. TOS의 나머지 비트에는 PIT의 멀티캐스트 그룹 레코드의 값을 담아 표 10의 PIT의 다중 페이스로 데이터 패킷을 멀티캐스트한다.

### 3) 라우터 간 동작 과정

CCN 라우터의 기능을 살펴보면 CCN 라우터에 그림 32와 같이 요청 패킷이 들어오면 멀티캐스트 패킷인지를 확인하기 위해 TOS의 0번째 비트를 검사한다. 멀티캐스트 패킷이면 자신의 CS를 검사하지 않고 PIT에 데이터 패킷이 돌아갈 경로를 설정한다. 멀티캐스트 패킷인 경우 CCN 코어 라우터와 동일하게 FIB 멀티캐스트에 해당하는 이름 접두사와 멀티캐스트 그룹 정보에 해당하는 모든 페이스 정보를 PIT에 저장한다. CCN 라우터는 수신한 요청 패킷을 코어 라우터로 전달하기 위해 자신의 FIB 테이블을 검사한다. CCN 라우터는 FIB에서 이름 접두사만을 이용해 라우팅 경로를 찾는 것이 아니라 이름 접두사와 코어 정보를 모두 확인해 요청 패킷을 전송할 페이스 정보를 찾는다. 이는 CCN 라우터가 멀티캐스트 요청 패킷이 다중 경로 우선순위에 의해 코어 라우터가 아닌 다른 라우터로 전달되는 것을 방지하기 위해서이다.

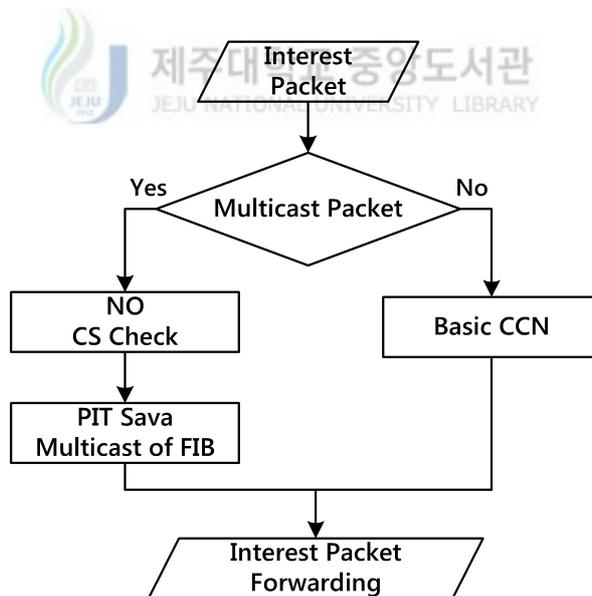


Fig. 32. Flow chart of the Interest Packet to the CCN Router

마지막으로 CCN 라우터에 그림 33과 같이 데이터 패킷이 들어오면 멀티캐스트

패킷인지를 확인하고 멀티캐스트 패킷인 경우 PIT에서 데이터 패킷의 이름 접두사와 동일한 필드를 찾는다. 동일한 필드가 있는 경우 페이스 레코드의 필드에 해당하는 페이스로 데이터 패킷을 내보낸다. 반면 자신이 요청한 데이터 패킷이 아닌 멀티캐스트 그룹에 대한 데이터 패킷이라면 PIT에 이름 접두사와 동일한 정보가 없을 것이다. 멀티캐스트 요청 패킷의 경로 설정과 같은 방법으로 멀티캐스트 데이터의 이름 접두사와 멀티캐스트 그룹 정보를 이용하여 자신의 FIB를 검색한다. CCN 라우터는 FIB 레코드의 이름 접두사와 멀티캐스트 그룹 정보가 일치하는 필드의 모든 페이스로 데이터 패킷을 전달한다.

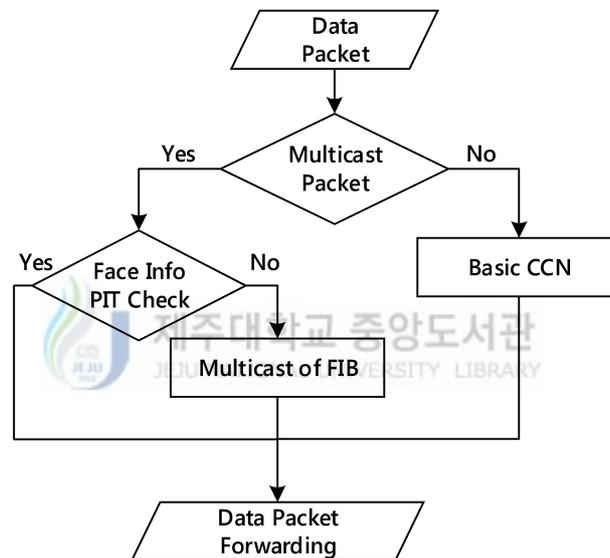


Fig. 33. Flow chart of the Data Packet to the CCN Router

#### 4. 시뮬레이션 결과 및 고찰

##### 1) 시뮬레이션 네트워크 토폴로지

CCN 환경에서 제안한 멀티캐스트 라우팅 프로토콜의 성능을 분석하기 위해 본

논문에서는 Network Simulator(ns2)에서 CCN 환경을 구성하였고 IP 기반의 멀티캐스트 라우팅 프로토콜인 CBT와 시뮬레이션 결과를 비교 분석한다. 표 11은 시뮬레이션에 사용된 파라미터로 시뮬레이션 네트워크의 각 라우터간의 링크 속도는 100Mbps로 가정하였으며 전송 선로의 전송 지연은 0.1ms로 하였다.

Table 11 Simulation Parameters

Parameters	Values
Link bandwidth	100Mbps
Link delay	0.1ms
Interest/Request Packet size	1Kbyte
Data Packet size	50Kbyte
Content size	10Mbyte

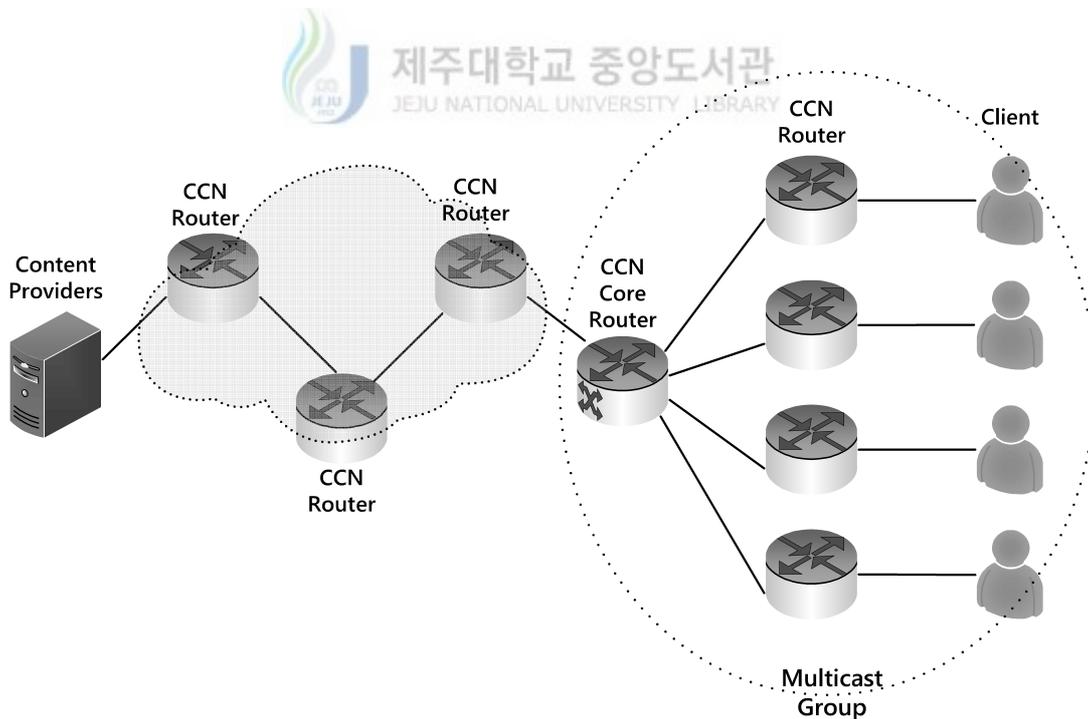


Fig. 34. Network topology

그림 34은 시뮬레이션 환경에 대한 네트워크 토폴로지로 콘텐츠 제공자, 멀티캐스트 그룹 그리고 콘텐츠 제공자와 멀티캐스트 그룹을 연결하는 CCN 라우터로 구성된다. 여기서 멀티캐스트 그룹은 그룹을 관리하는 CCN 코어 라우터, CCN 4개 그리고 각 CCN 라우터에 연결된 클라이언트 4개로 구성된다. 콘텐츠 제공자는 콘텐츠를 가지며, 클라이언트들은 콘텐츠 요청을 위해 요청 패킷을 발생시킨다. 또한 CBT의 라우터는 캐시를 가지지 않으며, CMR-OSPFN의 라우터는 캐시를 가진다. CMR-OSPFN의 라우터들은 콘텐츠 제공자로부터 전달받는 데이터 패킷의 데이터를 자신의 캐시에 저장한다.

시뮬레이션에서 사용되는 콘텐츠의 크기는 10Mbyte이다. 요청 패킷의 크기는 1Kbyte이고 데이터 패킷의 크기는 50Kbyte로 각각 고정한다. 또한 클라이언트들의 콘텐츠의 요청 시간과 순서는 랜덤하게 이루어지도록 하였다.

본 논문에서는 CMR-OSPFN와 CBT를 비교하기 위해 하나의 데이터 패킷을 요청할 때 하나의 데이터 요청 패킷을 전송하는 방식을 동일하게 사용한다. IP 기반의 경우 콘텐츠를 요청할 때마다 클라이언트로부터 콘텐츠 제공자에게 요청 패킷을 보내지만 제안한 프로토콜의 경우 콘텐츠 제공자가 아닌 콘텐츠를 보유한 코어 라우터로부터 가장 가까운 이웃 라우터에게 콘텐츠를 제공받는다. 이에 CMR-OSPFN와 CBT의 성능을 비교하기 위해 콘텐츠 제공자와 코어 라우터 사이의 홉 수 변화에 따른 콘텐츠 전달 시간 및 네트워크 대역폭 사용량을 측정한다. 또한 다양한 콘텐츠와 동일한 콘텐츠의 반복적인 요청에 따른 시뮬레이션 결과를 비교한다.

그림 35은 CBT의 데이터 요청 패킷과 멀티캐스트 데이터 패킷의 전달 과정을 나타낸 것으로 클라이언트가 콘텐츠를 서버로 요청하면 요청 패킷인 요청 패킷은 콘텐츠 제공자로 전달된다. 콘텐츠 제공자는 요청 패킷이 요청하는 멀티캐스트 데이터 패킷을 코어 라우터로 전달한다. 코어 라우터는 콘텐츠 제공자에서 수신한 데이터를 멀티캐스트 그룹에게 전달하여 데이터를 멀티캐스트 한다. CBT는 클라이언트들이 콘텐츠를 요청할 때마다 이 과정을 반복한다.

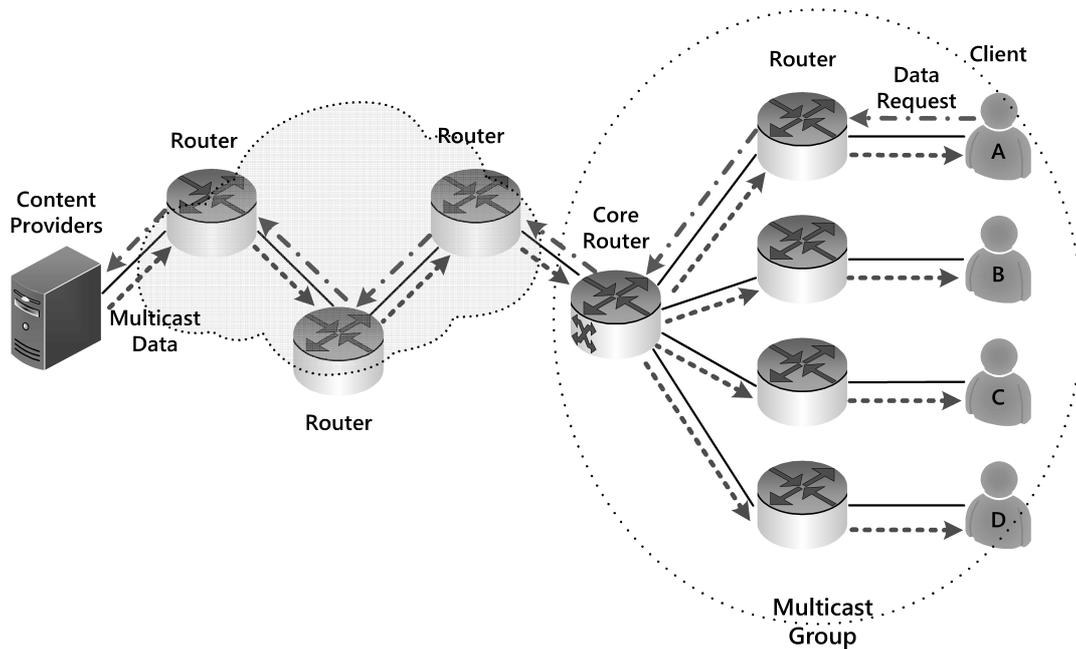


Fig. 35. Multicast data delivery process in CBT

그림 36과 그림 37은 CMR-OSPFN의 요청 패킷과 데이터 패킷의 전달 과정을 나타낸 것이다. 그림 36은 최초 요청되는 콘텐츠에 대한 CMR-OSPFN의 전달을 나타낸 것으로 클라이언트 A가 콘텐츠를 요청하면 요청 패킷은 CCN 코어 라우터로 전달된다. CCN 코어 라우터는 자신의 CS에 요청한 데이터 패킷이 없음을 확인하면 콘텐츠 제공자로 다시 요청 패킷을 보내 데이터 패킷을 요청한다. CCN 코어 라우터는 요청한 데이터 패킷을 수신하면 이를 멀티캐스트 그룹에게 전달한다. 이는 CBT와 동일한 포워딩 과정을 갖는다.

그림 37은 클라이언트 A가 요청한 동일한 콘텐츠를 클라이언트 B가 다시 요청한 경우이다. 이는 CCN 코어 라우터의 CS에 콘텐츠가 존재할 때의 멀티캐스트 포워딩 과정을 나타낸 것이다. CCN 코어 라우터는 클라이언트 B의 요청 패킷을 수신하면 이를 콘텐츠 제공자로 전송하지 않고 자신의 CS에 저장된 데이터 패킷을 바로 멀티캐스트 그룹에게 전달한다.

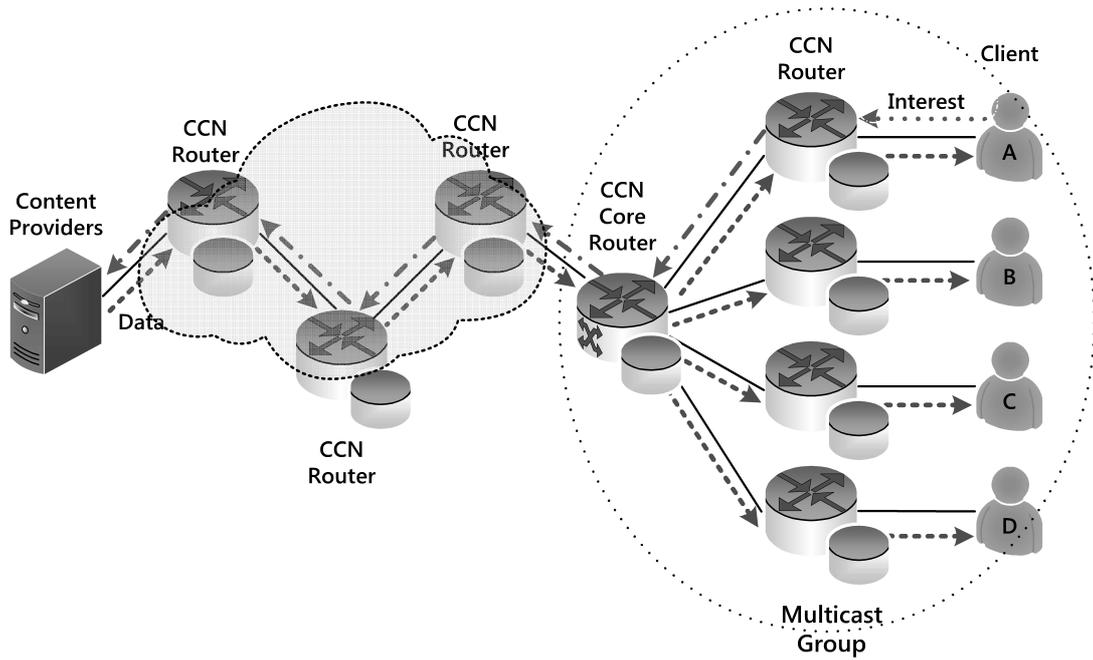


Fig. 36. Multicast data delivery process in CMR-OSPFN for the first request

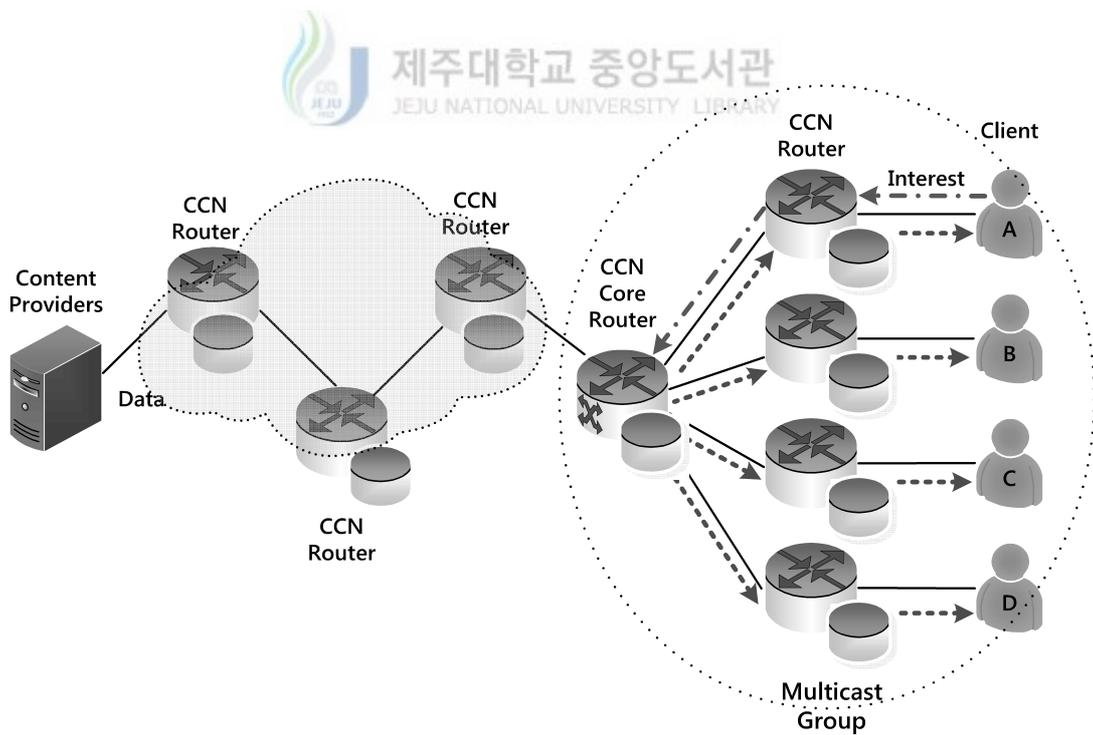


Fig. 37. Multicast data delivery process in CMR-OSPFN for the repeat request

식 (1)과 (2)는 각각 멀티캐스트 그룹 내 클라이언트들이 동일한 콘텐츠를 반복적으로 요청할 경우 평균 데이터 전달 시간을 나타낸 것이다. 식 (1)은 CBT의 평균 데이터 전달 시간  $T_{t\_CBT}$ 을 나타낸 것이며, 식 (2)는 CMR-OSPFN의 평균 데이터 전달 시간  $T_{t\_CMR}$ 을 나타낸 것이다.

$$T_{t\_CBT} = \frac{\sum_{j=1}^N [(T_{I_j} + T_{D_j}) \times H_{sc}]}{N} \quad (1)$$

$$T_{t\_CMR} = \frac{(T_{I_1} + T_{D_1}) \times H_{sc} + \sum_{j=2}^N [(T_{I_j} + T_{D_j}) \times H_{rc}]}{N} \quad (2)$$

여기서,  $T_{I_j}$ 는 홉에 대한  $j$ 번째 요청 패킷의 전달 시간,  $T_{D_j}$ 는 홉에 대한  $j$ 번째 데이터 패킷의 전달 시간을 나타낸다.  $H_{sc}$ 는 서버와 클라이언트까지의 전체 홉 수,  $H_{rc}$ 는 CCN 코어 라우터에서 클라이언트까지의 홉 수를 나타낸다.  $N$ 은 반복 요청된 횟수이다.

## 2) 콘텐츠 제공자와 코어 라우터 사이의 홉 수에 따른 콘텐츠 전달

콘텐츠 제공자인 콘텐츠 제공자와 코어 라우터 사이의 홉 수 증가에 따른 end-to-end 전송 시간과 네트워크 대역폭 사용량을 비교하기 위해 4개의 클라이언트들은 하나의 동일한 콘텐츠를 각각 임의의 시간에 10번씩 요청한다. 이는 하나의 콘텐츠를 40번 요청하는 형태로 콘텐츠 제공자와 코어 라우터 사이의 홉 수 증가에 따라 매번 콘텐츠 제공자로 데이터를 요청해야하는 CBT와 달리 CMR-OSPFN은 최초 요청을 제외하고는 콘텐츠 제공자와 코어 라우터 사이의 홉 수에 상관없이 코어 라우터의 CS에 저장된 데이터 패킷을 클라이언트들에게 멀티캐스트하는 방식이다.

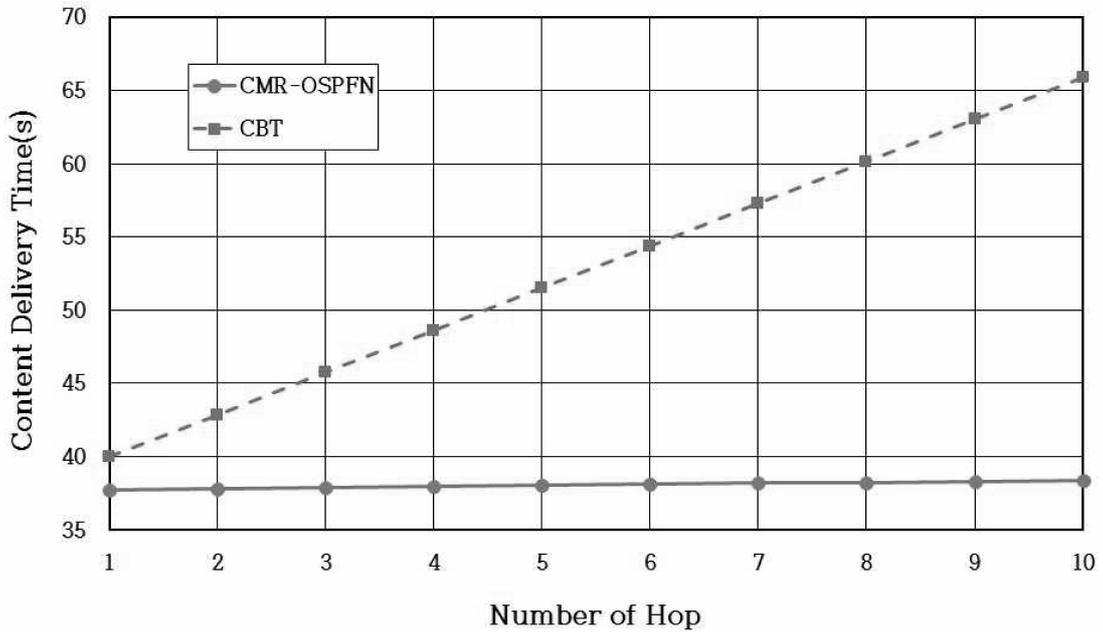


Fig. 38. Content delivery time relative to the number of hops

그림 38은 콘텐츠 제공자와 코어 라우터 사이의 홉 수를 1에서 10으로 증가시키면서 콘텐츠를 전달한 시간을 나타낸 것이다. 콘텐츠 제공자와 코어 라우터 사이의 홉 수가 1인 경우 동일한 콘텐츠를 4개의 클라이언트가 10번을 요청하였을 때 데이터 패킷 전달 시간은 CMR-OSPFN와 CBT는 각각 37.7초와 40.0초이다. 또한 홉 수가 10인 경우 CMR-OSPFN와 CBT는 각각 38.4초와 65.9초이다. CMR-OSPFN은 홉 수가 1에서 10까지 1씩 증가함에 따라 0.7초의 콘텐츠 전달 시간 차이를 보였고, CBT의 경우 28.2초 차이를 보였다. 이는 하나의 동일한 콘텐츠를 40번 요청하는 것으로 CMR-OSPFN의 경우 최초의 요청을 제외하고는 코어 라우터의 CS에 저장된 콘텐츠를 제공받는 것으로 콘텐츠 제공자와 코어 라우터 사이의 홉 수와 상관없이 콘텐츠를 코어 라우터가 제공하기 때문이다.

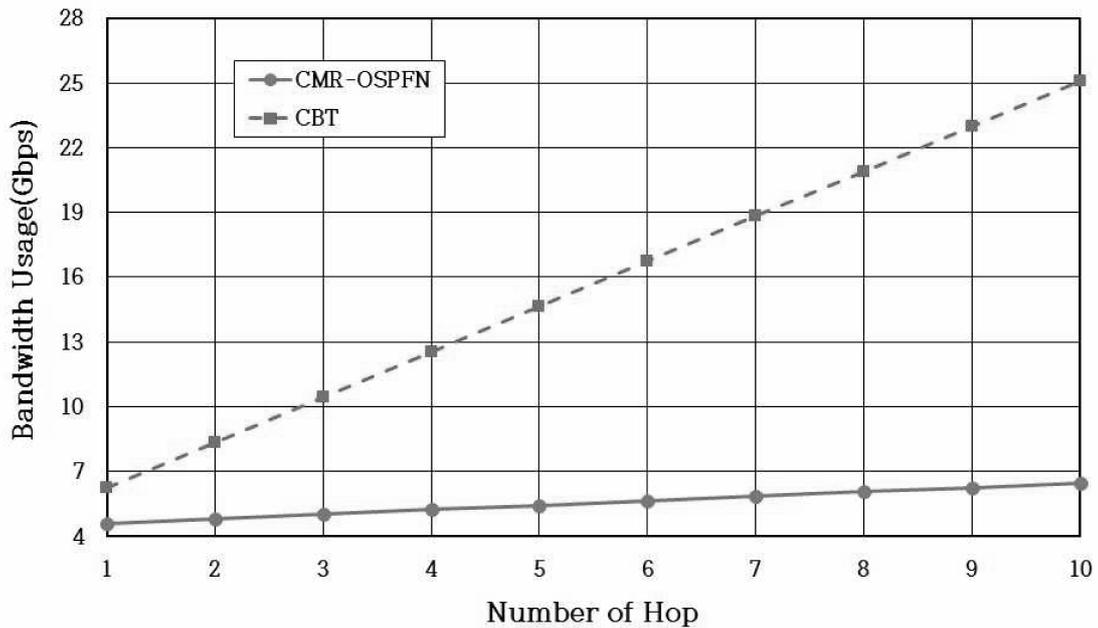


Fig. 39. Network bandwidth usage relative to the number of hops

그림 39는 콘텐츠 제공자와 코어 라우터 사이의 홉 수의 증가에 따른 네트워크의 총 대역폭 사용량을 나타낸 것이다. 여기서 네트워크 대역폭 사용량은 콘텐츠의 요청이 발생했을 때 네트워크 링크에 지나가는 요청 패킷과 데이터 패킷이 사용하는 대역폭의 양으로 정의한다. 콘텐츠 제공자와 코어 라우터 사이의 홉 수가 증가함에 따라 요청 패킷과 데이터 패킷의 전달 수가 증가한다. 콘텐츠 제공자와 코어 라우터 사이의 홉 수가 1일 때 네트워크 대역폭 사용량은 CMR-OSPFN와 CBT에 대해 각각 4.60Gbps와 6.27Gbps이며, 홉 수가 10인 경우는 각각 6.48Gbps와 25.09Gbps를 나타낸다. CMR-OSPFN에 비해 CBT의 대역폭 사용량이 25%이상 많은 것을 알 수 있다. 이는 CBT는 홉 수의 증가함에 따라 데이터를 전달해야 하는 홉 수가 증가하므로 네트워크 링크의 사용량이 많아지는 것을 의미한다.

### 3) 콘텐츠 요청 수에 따른 콘텐츠 전달 시간

CMR-OSPFN와 CBT의 콘텐츠 전달에 대한 성능을 비교하기 위해 클라이언트들

의 동일한 콘텐츠의 요청 횟수에 따른 콘텐츠 전달 시간을 검토한다. 10개의 콘텐츠를 임의의 클라이언트가 1-100번을 반복 요청한다. 이때 콘텐츠 제공자와 코어 라우터는 홉 수는 5로 하였다.

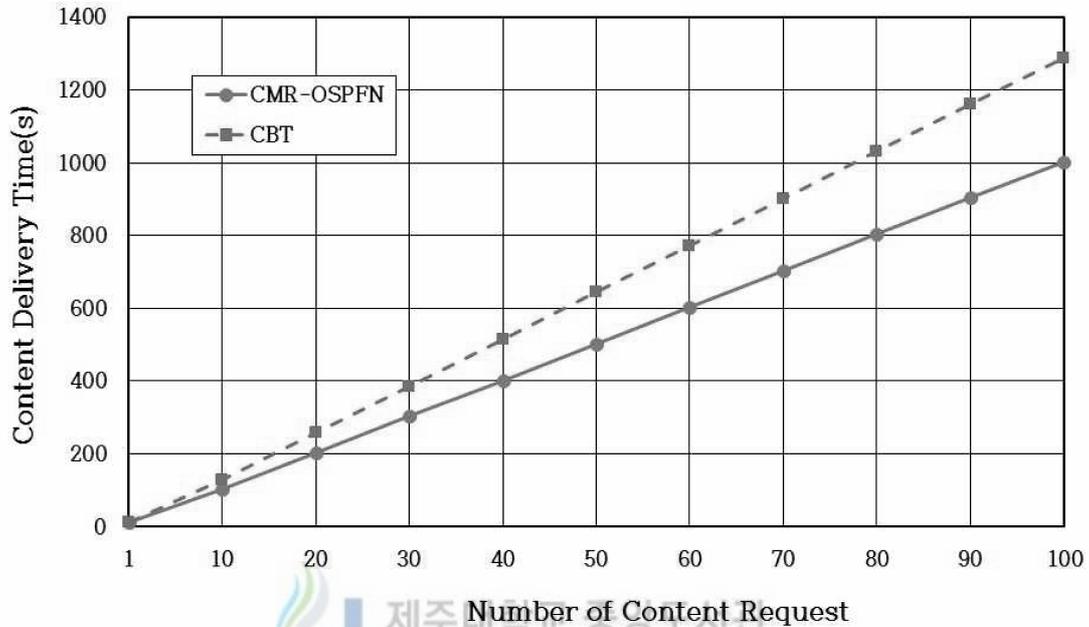


Fig. 40. Content delivery time relative to the number of content request

그림 40은 콘텐츠 요청 수에 따른 콘텐츠 전달 시간을 나타낸 것이다. 콘텐츠 요청 횟수가 1인 경우 10개의 콘텐츠를 CMR-OSPFN와 CBT 둘 다 최초로 요청하는 것이므로 CMR-OSPFN와 CBT의 모두 12.9초로 동일하다. 최초의 요청이 이루어진 이후 재요청되는 콘텐츠에 대해 CMR-OSPFN의 경우 콘텐츠 제공자가 아닌 코어 라우터로부터 멀티캐스트 되어 전송되어진다. 그러나 CBT의 경우 계속해서 콘텐츠 제공자로부터 콘텐츠를 전송받을 것이다. 10개의 콘텐츠가 100번 요청되었을 때 제안한 프로토콜은 1002.9초이고 CBT의 경우 1288.0초로 CMR-OSPFN을 사용했을 경우 CBT보다 285초 빠르게 콘텐츠를 전송받는 것을 확인할 수 있었다.

#### 4) 콘텐츠 수에 따른 콘텐츠 전달 시간

CCN은 라우터의 캐싱 기능 때문에 동일한 콘텐츠가 반복적으로 요청되는 네트워크 환경에 더욱 적합하다. 이에 CMR-OSPFN에서 동일한 콘텐츠를 반복적으로 요청하는 경우와 새로운 콘텐츠를 요청하는 경우 성능을 비교 분석한다. 동일한 콘텐츠를 반복적으로 요청하는 것은 그림 43에서 확인하였다. 이에 다양한 새로운 콘텐츠 요청에 대해 시뮬레이션 하기 위해 새로운 콘텐츠의 수를 10개에서 100개로 증가시키면서 콘텐츠 전달 시간을 측정하였다. 이때 새로운 콘텐츠에 대한 요청은 10번 이루어지며, 콘텐츠 제공자와 코어 라우터는 홉 수는 5로 하였다.

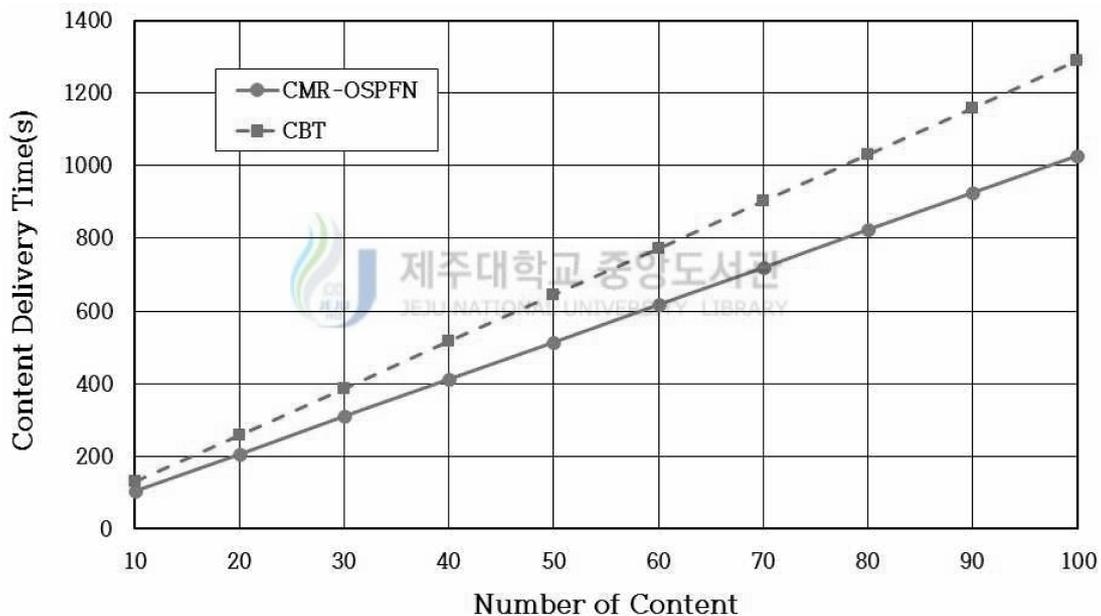


Fig. 41. Content delivery time relative to the number of content

그림 41은 콘텐츠 수에 따른 콘텐츠의 전달 시간을 나타낸다. 콘텐츠 수가 100개인 경우를 살펴보면 100개의 콘텐츠에 대해 임의의 클라이언트들이 각각 10번 요청하므로 최초 요청되는 100번은 콘텐츠 제공자에게 요청하며 반복되는 나머지 900번은 코어 라우터로 요청한다. 여기서 100개의 콘텐츠를 10번 요청하였을 때 콘텐츠 전달 시간은 CBT의 경우 1288.0초이며, 10개의 콘텐츠를 100번 요청한 경우와 동일하다. 그러나 CMR-OSPFN의 경우 100개의 콘텐츠를 10번 요청하였을

때 콘텐츠 전달 시간은 1028.8초이며 10개의 콘텐츠를 100번 요청한 경우 1002.9초로 측정되었다. 이 결과를 통해 콘텐츠를 반복적으로 요청하는 형태에서 더 효율적인 것을 확인할 수 있었다.



## IV. 밀집된 사용자를 위한 CMR-OSPFN 알고리즘의 라이프타임을 사용한 캐싱 기법과 데이터 전달 방식

### 1. 밀집된 사용자를 위한 콘텐츠 공유 서비스

최근 스마트 디바이스의 종류가 다양해지면서 네트워킹 기기를 이용하여 콘텐츠를 공유하는 기술에 대한 연구들이 많이 이루어지고 있다<sup>[36][37][38][39][40]</sup>. 스마트 디바이스 간 화면을 공유하는 기술로 삼성의 올쉐어캐스트와 그룹플레이, 구글의 크롬 캐스트 그리고 에브리온 TV의 에브리온 TV캐스트 등 실제 기기에 적용되고 있다<sup>[41]</sup>. 또한 다중 디바이스간 콘텐츠 공유를 통한 협업 시스템<sup>[42]</sup>, 화면 공유를 위한 N-스크 서비스와 1:N 미러링 플랫폼<sup>[43]</sup> 등 콘텐츠 공유에 관한 연구가 이루어지고 있다. 이러한 다양한 연구들을 통해 같은 공간의 구성원들이 함께 콘텐츠를 공유하기 위한 기술들이 개발되고 있다. 또한 사용자가 밀집한 네트워크 환경인 강의실, 경기장, 공연장, 회의장 등의 장소에서 콘텐츠 공유 서비스가 많이 이루어지고 있다.

예를 들어 그림 42와 같이 학교 강의실에서 콘텐츠를 공유하는 네트워크 환경을 살펴보자. 강의를 시작되기 전에 강의에 참여하는 학생들은 자신의 스마트 디바이스 이나 노트북 등을 이용해 강의실 입장을 서버로 알린다. 서버는 이 정보를 이용하여 출석 체크, 수업 자료 제공, 과제 체크 등 강의 시간 동안 학생 정보를 관리한다. 강의를 시작되면 서버는 강사가 요청하는 강의 자료를 강의실 컴퓨터로 제공하면서, 강의실 내 학생들의 디바이스에도 동일한 콘텐츠를 제공한다. 또한 강의 중 학생들이 생성한 콘텐츠를 제출할 수 있게 하여 서버는 이를 수집한다. 이와 같은 상황은 강의 가 시작되어 끝나는 동안 모든 트래픽이 서버로 집중된다. 또한 하나의 강의실에서 동일한 과목의 수업이 다시 진행되거나 여러 강의실에서 동일한 과목의 수업이 진행되는 경우 서버는 반복적으로 동일한 콘텐츠를 전송해야한다.

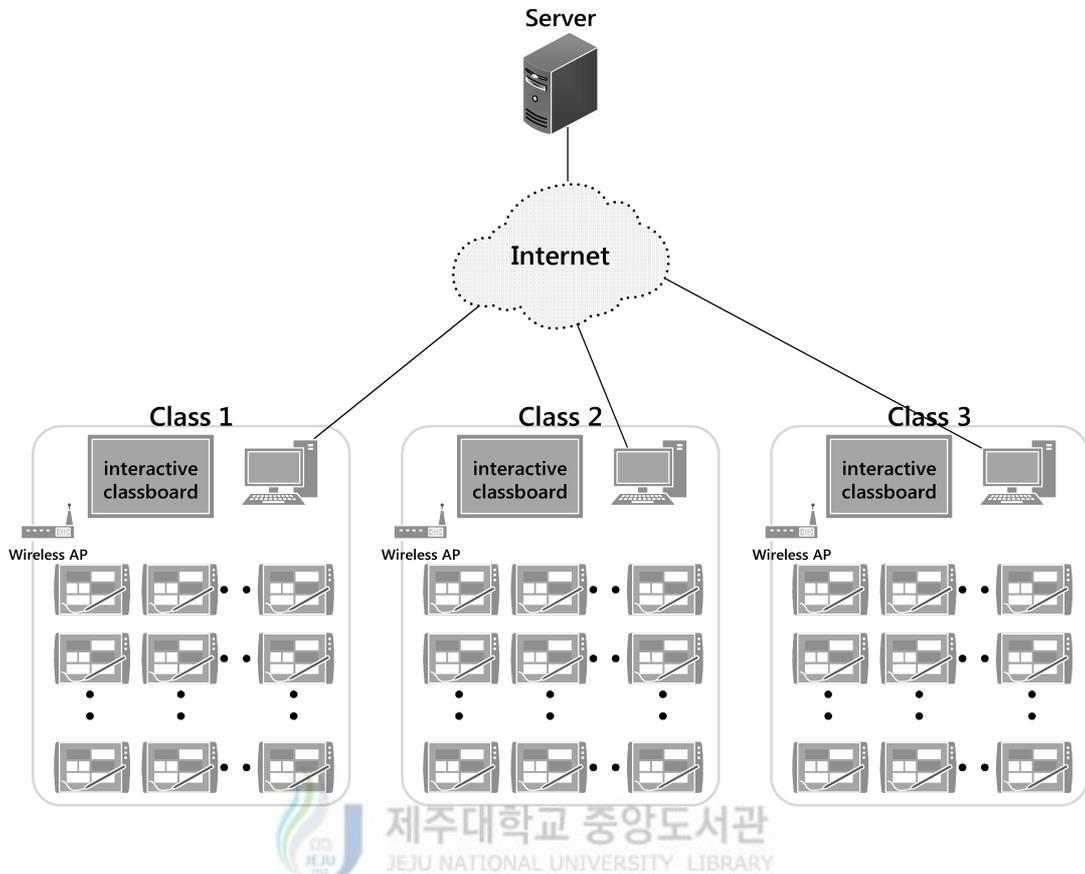


Fig. 42. Share contents in the classroom

그림 43은 경기장에서 각 팀의 응원 어플리케이션을 이용하여 콘텐츠를 공유하는 네트워크 환경을 나타낸 것이다. 경기장을 찾는 각 팀의 관람객은 경기장 입장과 함께 자신의 경기장 입장을 서버에게 알린다. 서버는 이 정보를 이용하여 각 팀의 응원 그룹을 형성한다. 경기가 시작되면 각 팀의 응원이 시작되고 응원그룹 운영자는 서버로 응원 정보를 요청한다. 서버는 경기장 내 모든 응원그룹이 콘텐츠를 공유할 수 있게 요청된 콘텐츠를 전달한다. 이 경우도 학교 강의실과 마찬가지로 여러 경기장에서 동일한 시간에 경기가 진행되는 동안 서버로 트래픽이 집중되며, 각 경기장 마다 반복된 응원으로 동일한 응원 콘텐츠를 서버로 요청한다<sup>[44]</sup>.

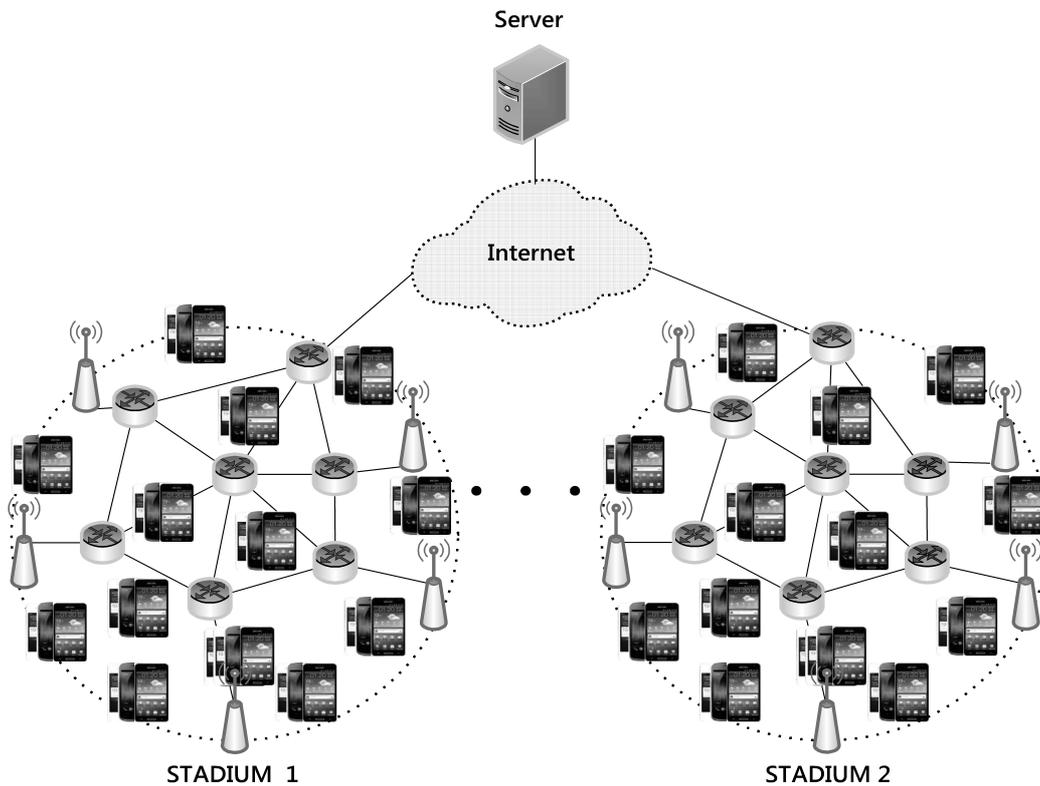


Fig. 43. Share contents in the stadium

인터넷을 사용하여 콘텐츠를 공유하기 위한 사용자가 밀집된 네트워크 환경의 특징은 동일한 시간에 서버로 트래픽이 집중된다는 것과 동일한 콘텐츠가 반복적으로 서버에게 요청된다는 것이다. 이와 같이 서버로 동일한 시간에 트래픽이 집중되는 현상은 서버 주변에 심각한 인터넷 병목현상을 일으킬 것이며 콘텐츠 공유를 위해 콘텐츠를 전송함에 있어 많은 지연을 발생시킨다. 이러한 문제를 해결하기 위해 CCN이 제시되었다.

CCN 기술은 콘텐츠 생산자가 직접 사용자에게 데이터를 전송해야 한다는 제한을 두지 않고 데이터를 보유하고 있는 임의의 CCN 노드들도 데이터를 배포할 수 있다. 그러므로 CCN은 하나의 서버로 콘텐츠의 요청이 집중되는 것을 효과적으로 분산시키며보다 빠르게 데이터를 전송할 수 있다. 또한 같은 데이터가 네트워크에서 반복되어 전송되는 횟수를 줄일 수 있다는 장점을 가진다. 하지만 CCN 기술은 제한된 캐시의 용량으로 인해 광범위하게 분포된 콘텐츠 요청자에게 전달되는 대량의 콘텐츠를

CCN 노드가 저장함에 있어 캐시의 잦은 교체가 발생된다. CCN의 기본 캐싱 기법은 요청 패킷을 기억하는 PIT에 의해 데이터 패킷으로 콘텐츠 전달 시 그림 44와 같이 역방향 경로를 사용하여 콘텐츠를 전송한다. 이는 저장과 전달을 통해 콘텐츠가 지나가는 모든 노드의 CS에 콘텐츠를 LRU 또는 LFU 교체 방식으로 저장하는 것이다<sup>[11]</sup>.

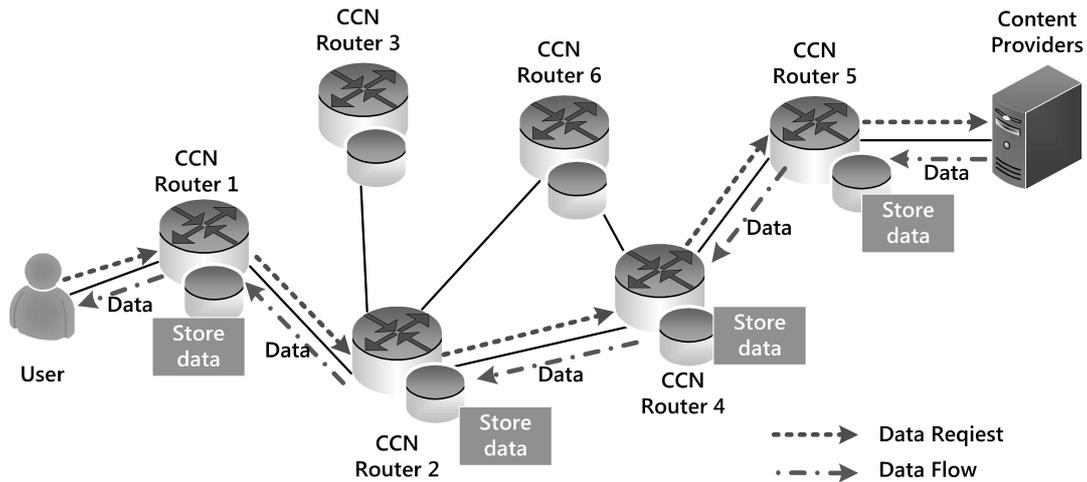


Fig. 44. Content delivery process

LRU 알고리즘은 최근 참조 경향에 기반을 두어 객체를 삭제하는 알고리즘으로 저장되어 있는 객체 중 가장 오래 참조되지 않은 객체를 삭제한다. 하지만 LRU 기법은 콘텐츠 변화에 잘 적응하지만 마지막 참조 시간을 이용하므로 빈번히 참조되는 콘텐츠와 그렇지 못한 콘텐츠를 구분하지 못한다. 또한 LFU 알고리즘은 참조 횟수에 기반을 두어 객체를 삭제하는 알고리즘으로 저장되어 있는 객체 중 가장 적게 참조된 객체를 삭제한다. LFU 기법의 문제점은 참조의 최근성을 고려하지 않아 과거에 빈번히 참조된 콘텐츠가 캐시를 채운 경우 캐시 오염(cache pollution) 현상이 발생할 수 있다<sup>[45]</sup>. LRU, LFU는 단순히 콘텐츠가 최신형이거나 요청 빈도가 많은 콘텐츠만을 캐싱하는 기법으로 콘텐츠의 특성과 CCN 노드의 제한된 캐시 용량에 대한 고려가 없어 잦은 캐시 교체를 발생시킬 수 있다. 이러한 캐시의 잦은 교체 문제를 해결하기 위해서는 노드들이 광범위하게 분포된 네트워크에서가 아닌 콘텐츠 특성이 고려된 소규모 네트워크에서의 연구가 필요하다.

이에 본 절에서는 콘텐츠 특성을 고려한 사용자가 밀집된 네트워크에서 콘텐츠 공

유를 위해 CMR-OSPFN 라우팅 알고리즘을 적용하여 효율적인 캐싱 기법과 데이터 전달 방식을 제안하고자 한다. 이는 CCN에서의 콘텐츠 공유를 위한 일반적인 인터넷 환경이 아닌 CCN을 적용하였을 경우 기술적인 효과를 극대화하고자 하는 것이다.

## 2. 밀집된 사용자 환경에서 라이프타임을 사용한 캐싱 기법

본 절에서는 밀집된 사용자 환경에서 3장에서 제안한 CCN에서 멀티캐스트 라우팅 알고리즘인 CMR-OSPFN 알고리즘을 적용한 콘텐츠 공유 그룹 내의 캐시의 효율을 높이기 위해 휘발성 콘텐츠를 이용한 새로운 메모리 관리 기법을 제안하고자 한다. 기존 CCN의 캐싱 기법은 LRU와 LFU로 각각 최근 사용된 콘텐츠와 자주 사용된 콘텐츠를 캐시에 저장하는 방식을 사용한다. 이는 사용자가 밀집된 CCN 환경에서 그룹 통신을 위해서는 일정 시간 서비스를 제공하는 경우에는 적합하지 않다.

본 논문에서 제안하는 캐싱 기법은 콘텐츠 제공자로부터 라이프타임(lifetime) 필드가 추가된 콘텐츠를 전달받아, 캐시에 저장하고 일정 시간 서비스한 후 라이프타임이 종료되면 삭제하는 방식이다. 캐시에 일정한 시간 동안만 생성되었다가 삭제되는 콘텐츠를 휘발성 콘텐츠(volatile content)라 정의한다. 콘텐츠 제공자는 콘텐츠 사용 장소의 특성을 파악하여 관련 콘텐츠를 CCN 라우터의 CS에 제공하며 라우터는 콘텐츠의 라이프타임이 지나면 콘텐츠를 CS에서 삭제한다. 휘발성 콘텐츠의 이용은 특정 장소의 트래픽이 많은 환경에서 콘텐츠를 더욱더 빠르게 사용자에게 제공할 수 있을 것이다. 또한 휘발성 콘텐츠는 많은 사람들이 모여 일정 시간 동안 그룹 통신을 하는 스포츠 경기장, 학교 교실, 공연장 등의 공간에서 활용한다면 매우 효율적일 것이다.

그림 45는 휘발성 콘텐츠를 이용한 스포츠 경기장에서의 캐싱 과정을 나타낸 것이다. 그림 45의 State1은 경기장을 찾은 클라이언트들이 콘텐츠 공유 그룹에 참가 메시지를 전송하면 코어 라우터는 콘텐츠 공유 그룹을 형성하여 서버에게 알린

다. 서버는 해당 경기장에 대한 관련된 라이프타임이 포함된 모든 데이터들을 코어 라우터로 전송한다. 그림 45의 State 2는 경기가 진행 중에 클라이언트들이 요청하는 콘텐츠를 코어 라우터는 서버에게 요청하지 않고 CS에 저장된 데이터를 이용해 전달한다. 그림 45의 State 3은 경기가 끝난 후 경기장 내의 라우터들은 자신에게 저장된 데이터의 라이프타임을 이용하여 삭제한다.

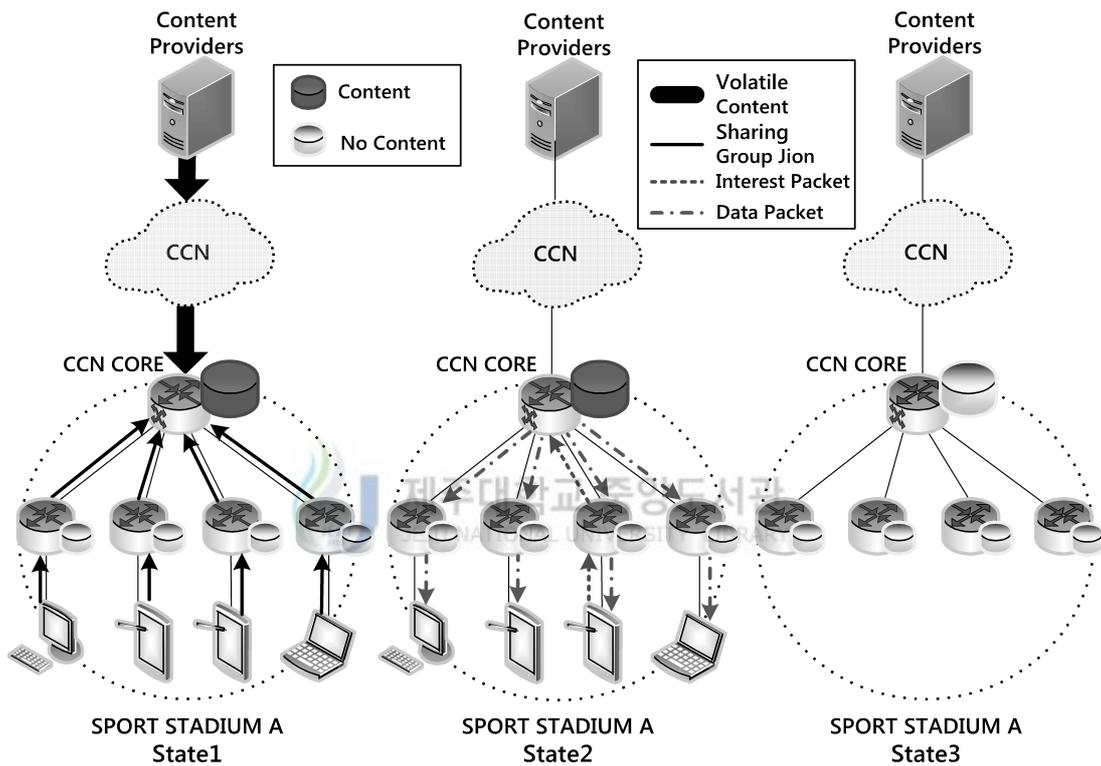


Fig. 45. Caching and data delivery in sport stadium

그림 46과 그림 47은 본 논문에서 제안한 휘발성 콘텐츠를 이용한 서버와 코어 라우터의 캐싱 및 데이터 전달 절차를 흐름도로 나타낸 것이다. 그림 46은 서버의 캐싱 및 데이터 전달 과정으로 네트워크에서 콘텐츠 공유 그룹 형성에 대한 과정이 이루어지고 나면 서버는 해당 경기장에 경기 정보를 확인하여 경기에 해당하는 관련 콘텐츠에 라이프타임 필드를 추가하여 코어 라우터로 전송한다. 또한 서버는 경기가 끝나면 경기 진행 중 코어 라우터가 수집한 새로운 정보를 수신하여 이를 서버의 데이터베이스에 저장한다.

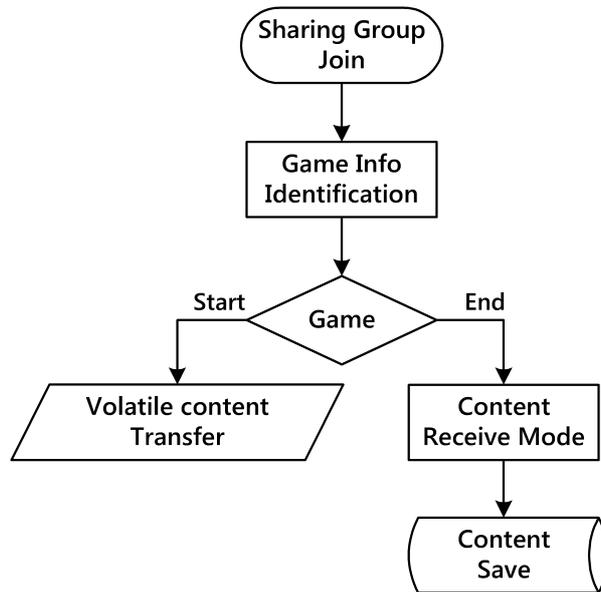


Fig. 46. Caching and Data delivery on the server

그림 47은 코어 라우터의 캐싱 및 데이터 전달 절차로 서버와 같이 네트워크에서 라우팅과 콘텐츠 공유 그룹 형성에 대한 과정이 이루어지고 나면 서버로부터 경기에 해당하는 휘발성 콘텐츠를 수신 받는다. 코어 라우터가 콘텐츠를 수신하여 자신의 CS에 저장한다. 코어 라우터는 공유 그룹에 대한 멀티캐스트 요청 패킷을 수신하면 자신의 CS를 확인하여 콘텐츠가 있으면 멀티캐스트 그룹으로 데이터 패킷을 전달하고 콘텐츠가 없는 경우 콘텐츠를 찾기 위해 다시 요청 패킷을 이웃 라우터로 전달한다. 코어 라우터는 요청 패킷에 해당하는 데이터 패킷을 수신하여 자신의 CS에 저장한다. 경기가 끝나면 코어 라우터는 자신의 CS에서 서버로부터 수신한 콘텐츠는 라이프타임을 이용하여 삭제하며, 경기 중 새로 수신된 콘텐츠는 서버로 전송 후 삭제한다.

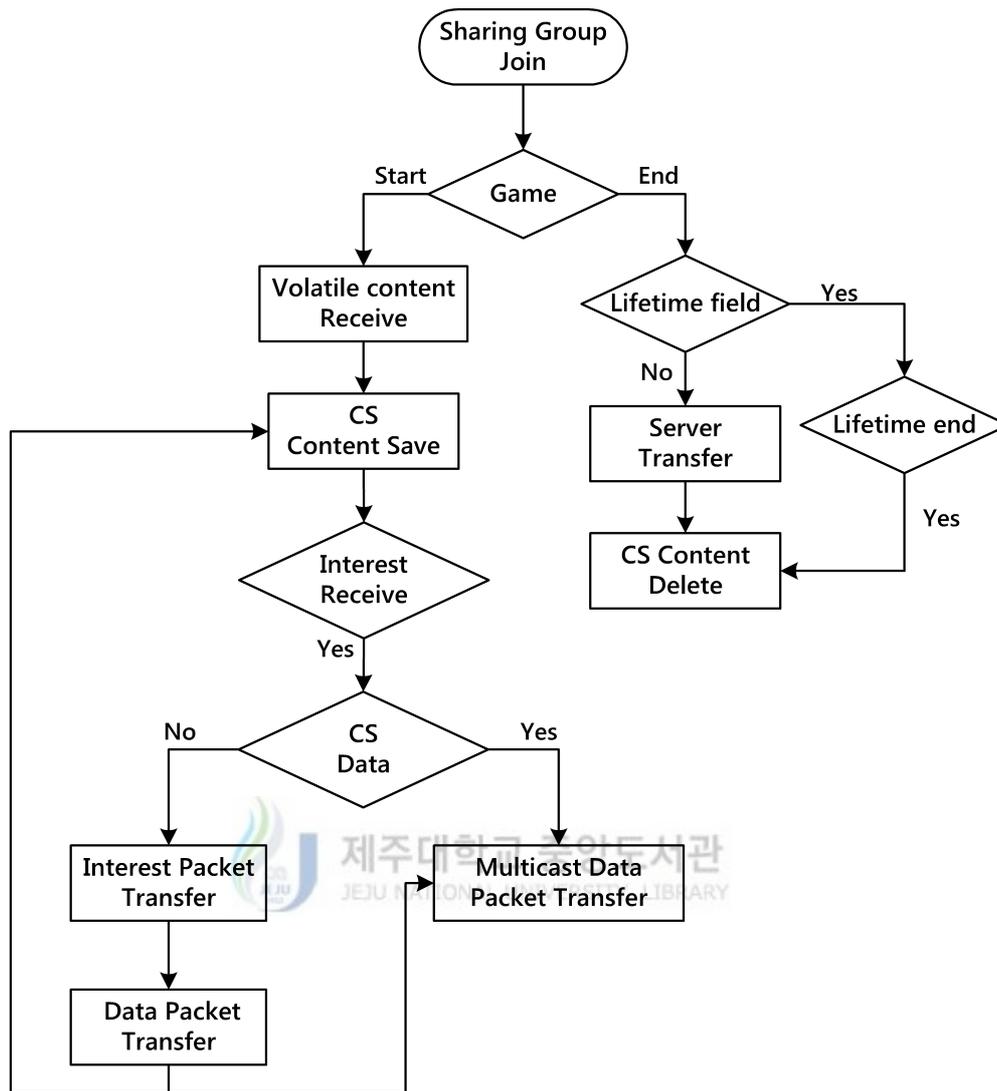


Fig. 47. Caching and Data delivery on the core router

### 3. 콘텐츠 서비스 시간 단축을 위한 OROC 전달 방식

본 절에서는 밀집된 사용자 환경에서 CMR-OSPFN 알고리즘을 적용한 콘텐츠 공유 그룹 내의 콘텐츠 서비스 시간 단축을 위한 새로운 OROC(One-Request-One-Content)전송 방식을 제안한다. CCN은 하나의 콘텐츠를 청크 단위인  $N$ 개의

데이터 패킷으로 나누어 전송하며, 하나의 데이터 패킷 전송에 대해 하나의 요청 패킷을 전달하는 OROP(One-Request-One-Packet)방식을 사용한다. OROP는 하나의 콘텐츠를 전달하기 위해서는  $N$ 개의 요청 패킷이 필요하다. 그룹 통신을 위하여 사용자가 밀집된 CCN 환경에서 CMR-OSPFN 알고리즘을 사용하여 그룹 통신을 할 때 모든 멀티캐스트 데이터는 CCN 코어 라우터로부터 전달된다. 이때 OROP 전송 방식을 이용할 경우 신뢰성 있는 전송을 보장할 수 있지만 불필요한 요청 패킷으로 데이터 전송 지연이 발생한다.

제안하는 OROC는 하나의 요청 패킷에 대해 하나의 콘텐츠인  $N$ 개의 데이터 패킷을 전달하는 방식이다. OROC를 이용할 경우 하나의 콘텐츠에 대한 연속적인 시퀀스를 갖는 데이터 패킷을 전송할 때 연속적인 시퀀스의 데이터 패킷에 대해 반복적으로 요청되는 요청 패킷을 줄여 콘텐츠 서비스 시간을 줄일 수 있다.

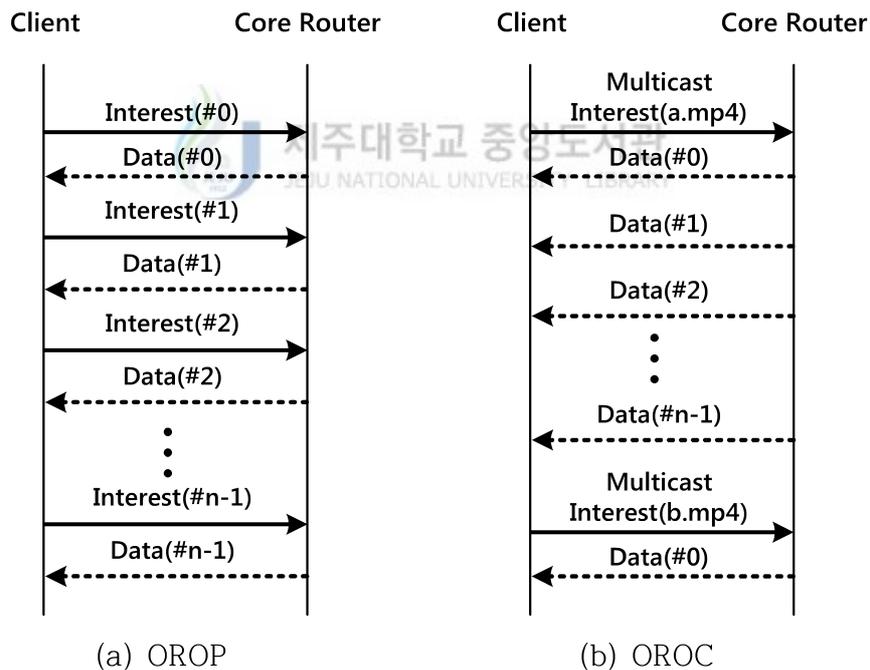


Fig. 48. The comparison of the two transmission method

그림 48은 OROP 전달 방식과 OROC 전달 방식의 동작 과정을 나타낸 것이다. 그림 48 (a)는 OROP 데이터 전달 과정을 살펴보면 클라이언트가 콘텐츠의 첫 번째 데이터 패킷 조각인 #0의 데이터 패킷을 요청하기 위해 요청 패킷의 콘텐츠 이

름에 시퀀스 정보 0을 포함하여 콘텐츠 제공자에게 전달한다. 콘텐츠 제공자는 요청 패킷이 요구한 #0에 대한 데이터 패킷을 클라이언트에게 제공한다. OROP는 클라이언트가  $n$ 개로 구성된 하나의 콘텐츠를 제공받기 위해서는  $n$ 개 요청 패킷 전송과  $n$ 개의 데이터 패킷 수신 과정을 반복해야 한다. 식 (3)은 OROP 전달 방식에서  $n$ 개로 구성된 하나의 콘텐츠를 전달하는데 걸리는 전체 시간  $T_{t\_OROP}$ 을 나타낸 것이다.

$$T_{t\_OROP} = \sum_{j=1}^N [(T_{I_j} + T_{D_j}) \times H_{rc}] \quad (3)$$

여기서,  $T_{I_j}$ 는 홉에 대한  $j$ 번째 요청 패킷의 전달 시간,  $T_{D_j}$ 는 홉에 대한  $j$ 번째 데이터 패킷의 전달 시간,  $H_{rc}$ 는 CCN 코어 라우터에서 클라이언트까지의 홉 수,  $N$ 는 하나의 콘텐츠에 대한 데이터 패킷의 수를 나타낸다.

그림 50 (b)는 OROC의 동작 과정을 나타낸 것으로 클라이언트는 하나의 콘텐츠를 제공받기 위해 하나의 요청 패킷을 전송한다. CCN 코어 라우터는 a.mp4에 해당하는 요청 패킷을 수신하면 a.mp4에 해당하는  $n$ 개의 데이터 패킷을 순차적으로 내보낸다. 이때 콘텐츠를 전달하는데 걸리는 전체 시간  $T_{t\_OROC}$ 은 식 (4)와 같다.

$$T_{t\_OROC} = (T_{I_c} + H_{rc}) + \sum_{j=1}^N [T_{D_j} \times H_{rc}] \quad (4)$$

여기서,  $T_{I_c}$ 는 홉에 대한 콘텐츠 요청 패킷의 전달 시간이다. 식 (3)과 식 (4)에서도 알 수 있듯이 OROC는 OROP 전달 방식에 비교해  $N-1$ 개의 요청 패킷을 전달하는데 걸리는 시간만큼 데이터 전송 시간을 줄일 수 있다.

#### 4. 시뮬레이션 결과 및 고찰

##### 1) 시뮬레이션 네트워크 토폴로지

제안하는 밀집된 사용자를 위한 CMR-OSPFN 알고리즘의 효율적인 캐싱 기법과 데이터 전달 방식에 대한 성능 분석을 위한 NS2를 사용하며, 시뮬레이션 토폴로지는 그림 49와 같다. 토폴로지는 콘텐츠 제공자, CCN 코어 라우터, CCN 라우터, 클라이언트로 구성된다. 콘텐츠 제공자는 각 영역의 멀티캐스트 그룹을 관리하는 코어 라우터와 연결되어 있다.

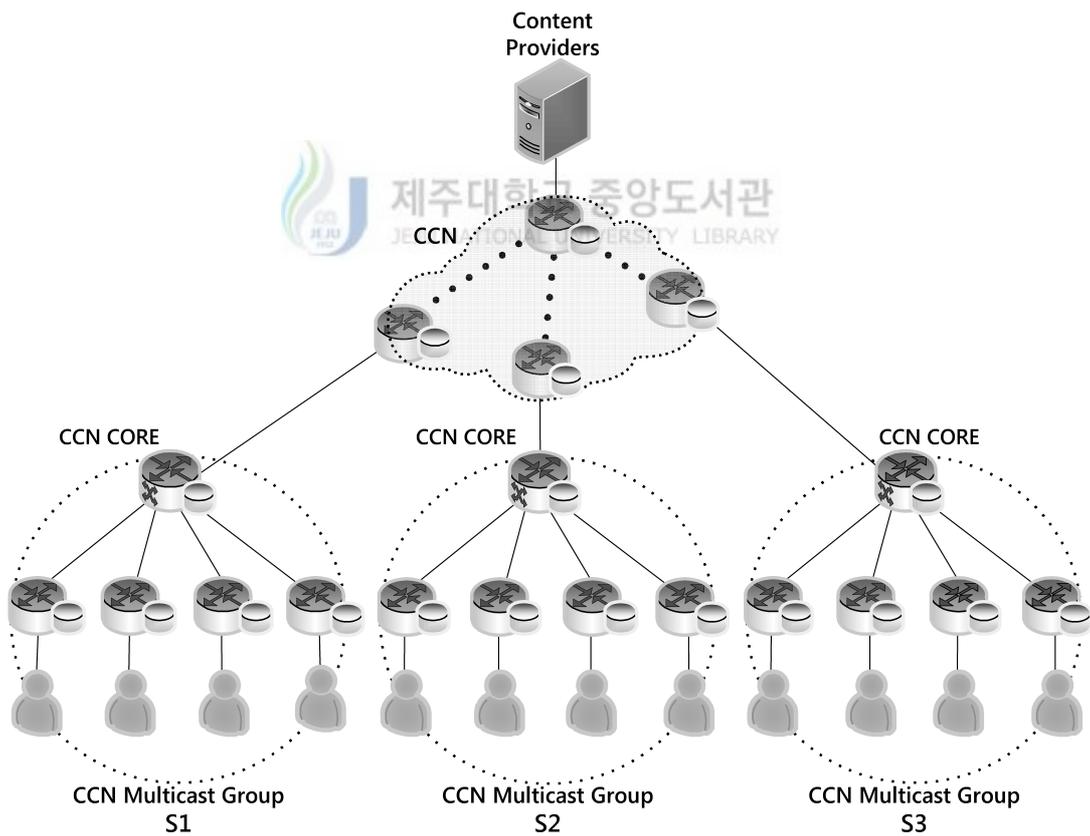


Fig. 49. Network topology of the user dense CCN environment

시뮬레이션에 사용된 파라미터는 각 노드간의 링크 속도는 100Mbps, 전송 선로의 전송 지연은 0.1ms로 가정하였다. 또한 요청 패킷의 크기는 1Kbyte이고 데이터 패킷의 크기는 50Kbyte이며 CCN 라우터의 캐시 용량은 10Gbyte로 각각 고정한다.

## 2) 라이프타임을 사용한 캐시의 성능 분석

### (1) 콘텐츠 수에 따른 캐시 성능 분석

클라이언트가 새로운 콘텐츠를 요청할 경우 요청 패킷은 CCN 코어 라우터로 전달된다. 휘발성 콘텐츠를 사용하지 않은 기존 캐싱 방식은 CCN 코어 라우터가 콘텐츠 제공자인 콘텐츠 제공자로 콘텐츠를 요청할 것이지만 휘발성 콘텐츠를 사용할 경우 콘텐츠를 제공자를 통해 각 코어 라우터는 그룹 통신에 필요한 콘텐츠들이 저장되어 있으므로 CCN 코어 라우터는 콘텐츠 제공자로 콘텐츠를 요청하지 않고 바로 클라이언트에게 콘텐츠를 전달한다. 제안하는 휘발성 콘텐츠를 사용한 캐싱 방식의 성능을 분석하기 위해 CMR-OSPFN의 CCN 코어 라우터에 휘발성 콘텐츠를 사용한 경우와 사용하지 않은 경우에 대해 그룹 내 임의의 한 클라이언트가 계속적으로 새로운 콘텐츠를 요청할 때 평균 콘텐츠 서비스 시간을 측정하였다.

그림 50은 콘텐츠 수의 증가에 따른 콘텐츠 전달 시간을 나타낸 것이다. 휘발성 콘텐츠를 사용하지 않은 캐싱 방식은 새로운 콘텐츠를 요청할 때마다 콘텐츠 제공자인 콘텐츠 제공자로 데이터 패킷을 요청하기 때문에 콘텐츠를 CCN 코어 라우터로부터 제공받는 휘발성 콘텐츠를 사용한 캐싱 방식보다 평균 콘텐츠 서비스 시간이 증가하는 것을 알 수 있다. 10개의 새로운 패킷을 전달했을 때 휘발성 콘텐츠를 사용한 캐싱 방식은 9.62초의 시간이 걸리고 휘발성 콘텐츠를 사용하지 않은 캐싱 방식은 13.08초로 차이를 보인다. 또한 80개의 새로운 패킷을 전달했을 때 휘발성 콘텐츠를 사용한 캐싱 방식은 86.62초의 시간이 걸리고 휘발성 콘텐츠를 사용하지 않은 캐싱 방식은 119.72초로 차이를 보인다.

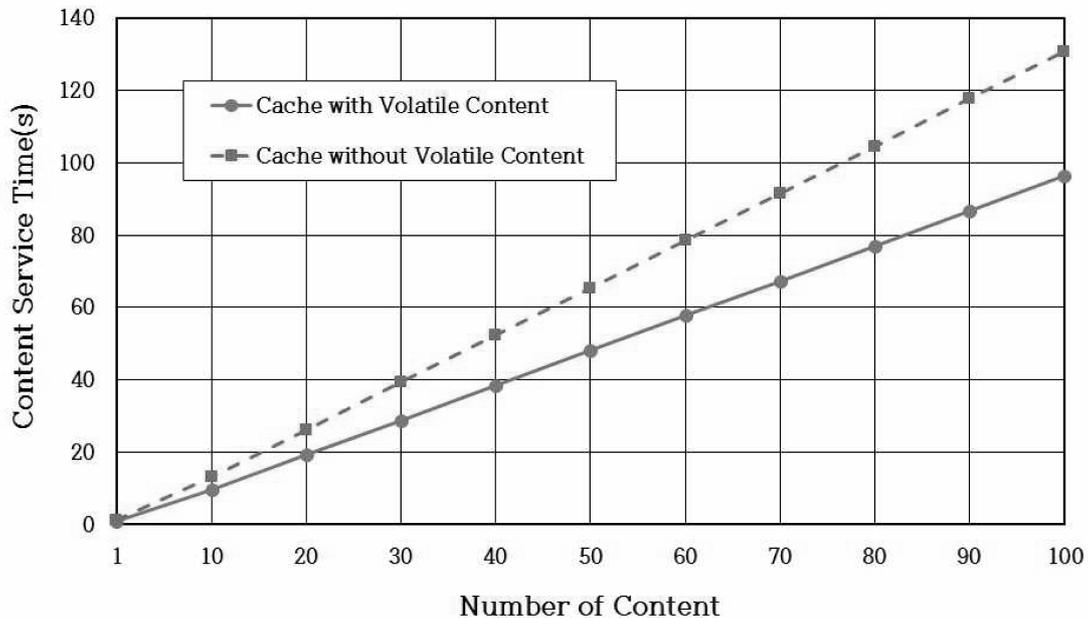


Fig. 50. Content service time relative to the number of content

## (2) 콘텐츠 제공자와 코어 라우터간 홉 수 따른 캐쉬 성능 분석

새로운 콘텐츠에 대해 콘텐츠 제공자가 아닌 CCN 코어 라우터로부터 콘텐츠를 제공받는 제안한 휘발성 콘텐츠를 사용한 캐싱 방식의 콘텐츠 서비스 효율을 살펴 보기 위해 콘텐츠 제공자와의 홉 수에 따른 성능을 분석한다. 콘텐츠의 요청은 하나의 임의의 클라이언트로부터 요청되며, 10개의 새로운 콘텐츠를 콘텐츠 제공자와 CCN 코어 라우터간 홉 수를 1에서 10까지 1홉씩 변화시키며 평균 콘텐츠 서비스 시간을 확인하였다.

그림 51은 콘텐츠 제공자와 CCN 코어 라우터간의 홉 수에 따른 콘텐츠 서비스 시간을 나타낸 것이다. 제안된 휘발성 콘텐츠를 사용한 캐싱 방식의 경우 CCN 코어 라우터로부터 콘텐츠를 제공 받으므로 콘텐츠 제공자와 CCN 코어 라우터간 홉 수가 증가함에도 영향을 받지 않는 것을 확인할 수 있다. 하지만 휘발성 콘텐츠를 사용하지 않는 캐싱 방식의 경우 콘텐츠 제공자로 콘텐츠를 요청하므로 홉 수의 증가에 따라 콘텐츠의 서비스 시간 역시 증가하는 것을 볼 수 있다.

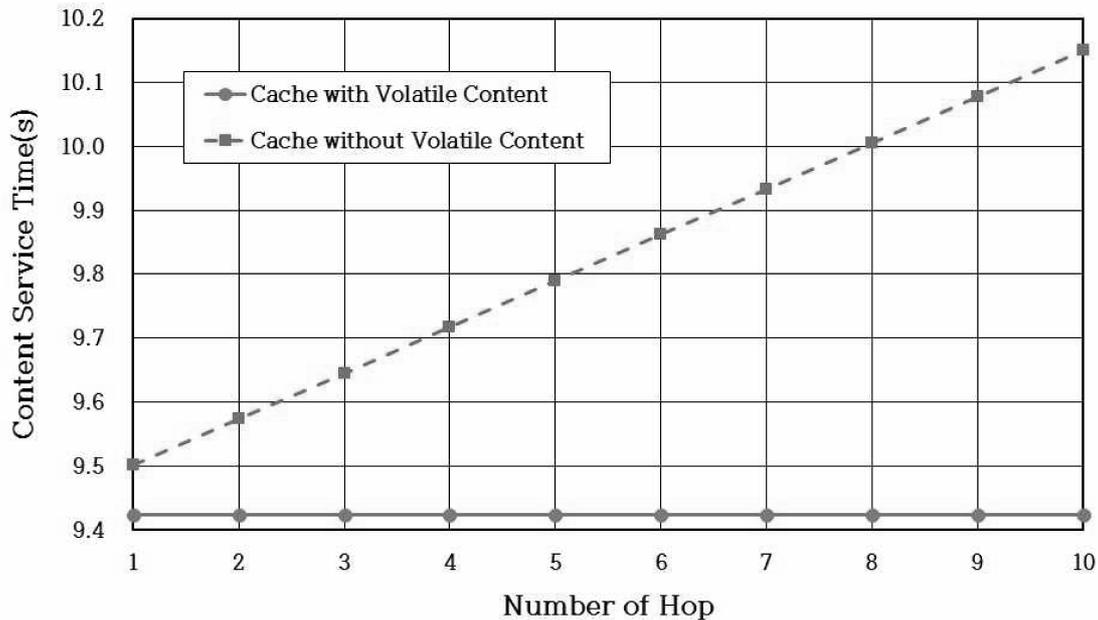


Fig. 51. Content service time relative to the number of hops

### 3) OROC 전달 방식의 성능 분석



제안한 OROC 전달 방식에 대한 성능을 분석하기 위해 콘텐츠의 크기를 변화시키며 평균 콘텐츠 서비스 시간을 측정하였다. 하나의 콘텐츠는 50Kbyte의 데이터 패킷으로 나누어진다. OROP 전달 방식은 여러 개로 나뉘어진 데이터 패킷을 요청하기 위해 데이터 패킷마다 요청 패킷을 전달한다. 하지만 OROC 전달 방식은 하나의 콘텐츠에 대한 하나의 요청 패킷을 전달하도록 하였다.

그림 52는 콘텐츠 크기에 따른 네트워크에서 발생된 요청 패킷의 크기로 OROC 전달 방식은 하나의 콘텐츠의 요청에 대한 하나의 요청 패킷을 보내므로 콘텐츠의 크기와 상관없이 요청 패킷의 크기는 동일하다. 하지만 데이터 패킷에 대해 각각의 요청 패킷을 요청해야하는 OROP 전달 방식의 경우는 콘텐츠의 크기가 커질수록 요청한 요청 패킷의 수의 증가로 네트워크 내에서 발생된 요청 패킷의 크기가 커지는 것을 볼 수 있다.

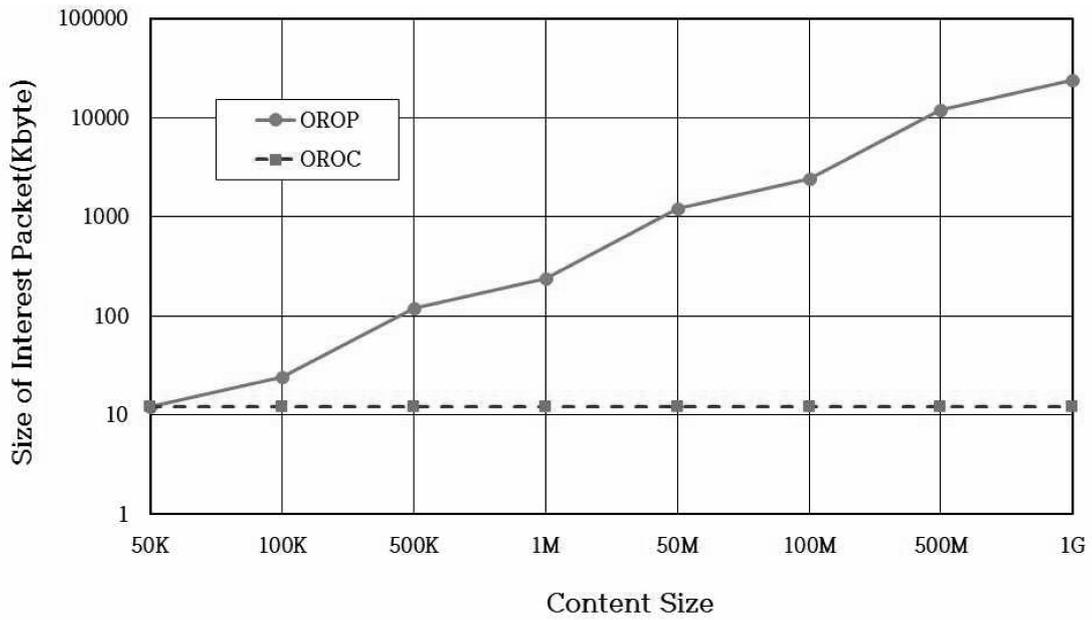


Fig. 52. The size of Interest Packet relative to the content size

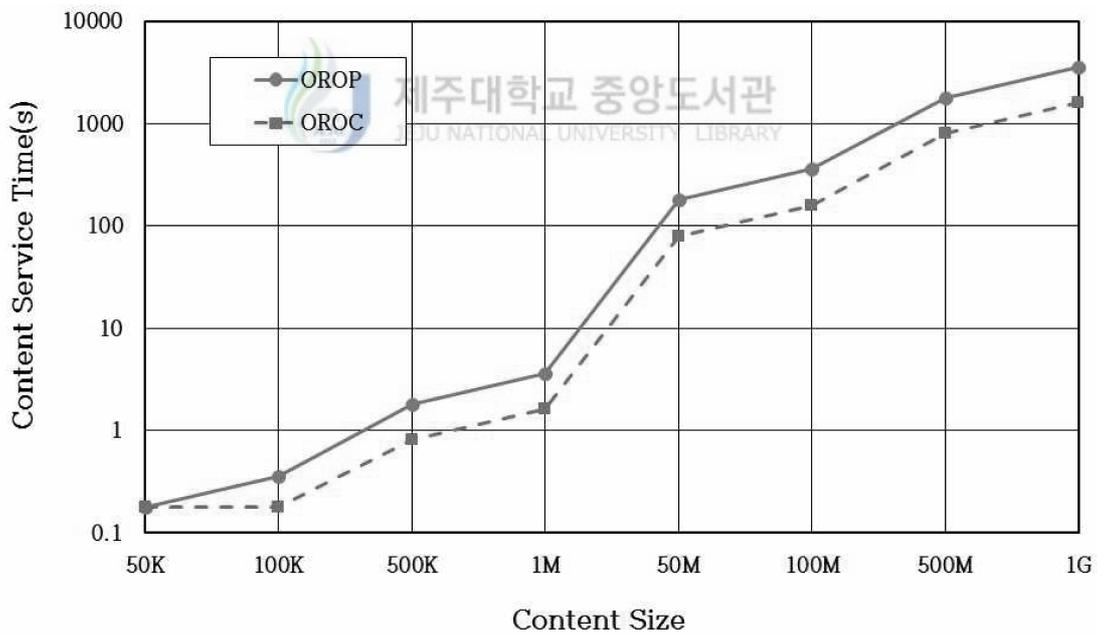


Fig. 53. Content service time relative to the content size

그림 53은 콘텐츠 크기에 따른 평균 콘텐츠 서비스 시간을 나타낸 것으로 50Kbyte로 데이터 패킷의 크기를 고정한 경우 1Mbyte에서 평균 콘텐츠 서비스

시간은 OROC 전달 방식에서는 1.62초, OROP 전달 방식은 3.60초로 대략 2초 정도의 평균 콘텐츠 서비스 시간차를 보인다. 콘텐츠의 크기를 1Gbyte인 경우 OROC 전달 방식에서는 1599.62초, OROP 전달 방식은 3589.92초로 1990.31초의 차이를 보여 콘텐츠의 크기가 커질수록 평균 콘텐츠 서비스 시간의 차이가 현저히 커지는 것을 알 수 있다.



## V. CMR-OSPFN 검증을 위한 스포츠 모바일 서비스 구현

CCN이 현 인터넷의 근간을 이루는 IP를 대체할 가능성은 그리 높지 않다. CCN이 라우터나 스위치와 같은 네트워킹 기기에 적용되기 위해서는 CCN 효과나 기술 성숙도가 충분히 검증되어야 할 것이다. 이에 CCN 기술을 다양한 응용 서비스에 적용하여 기술의 가치가 검증되고 있다.

본 논문에서는 기존 IP 기반을 그대로 이용하면서 스포츠 경기장이라는 사용자가 밀집된 CCN 환경에서 멀티캐스트 그룹을 형성하여 그룹 내 사용자가 콘텐츠를 공유할 수 있는 스포츠 모바일 플랫폼과 모바일 응원 어플리케이션을 개발하고자 한다. 또한 CCN이 스포츠 모바일 플랫폼 구축과 모바일 응원 어플리케이션이 어떻게 활용될 수 있는지를 검토함으로써 CCN 기술의 유용성과 향후 기술 확산 가능성에 대해 알아보하고자 한다.



### 1. 스포츠 모바일 서비스의 사용자 동기 방식

스포츠 모바일 서비스 플랫폼을 CCN에서 적용하고자 하나 아직 CCN이 라우터나 스위치에 적용되지 않았다. 이에 본 논문에서는 스포츠 경기장 내에 CCN 멀티캐스트 환경을 구축하기 위해 CCN 코어 라우터와 CCN 라우터의 기능을 수행할 수 있는 네트워크 기기를 이용한다.

서버는 경기장마다의 경기 일정을 관리하며 경기 일정에 맞추어 경기에 관련된 콘텐츠와 멀티캐스트 그룹 정보를 CCN 코어 라우터로 전송한다. 경기장을 찾는 스포츠 모바일 서비스 플랫폼 사용자들의 멀티캐스트 그룹 참가 요청을 통해 CCN 코어 라우터는 멀티캐스트 그룹에 대한 테이블을 형성한다. 멀티캐스트 콘텐츠 요청은 스포츠 경기 운영자에게 권한을 주어 경기장 내 멀티캐스트 그룹에게 멀티캐스트 하고자 하는 콘텐츠를 CCN 코어 라우터로 요청하여 전달한다. CCN 코어 라우터가 멀티

캐스트 그룹 사용자에게 같은 시간에 콘텐츠를 공유할 수 있게 하기 위해서는 플랫폼 내의 기기들은 계층적 구조를 이루어야 한다. 그림 54는 스포츠 모바일 서비스 플랫폼 내의 기기들의 계층적 구조를 나타낸다.

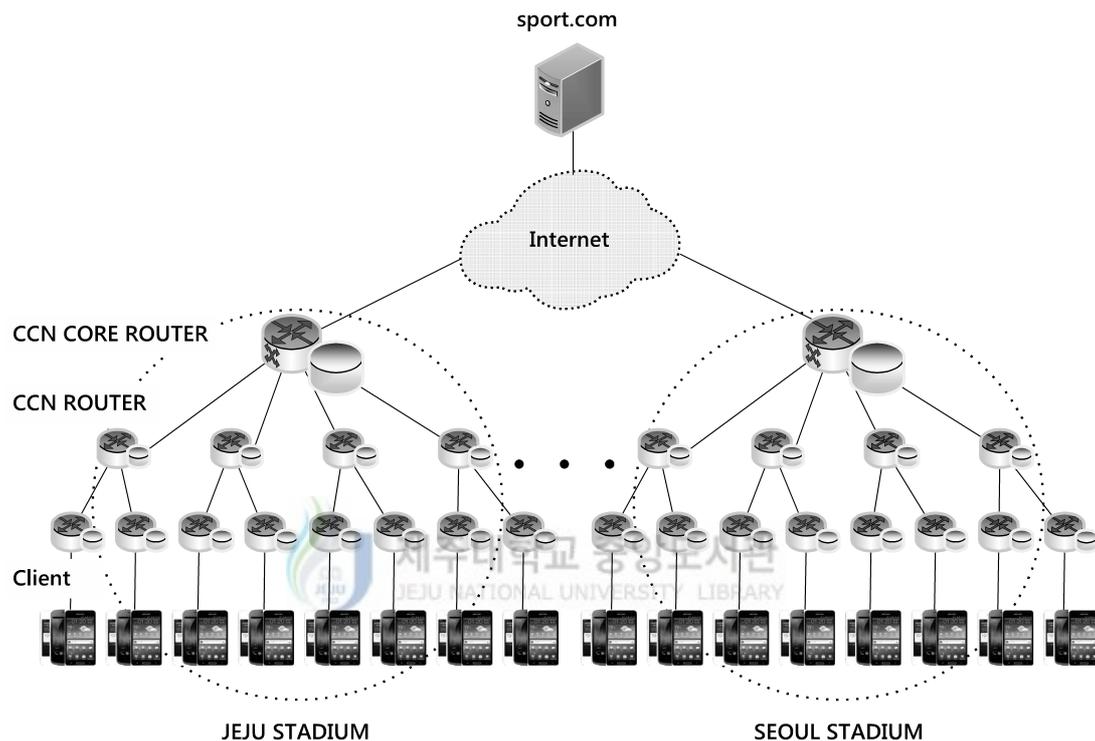


Fig. 54. The hierarchical structure of devices in the platform

## 2. 스포츠 모바일 서비스의 데이터베이스 구조

스포츠 모바일 서비스 플랫폼은 플랫폼내의 기기들의 계층적 구조를 형성하고 있다. 각 스포츠 경기장마다 다른 구단과의 경기가 있을 경우 서버는 각 경기장에서 요청하는 콘텐츠를 CCN 코어 라우터로 제공할 것이다. 이에 본 논문에서는 CCN에서 스포츠 모바일 서비스 플랫폼을 효율적으로 활용하기 위한 관계형 데이터베이스를

구축하였다.

그림 55는 데이터베이스의 구조로 전체 스포츠에 대한 공통된 콘텐츠 및 가입자 관리가 가능하게 구축하였다. 전체 스포츠에 대한 공통된 데이터를 다시 각 스포츠 종목으로 분리하여 스포츠 종목에 해당하는 공통된 콘텐츠 및 가입자를 관리할 수 있도록 하였으며 이를 다시 각 종목의 프로 구단들로 분리하였다. 이렇게 데이터베이스를 분리하여 구축한 이유는 경기장 마다 경기 일정에 따라 진행되는 여러 구단의 콘텐츠 및 가입자 정보를 효율적으로 활용하기 위해서이다. 관리형 데이터베이스는 기존 CCN 라우터의 캐싱 기능의 문제점인 캐시용량의 한계로 인한 잦은 캐쉬 교체 의 문제 또한 해결할 수 있다. 각 경기 일정에 따라 경기장에서 사용되는 휘발성 콘텐츠는 경기 참가 구단의 콘텐츠로 경기 일정에 맞추어 CCN 코어 라우터로 전송한다. 경기가 끝나면 CCN 코어 라우터에 저장된 콘텐츠 정보는 삭제 가능하다.

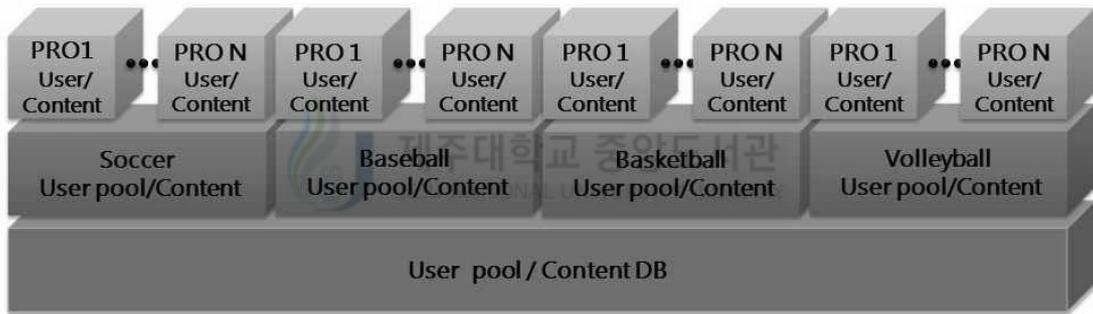


Fig. 55. Relational database structure of the sports mobile services platform

그림 56은 스포츠 모바일 플랫폼 관계형 데이터베이스의 실제 구성된 형태를 나타낸 것이다. 이는 CCN 환경에 효율적일 뿐만 아니라 프로 스포츠 구단, 서포터즈 등 여러 비즈니스 참여자들에게 효과적으로 제공하기에도 매우 효율적인 관리형 데이터베이스이다. 프로 스포츠의 범위를 축구, 야구, 농구, 배구 4가지 종목에 대해 접근하고 있으며, 전체 가입자 정보와 콘텐츠 항목을 구성한 후 4개의 스포츠에 대해 다시 나누었으며 이를 각 구단에 대하여 다시 가입자 정보와 콘텐츠 항목을 만들었다.

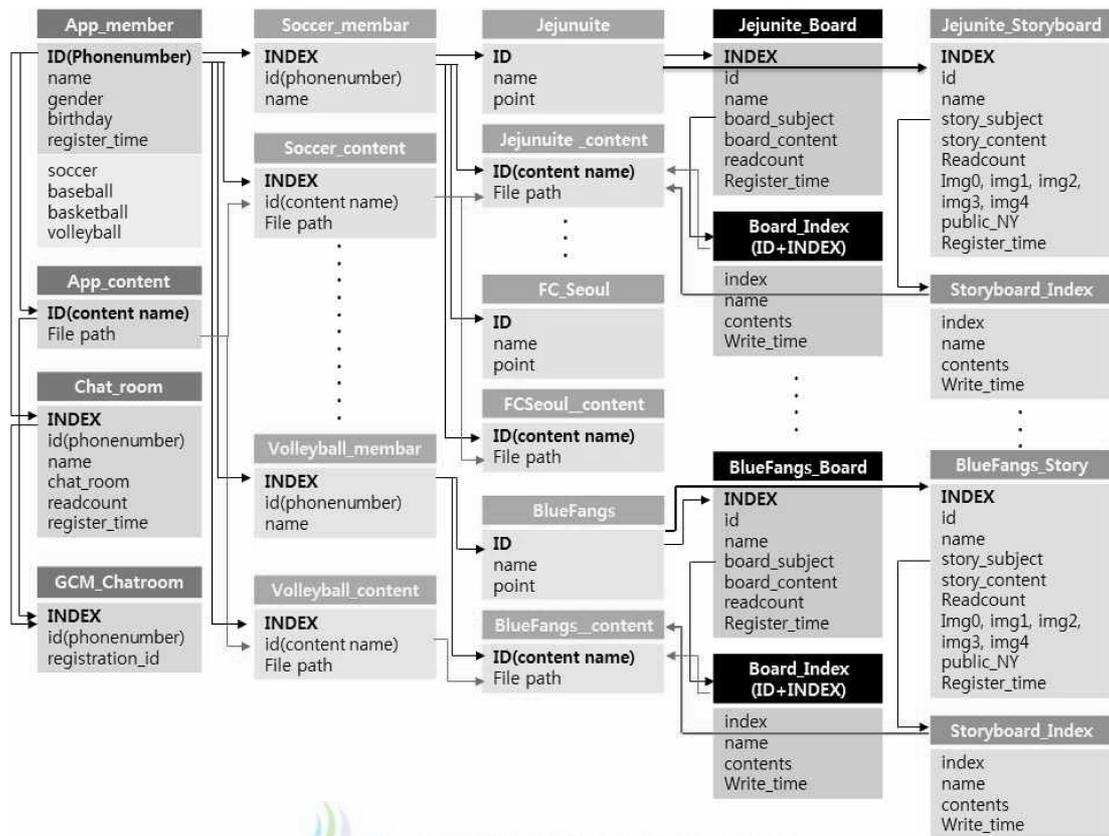


Fig. 56. Relational database of the sports mobile services platform

### 3. 스포츠 모바일 서비스 서버

스포츠 모바일 서비스 플랫폼에서는 사용자들의 정보공유 및 다양한 서비스를 제공하기 위해 게시판, 응원 스토리, 채팅방 등의 기능을 제공하기 위해 서버를 개발하였다.

그림 57은 로그인에 대한 동작 과정을 나타낸 것으로 Client가 서버로 로그인 화면 요청하면 클라이언트가 로그인 정보를 이용해 로그인을 승인한다. 사용자 정보를 서버를 전송하고 C2DM(cloud to device message)서버를 통해 registration ID를 부여받는다. 서버는 사용자 registration ID를 저장하며 스포츠 팀 선택 화면 요청

및 승인 요청에 대한 응답을 처리한다.

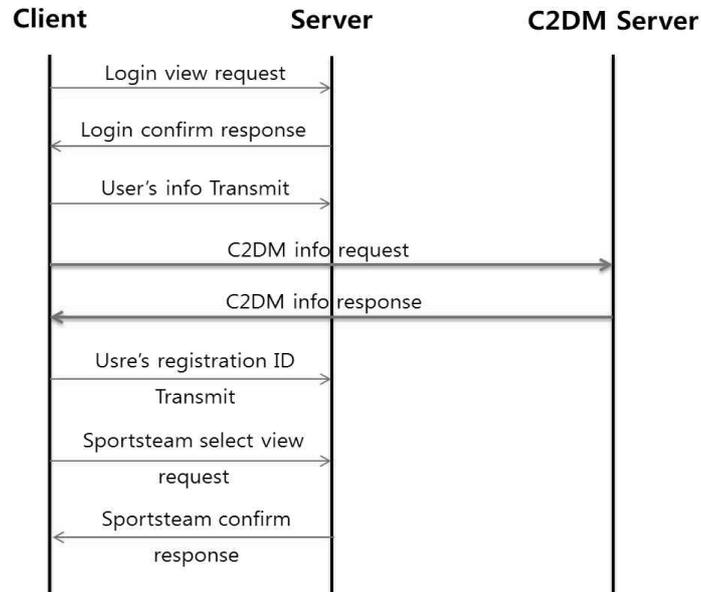


Fig. 57. Login UML sequence diagrams

그림 58은 포인트를 이용한 랭킹 서비스를 제공하는 것으로 사용자의 각 스포츠 팀별 포인트를 이용자의 회원가입, 응원스토리, 친구초대, 구매 등의 활동 현황에 따라 포인트를 부여하며 이를 사용자에게 랭킹으로 제공한다.

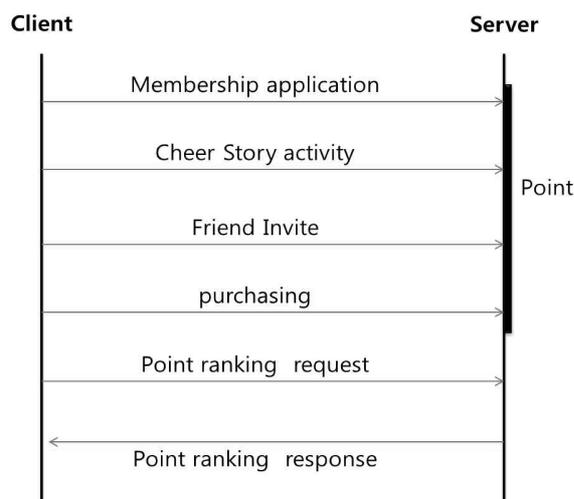


Fig. 58. Point UML sequence diagrams

그림 59는 게시판의 동작 과정을 UML 시퀀스 다이어그램으로 클라이언트는 서버에게 게시판 리스트를 요청하고 서버는 notice\_board DB이용 게시판 리스트를 처리하며 게시글을 선택하면 서버에게 정보를 요청하고 서버는 게시글의 내용과 댓글 정보를 제공하여 추가 댓글이 있는 경우 C2DM 서버를 이용하여 다른 클라이언트들에게 푸시 알림을 제공한다. 새로운 게시글이 작성되면 서버는 이를 DB에 저장하고, C2DM 서버를 이용하여 다른 클라이언트들에게 알림을 제공한다.

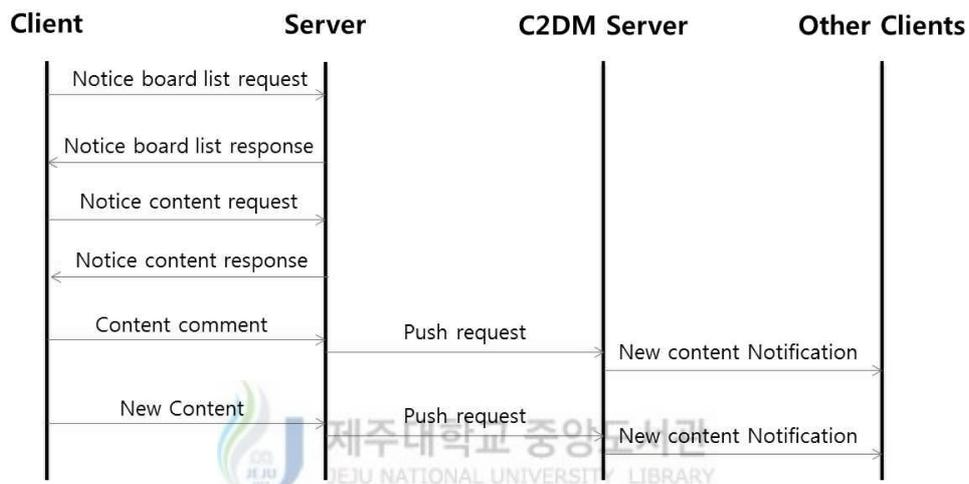


Fig. 59. Board UML sequence diagrams

그림 60은 응원 스토리의 UML 시퀀스 다이어그램으로 클라이언트는 서버에게 응원스토리 리스트를 요청하고 서버는 cheerstory DB이용 응원스토리 리스트와 이미지 파일을 제공한다. 서버와 App는 응원스토리의 내용과 댓글 정보를 제공 추가 댓글이 있는 경우 C2DM 서버를 이용하여 다른 클라이언트들에게 알림을 제공한다. 새로운 응원스토리를 작성 및 선택한 이미지를 서버로 전송하면 서버는 응원스토리는 DB에 저장하고 이미지 파일은 서버에 저장하고 새로운 응원스토리에 대해 다른 클라이언트들에게 알린다.

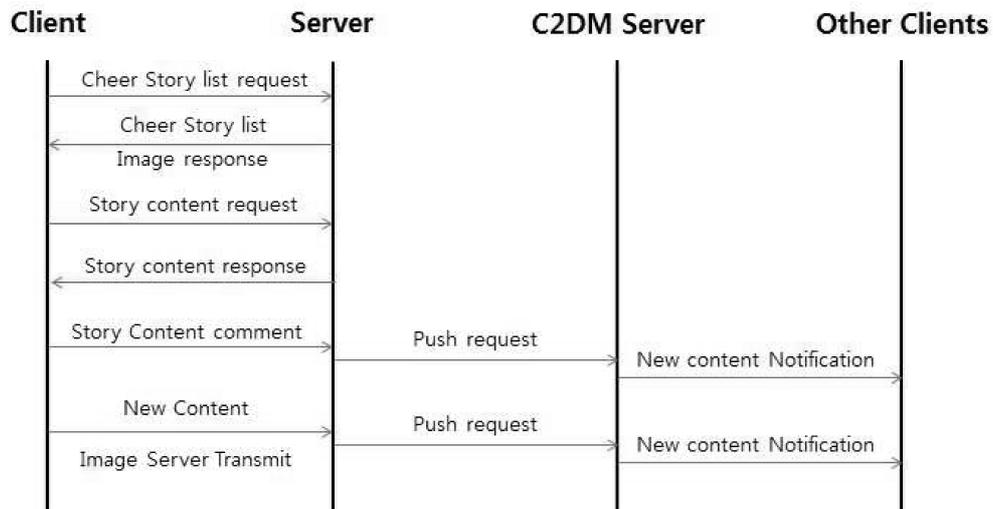


Fig. 60. Cheer story UML sequence diagrams

그림 61은 채팅의 UML 시퀀스 다이어그램으로 서버에게 채팅방 리스트 요청에 대한 화면을 제공한다. 채팅 친구 목록 요청을 통해 채팅 환경 구축 및 채팅방 생성 채팅 정보를 서버로 보내면 대상 클라이언트들에게 C2DM 서버를 통해 클라이언트들에게 알린다. 채팅 푸시 알림을 받은 클라이언트들은 동일한 방법으로 자신의 채팅방에 소속된 클라이언트들과 채팅이 이루어진다.

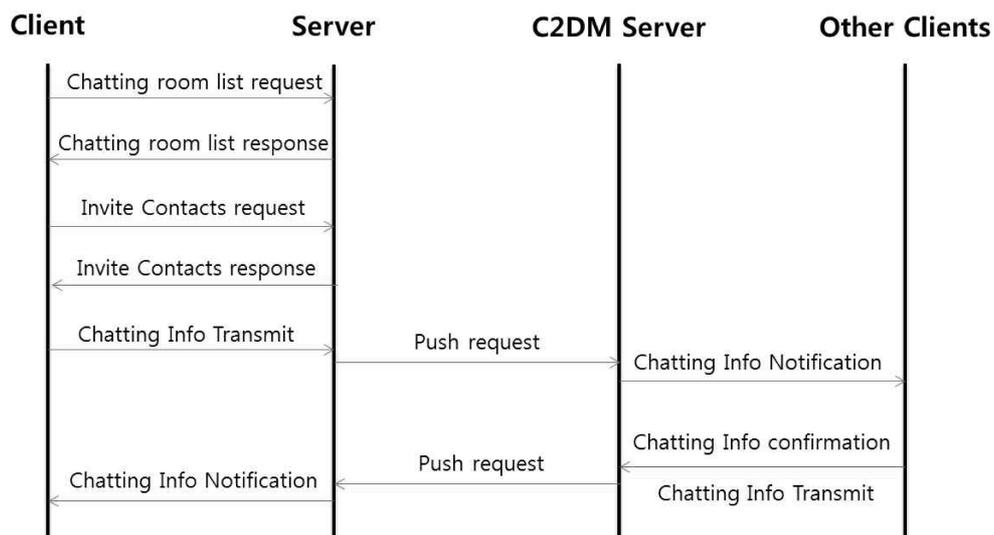


Fig. 61. Chatting UML sequence diagrams

#### 4. 스포츠 모바일 서비스 어플리케이션

그림 52는 초기화 설정 과정을 UML 클래스들의 다이어그램을 나타낸 것으로 사용자의 기본 정보를 이용해 사용자 목록 갱신을 수행하기 위해 pluginEvent를 이용하여 서버에 접속한다. 서버는 스마트 폰의 정보를 수집하고 추가 변경된 정보는 스마트 폰의 데이터베이스에 저장한다. 또한 사용자 정보를 서버로 전송하여 플러그인을 설정하고 접속에 대한 결과를 알려준다.



Fig. 62. Initial setup and main screen

그림 63은 스포츠 모바일 플랫폼 관계형 데이터베이스를 구축하기 위해 최초 사용자가 로그인을 통해 스포츠 종목 선택 및 구단 선택까지 사용자 정보를 입력받아 서버의 데이터베이스에 등록한다. 서버는 이 정보를 이용해 각 경기장의 경기 일정에 맞게 해당하는 사용자 정보를 제공한다

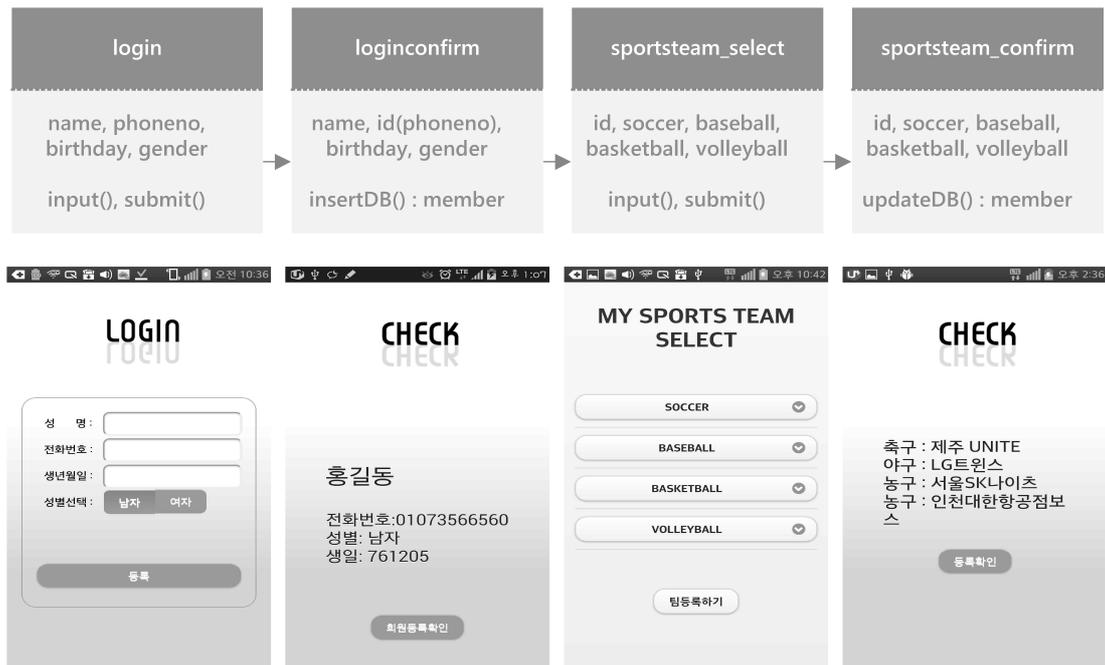


Fig. 63. Login Screen

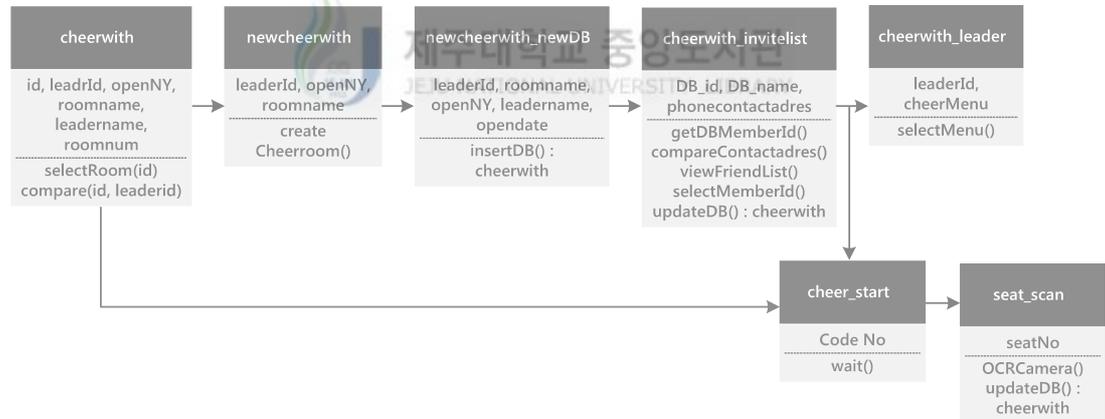


Fig. 64. UML diagrams for cheer group formation

본 논문에서는 콘텐츠 공유를 위해 응원그룹을 생성한다. 그림 64는 응원그룹 형성 및 관리에 대한 UML 클래스들의 다이어그램을 나타낸 것이다. 응원그룹을 새롭게 생성하여 응원그룹 리더로 단체응원을 진행할 수 있고, 응원그룹 멤버로 참여하고자 하면 응원그룹 리스트에서 응원그룹을 선택하거나 응원그룹 리더의 초대메시지로 응원그룹에 참여할 수 있다. 응원그룹 정보는 응원그룹에 가입한 경기장 내의 사용자

들이 동시에 콘텐츠를 실시간으로 공유할 수 있는 카드섹션과 동시 응원가 플레이 등의 응원 콘텐츠를 제공한다.

CCN의 코어 라우터에서 제공하는 콘텐츠 이름은 사용자가 쉽게 기억할 수 있는 이름도 있지만 사용자가 기억하고 입력하기 어려운 다소 복잡하고 긴 이름도 있다. 이에 본 논문에서는 사용자가 콘텐츠 이름을 직접 찾을 필요 없이 아이콘을 클릭만 하면 원하는 콘텐츠를 가져올 수 있게 하였다. 그림 65는 CCN 응원 그룹 형성, 응원 그룹 운영 및 초대, 응원 콘텐츠 정보 등을 사용자가 쉽게 접근할 수 있도록 아이콘으로 구성된 어플리케이션 화면이다.



Fig. 65. Cheer group leader screen

그림 66은 멀티캐스트 그룹에 사용자들이 동시에 콘텐츠를 제공받을 수 있는 화면을 나타낸 것이다.

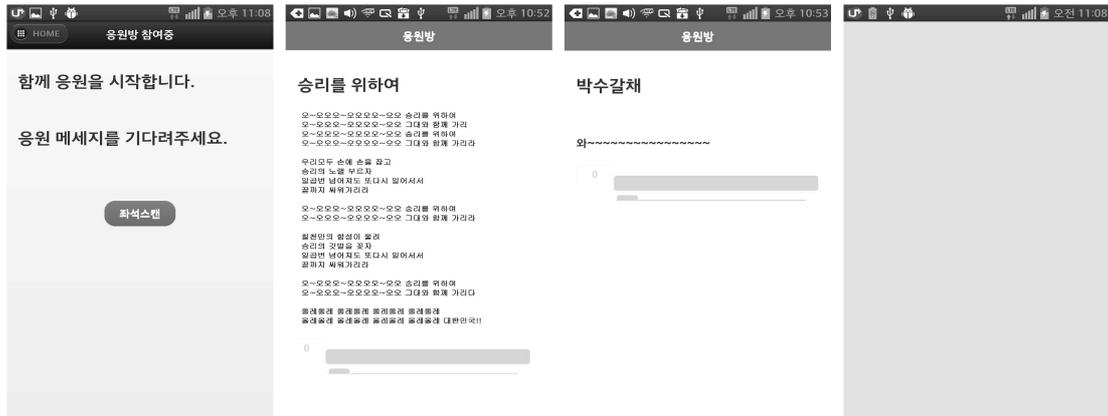


Fig. 66. Cheer group user screen

또한 스포츠 플랫폼 모바일에서는 사용자들의 정보공유 및 다양한 서비스를 제공하기 위해 게시판, 응원 스토리, 채팅방 등의 기능을 제공한다.



Fig. 67. Board screen

그림 67은 게시판의 구현 UML 클래스들의 다이어그램과 어플리케이션 화면으로 게시판은 사용자가 각 팀별 게시판에 게시글과 댓글을 작성할 수 있으며 새 게시글과 댓글 작성 시 서버는 새 글 알림을 푸시메시지를 통하여 구단에 속해 있는 사용자들에게 전송한다.

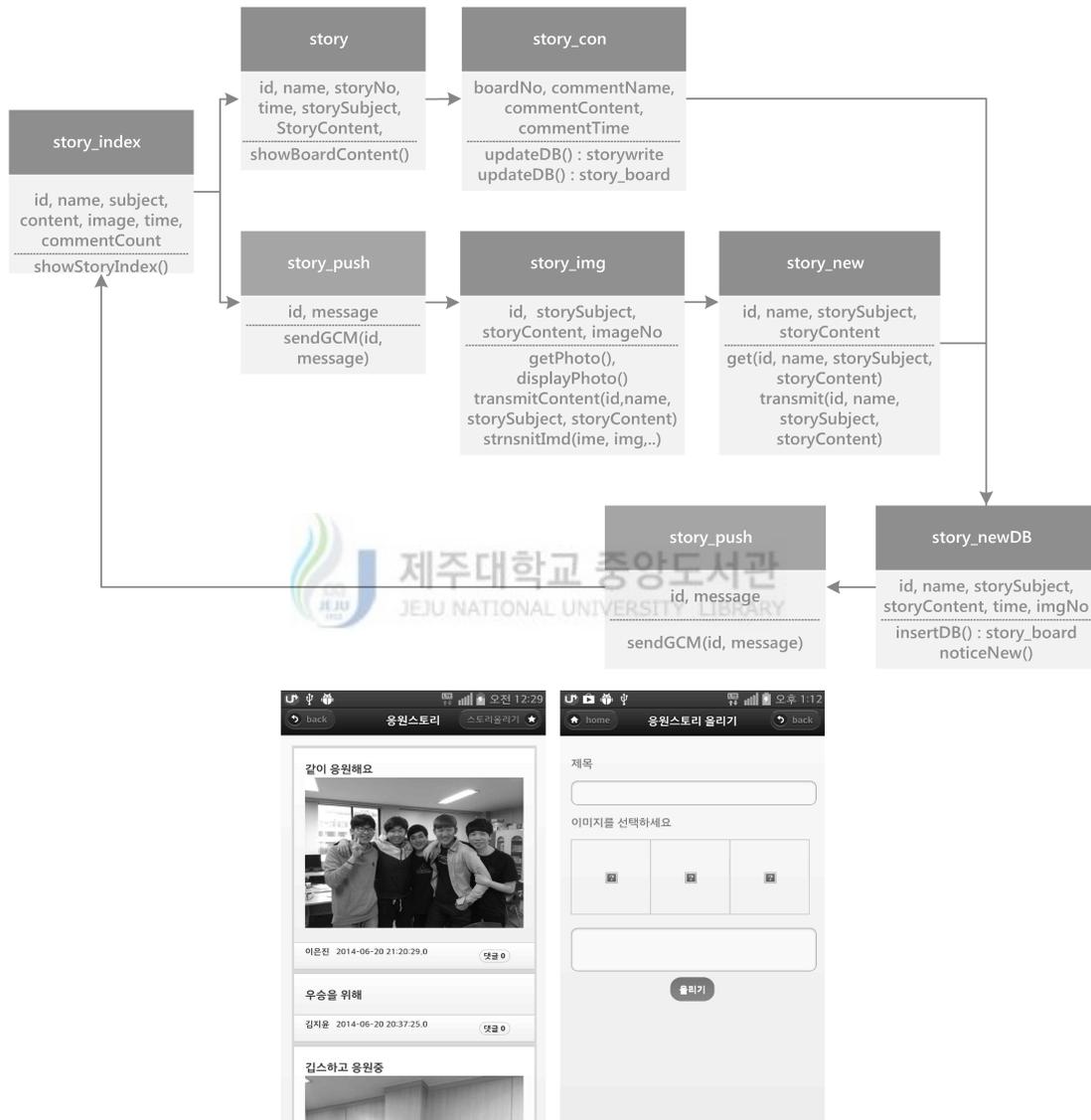


Fig. 68. Cheer story screen

그림 68은 응원스토리의 구현 UML 클래스들의 다이어그램과 어플리케이션 화면으로 응원스토리는 사용자가 각 팀별 응원스토리 게시판에 스토리와 댓글을 작성할

수 있으며, 새 스토리와 댓글 작성 시 서버는 새 글 알림을 푸시메시지를 통하여 구단에 속해있는 사용자들에게 전송한다.

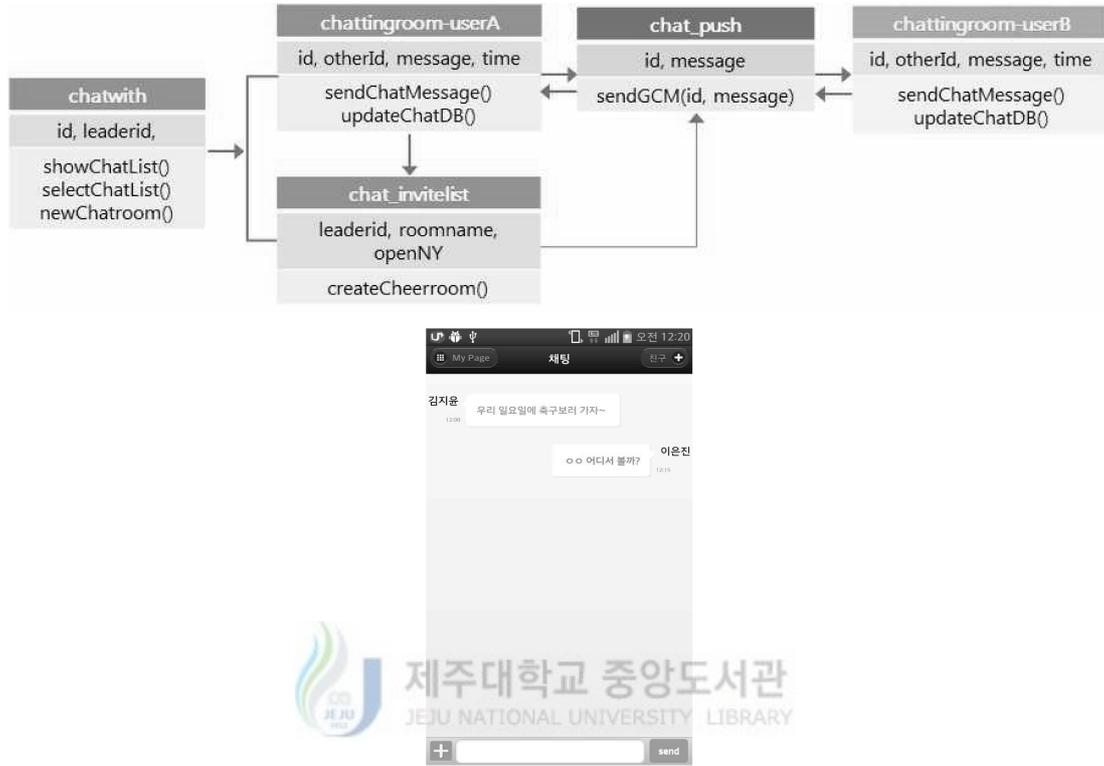


Fig. 69. Chatting screen

그림 69는 채팅방의 구현 UML 클래스들의 다이어그램으로 채팅방은 사용자가 스마트폰 내부에 저장된 채팅방리스트에서 채팅방을 선택하여 채팅을 다시 시작할 수 있고, 다른 친구를 초대하여 새로운 채팅방을 만들 수 있다. 채팅방에 새로운 대화글을 작성하면 폰내부 DB에 업데이트 하고 푸시메시지를 통하여 상대방에게 전달한다.

## 5. 스포츠 모바일 서비스를 위한 멀티캐스트 라우팅 프로토콜 구현

본 논문에서는 CCN에서 CMR-OSPFN 알고리즘을 사용하여 콘텐츠를 공유를 위해 응원그룹을 생성한다. 그림 70은 응원그룹 형성 및 관리를 위한 서버와 모바일 응원 어플리케이션의 동적인 관계를 나타내기 위해 UML 시퀀스 다이어그램을 이용해 나타낸 것이다. 클라이언트가 모바일 응원 어플리케이션을 설치하면 서버는 클라이언트의 로그인 정보를 데이터베이스에 저장하고 plugin을 이용하여 클라이언트의 정보를 수집한다. 또한 서버는 모바일 응원 어플리케이션의 운영을 위해 스마트폰에 데이터를 저장하며 모바일 응원 어플리케이션의 업데이터를 가능하게 하였다.

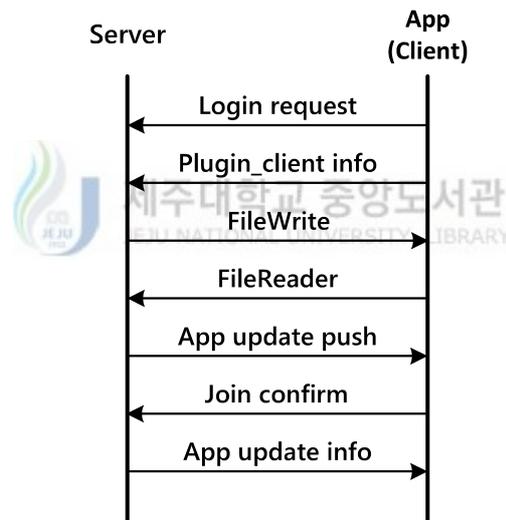


Fig. 70. UML sequence diagram of between server and App

그림 71은 서버와 CCN 코어 라우터의 UML 시퀀스 다이어그램으로 서버는 경기 시작 전에 CCN 코어 라우터로 경기가 있는 팀의 플랫폼 가입자 정보, 멀티미디어 콘텐츠 정보(콘텐츠 이름, 콘텐츠 life time), 데이터 파일을 전송한다. CCN 코어 라우터는 서버의 정보를 저장한다. 경기가 끝나면 CCN 코어 라우터는 콘텐츠 life time 이 설정된 콘텐츠를 삭제한다. 또한 서버는 경기가 끝난 후 코어 라우터가 수집한 정

보를 수신하기 위해 콘텐츠 요청 메시지를 CCN 코어 라우터로 전달한다. CCN 코어 라우터는 자신의 CS에 있는 콘텐츠를 서버로 전송한다.

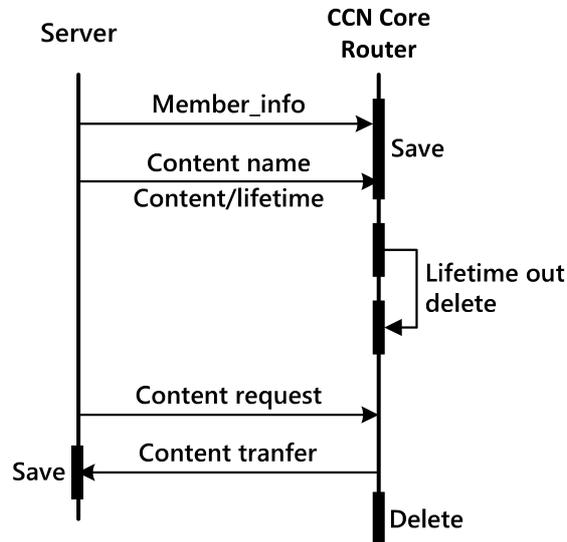


Fig. 71. UML sequence diagram of between server and CCN core router



그림 72은 경기장 내에서 CCN 코어 라우터, CCN 라우터, 모바일 응원 어플리케이션(응원그룹 운영자와 응원그룹의 클라이언트)의 UML 시퀀스 다이어그램으로 모바일 응원 어플리케이션의 시작을 통한 스포츠 모바일 서비스 플랫폼에 가입된 사용자인지를 확인한다. 응원그룹에 소속된 클라이언트들은 응원그룹을 선택하여 콘텐츠 수신 대기 상태를 유지한다. 응원그룹 운영자는 응원그룹에게 전송할 콘텐츠를 선택하여 CCN 코어 라우터로 요청한다. 이를 수신한 CCN 코어 라우터는 콘텐츠 이름 정보를 이용하여 멀티캐스트 그룹에게 콘텐츠를 전달한다. 이는 CCN 라우터를 거쳐 응원그룹의 클라이언트 모바일 응원 어플리케이션으로 전송되고 모바일 응원 어플리케이션은 이를 자동 실행하여 경기장 내 응원그룹이 동시에 콘텐츠를 플레이시킬 수 있게 하였다.

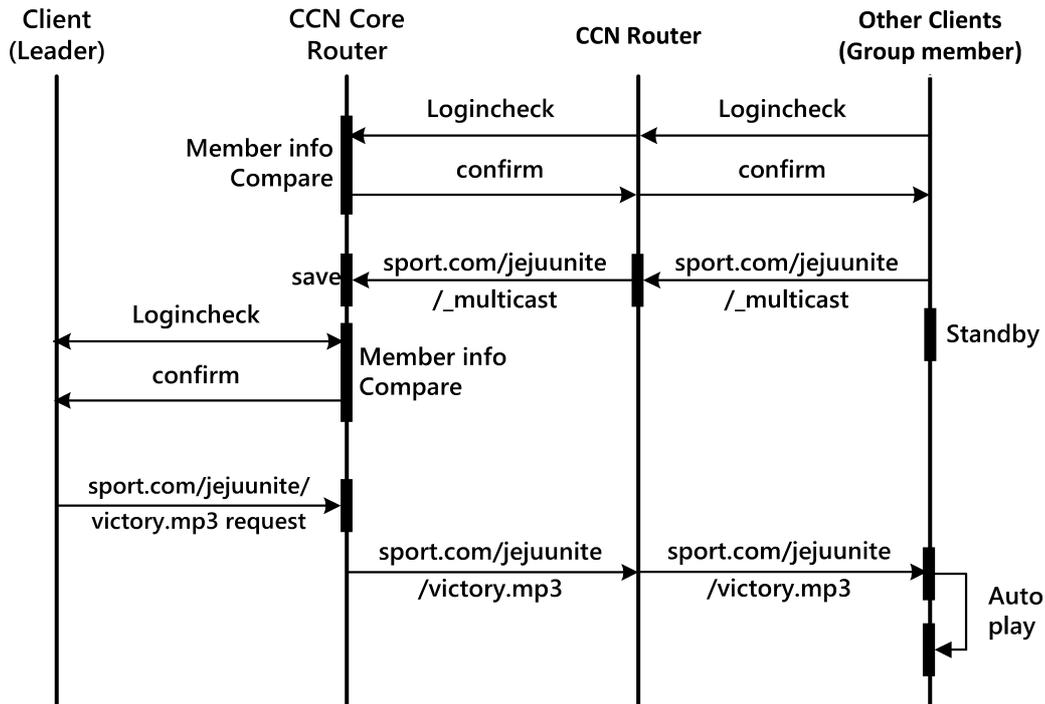


Fig. 72. UML sequence diagram for group communication



## 6. 고찰

본 논문에서는 스포츠 경기장에서 응원 그룹 서비스를 제공하기 위해 제안하는 CMR-OSPFN 알고리즘을 사용한 스포츠 모바일 서비스 플랫폼과 모바일 응원 어플리케이션을 개발하였으며, 모바일 응원 어플리케이션의 응원그룹 기능을 통하여 제안하는 CMR-OSPFN 알고리즘의 효율적인 캐싱 기법과 데이터 전달 방식을 설계하였다.

경기장 내 CCN 환경은 클라이언트들이 동시간에 콘텐츠를 공유하기 위해 네트워크 기기간 계층적 구조를 형성하였다. 이 구조는 경기장 전체를 관리하는 CCN 코어 라우터, CCN 코어 라우터 하위 계층인 동일 홉을 갖는 CCN 라우터, 마지막 계층으로 클라이언트를 갖도록 형성되었다. 이는 본 논문에서 제안한 CCN 멀티캐스트 그룹 생성을 위한 구조이며, CCN 코어 라우터와 CCN 라우터에 FIB 라우팅 테이블을

형성하여 경기장 내 응원그룹에 대한 멀티캐스트 그룹을 형성할 수 있게 하였다. 제안하는 CMR-OSPFN 알고리즘과 CBT 알고리즘에 대한 시뮬레이션을 통한 성능 분석 결과 제안하는 CCN에서 CMR-OSPFN 알고리즘의 네트워크 내 대역폭 사용량은 그림 42에서와 같이 25% 감소하였다. 이를 토대로 경기장 내 밀집된 사용자들 간의 실시간 응원 정보 전송이 가능한 CCN 환경을 구축하였다.

CCN에서 멀티캐스트 라우팅 테이블 형성을 위해 클라이언트들이 응원 어플리케이션을 통해 응원그룹 참가 버튼을 클릭하면 sport.com/응원그룹/\_multicast의 멀티캐스트 이름 접두사가 CCN 라우터와 CCN 코어 라우터의 라우팅 테이블에 저장되도록 하였다. 응원그룹의 운영자가 콘텐츠 이름에 해당하는 콘텐츠 아이콘을 클릭하면 콘텐츠 이름(sport.com/jejuunite/victroy.mp3)과 멀티캐스트 그룹(jejuunite)이 포함된 멀티캐스트 요청 패킷이 CCN 코어 라우터로 전송되며, CCN 코어 라우터는 콘텐츠 이름을 이용해 자신에게 저장된 콘텐츠를 멀티캐스트 그룹에게 전달하는 프로세서를 설계하였으며, 이 설계된 프로세서가 정상적으로 동작하는 것을 확인하였다.

경기 진행 중 경기장 내에 트래픽 사용량이 많아지므로 응원그룹에게 제공하는 콘텐츠를 서버로 요청할 경우 전송지연이 발생한다. 본 논문에서는 멀티캐스트 그룹 통신이 가능한 CCN에서 콘텐츠 서비스 시간 단축을 위해 제안한 휘발성 콘텐츠를 이용한 효율적인 캐싱 기법을 적용하였다. CCN 코어 라우터는 경기 시작 전에 서버가 전송하는 경기와 관련된 콘텐츠 이름, 콘텐츠, 콘텐츠의 라이프타임이 포함된 콘텐츠의 데이터 패킷을 저장하는 기능을 가진다. 또한 CCN 코어 라우터는 콘텐츠의 라이프타임을 이용하여 콘텐츠를 삭제하는 기능을 가지도록 설계하였다. CCN 코어 라우터는 응원그룹 운영자가 요청하는 응원 콘텐츠를 서버로 요청하지 않고 자신에게 저장된 콘텐츠를 전송하므로 서버에 집중되는 네트워크 내의 트래픽 사용량과 상관없이 콘텐츠를 제공함으로써 전송지연의 문제가 발생하지 않는다.

## VI. 결론

본 논문에서는 미래 인터넷의 주요 핵심 기술로 주목 받고 있는 콘텐츠 중심 네트워크 기술인 CCN에서의 사용자가 밀집된 환경에서 그룹 통신을 위해 효율적인 멀티캐스트 라우팅 알고리즘을 제안하고, 이를 사용한 새로운 데이터 전송 방식을 제안하였다. 제안하는 CMR-OSPFN 알고리즘은 코어 라우터와 노드가 콘텐츠 이름 접두사와 그룹 이름 접두사를 네트워크로 플로딩하기 위해 OSPFN의 OLSA 메시지를 수정하였다. 코어 기반 멀티캐스트 라우팅을 제공하기 위해 그룹 이름과 코어라우터의 정보는 FIB 테이블을 수정하여 저장하였다. 또한 기존 CCN의 메시지 패킷 내에 삽입된 TOS 필드를 사용하여 새로운 데이터 전송 방식을 제안하였다. 제안된 CMR-OSPFN은 사용자의 수가  $N_u$ 이고, 요청 패킷과 데이터 패킷의 전달시간이 각각  $T_I$  와  $T_D$  일 때 사용자가 밀집된 환경에서 평균 콘텐츠 서비스 시간을  $N_u T_I + N_u T_D$  에서  $T_I + N_u T_D$ 로 줄일 수 있다. CMR-OSPFN 알고리즘을 사용한 새로운 데이터 전송 방식의 성능을 분석하기 위해 NS-2 시뮬레이터를 사용하여 CBT와 비교하였다. 콘텐츠 제공자와 코어 라우터 사이의 홉 수를 변화시키면서 동일한 콘텐츠를 반복적으로 요청할 경우 CMR-OSPFN과 CBT의 콘텐츠 전달 시간과 네트워크 대역폭 사용량을 측정하였다. 측정 결과 홉 수가 10일 때 CBT에 비해 제안하는 CMR-OSPFN의 평균 콘텐츠 전달 시간이 최대 41% 향상되었고, 평균 대역폭 사용량은 최대 74% 줄어들었다.

본 논문에서는 CCN 환경에서 사용자가 밀집된 그룹 통신을 위한 효율적인 캐싱 기법과 새로운 데이터 전달 방식을 제안하였다. 사용자가 밀집된 네트워크에서의 그룹 통신은 서버에 트래픽 집중으로 병목현상을 발생시켜 전송 지연이 발생된다. 기존 CCN의 캐싱 기법은 LRU와 LFU로 각각 최근 사용된 콘텐츠와 자주 사용된 콘텐츠를 캐시에 저장하는 방식으로 이는 사용자가 밀집된 CCN 환경에서 그룹 통신을 위해서는 일정 시간 서비스를 제공하는 경우에는 적합하지 않다. 본 논문에서 제안된 캐싱 기법은 콘텐츠 제공자로부터 라이프타임 필드가 추가된 콘텐츠를 전달받아, 캐시에 저장하고 일정 시간 서비스한 후 라이프타임이 종료되면 삭제하는

방식이다. 제안된 코어 라우터 기반의 캐싱 기법의 평균 콘텐츠 서비스 시간은 코어 라우터와 서버 사이의 홉 수가  $N_{SR}$  일 때  $N_{SR}T_I$  만큼 줄일 수 있다. 제안하는 캐싱 기법의 성능을 분석하기 위해 CMR-OSPFN의 CCN 코어 라우터에 제안하는 캐싱 기법을 적용한 경우와 적용하지 않은 경우에 대해 평균 콘텐츠 서비스 시간을 측정하였다. 측정 결과 콘텐츠 요청 수에 따른 평균 콘텐츠 서비스 시간을 비교하였을 때 제안하는 캐싱 기법의 경우 평균 콘텐츠 서비스 시간이 26% 향상되었다. 콘텐츠가 N개의 데이터 패킷으로 구성된 경우, 기존 CCN에서는 하나의 요청 패킷에 대해 하나의 데이터 패킷을 전달하는 OROP 방식을 사용하므로, 콘텐츠를 전달하기 위해서는 N개의 요청 패킷이 필요하다. 본 논문에서 제안하는 데이터 전달 방식은 하나의 요청 패킷에 대해 N개의 데이터 패킷을 전송하는 OROC 전달 방식이다. OROC는 OROP에 비해 평균 콘텐츠 서비스 시간을  $NT_I+NT_D$ 에서  $T_I+NT_D$ 로 줄일 수 있다. OROC의 성능을 분석하기 위해 OROP와 비교하였으며, 콘텐츠의 크기를 변화시켜 평균 콘텐츠 서비스 시간을 측정한 결과 OROC가 사용하였을 때 평균 콘텐츠 서비스 시간이 최대 55% 향상되었다.

제안하는 CMR-OSPFN 알고리즘의 캐싱 기법과 데이터 전달 방식은 스포츠 경기장과 같은 사용자가 밀집된 네트워크 환경에서 사용할 수 있다. 본 논문에서는 스포츠 경기장에서 응원 그룹 서비스를 제공하기 위해 제안하는 CMR-OSPFN 알고리즘을 사용한 스포츠 모바일 서비스 플랫폼과 모바일 응원 어플리케이션을 개발하였다. 응원 그룹 운영을 위해 CCN 코어 라우터는 경기 시작 전에 서비스 서버로부터 경기 관련 콘텐츠를 전달받아, 경기가 시작되면 서비스하고, 경기가 끝나면 라이프타임 후에 삭제할 수 있도록 설계하였다. CCN 코어 라우터는 응원그룹 운영자가 요청하는 응원 콘텐츠를 서버로 요청하지 않고 자신에게 저장된 콘텐츠를 전송하므로 서버에 집중되는 네트워크 내의 트래픽 사용량과 상관없이 콘텐츠를 제공함으로써 전송 지연의 문제가 발생하지 않는다.

향후 본 논문에서 제안하는 CCN 멀티캐스트 라우팅 및 캐싱 기법을 이용하여 학교 강의실에서 수업 진행 및 관리, 회의장에서의 국제회의 진행 및 관리, 공연장에서 공연 진행 및 관리 등 사용자가 밀집된 다양한 환경에서 그룹 통신에 효율적으로 사용될 수 있을 것이다.

## 참고문헌

- [1] 신명기, “미래인터넷 기술 및 표준화 동향”, 전자통신동향분석, 제 22권 제 6호, 한국전자통신연구원, 2007, pp. 116-128.
- [2] 김도현.이영희, “미래인터넷 서비스 기술”, 한국통신학회지 정보와 통신, 제 25권 제 3호, 한국통신학회, 2008, pp. 11-20.
- [3] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs and R. L. Braynard, "Networking named content," in Proceedings of the 5th International Conference on Emerging Networking Experiments and Technologies, 2009, pp. 1-12.
- [4] 장명욱, “CCN 기반 SNS 플랫폼 구축”, 한국통신학회 정보와 통신, 제 30권 제 3호, 한국통신학회, 2013, pp. 60-67.
- [5] V. W. Luk, A. K. Wong, C. T. Lea and R. W. Ouyang, “RRG: Redundancy Reduced Gossip Protocol For Real-Time N-to-N Dynamic Group Communication”, Journal of Internet Services and Applications , 2013, pp. 1-19.
- [6] Kenneth Hopkinson, Kate Jenkins, Kenneth Birman, James Thorp, Gregory Toussaint, and Manu Parashar, "Adaptive Gravitational Gossip: A Gossip-Based Communication Protocol with User-Selectable Rates," IEEE Transactions on Parallel and Distributed Systems, Vol. 20, No. 12, December 2009, pp. 1830-1843.
- [7] Vincent Wing-Hei Luk, Albert Kai-Sun Wong, Chin-Tau Lea, “Peer-to-Peer Real-Time Group Communication over Content-Centric Network”, 2012 IEEE 13th International Conference, IEEE, 2012, pp. 109-114.
- [8] Zhenkai Zhu, “ACT: Audio Conference Tool Over Named Data Networking”, ICN '11, Information-centric networking, 2011, pp. 68-73.
- [9] L. Wang, A K M Mahmudul Hoque, Cheng Yi, Adam Alyyan, Beichuan

- Zhang, “OSPFN:An OSPF Based Routing Protocol for Named Data Networking”, NDN Technical Report-0003, 2012.
- [10] Tom Billhartz, J. Bibb Cain, Ellen Farrey-Goudreau, Doug Fieg, and Stephen Gordon Batsell, “Performance and Resource Cost Comparisons for the CBT and PIM Multicast Routing Protocols”, IEEE Journal on Selected Areas in Communications, Vol. 15, No. 3, April, 1997, pp. 304-315.
- [11] WANG Guoqing, HUANG Tao, LIU Jiang, CHEN Jianya, LIU Yunjie, “Approximate Models for CCN Data Transfer in General Topology”, China Communications IEEE, Vol. 11 Issue 7, July. 2014, pp. 40-47.
- [12] Cisco Visual Networking Index: Forecast and Methodology, 2013-2018, [http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white\\_paper\\_c11-481360.html](http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white_paper_c11-481360.html)
- [13] 최시중, “미래를 대비한 인터넷 발전전략”, 한국IT서비스산업협회, 2011.
- [14] 김안빈, 이동통신망에서 콘텐츠 중심 네트워킹 기반의 멀티미디어 서비스 품질 지원, 한국외국어대학교 석사학위논문, 2013.
- [15] T. Koponen et al, “A Data-Oriented (and Beyond) Network Architecture,” SIGCOMM ‘07, Applications, technologies, architectures, and protocols for computer communications, 2007, pp. 181-192.
- [16] Lixia Zhang, et al., “Named Data Networking”, ACM SIGCOMM Computer Communication Review, vol 44 Issue 3, July 2014.
- [17] OC-3 전용선 비용 통계, <http://dri-korea.com/doc/202>.
- [18] 메모리 가격 통계, <http://www.jcmit.com/memoryprice.htm>.
- [19] R. JAYAKUMAR, K. THULASIRAMAN, AND M. N. S. SWAMY, “Complexity of Computation of a Spanning Tree Enumeration Algorithm” IEEE Transactions on Circuits and Systems, Vol. 31, No. 1, October, 1984, pp. 853-860.
- [20] Lixia Zhang, et al., Named Data Networking (NDN) Project, NDN

Technical Report NDN-0001, Oct. 2010.

- [21] Jihoon Lee, Daeyoub Kim, "Proxy-assisted Content Sharing Using Content Centric Networking (CCN) for Resource-limited Mobile Consumer Devices", IEEE Transactions on Consumer Electronics, Volume 57 No. 2, may, 2011, pp. 477-483.
- [22] Longzhe Han, Seung-Seok Kang, Hyogon Kim, and Hoh Peter In, "Adaptive Retransmission Scheme for Video Streaming over Content-Centric Wireless Networks", IEEE Communications Letters, Vol. 17, No. 6, June 2013, pp. 1292-1295.
- [23] S.Shanbhag, N.Schwan, I.Rimac, M.Varvello, "SoCCeR:Service over content-centric routing" ICN '11, Information-centric networking, 2011, pp. 62-67.
- [24] Sadd Ghaleb Yaseen, Nada M.A. AL-Slamy, "Ant Colony Optimization" IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.6, June 2008, pp. 351-357.
- [25] J. Moy, "OSPF Version 2", IETF RFC 2328, April 1998.
- [26] Chengming LI, Wenjing LIU, Koji OKAMURA, "A Greedy And Colony Forwarding Algorithm for Named Data Networking", 2012 APAN, Asia-Pacific Advanced Network, 2012.
- [27] Reaz Ahmed, et al., " $\alpha$ Route:A Name Based Routing Scheme for Information Centric Networks", INFOCOM 2013, IEEE, 2013, pp.14-19.
- [28] Kari Visala, Dmitriy Lagutin, Sasu Tarkoma, "LANES:An Inter-Domain Data-Oriented Routing Architecture", ReArch '09, Re-architecting the internet, 2009, pp. 55-60.
- [29] 고석주, 박정수, 김광수, 김용진, "인터넷 멀티캐스트 라우팅 프로토콜 분석", 한국전자통신연구원 전자통신동향분석, 제14권 제5호, 1999, pp. 99-110.
- [30] D. Waitzman, C. Partridge, S. Deering, "Distance Vector Multicast Routing Protocol", RFC 1075, 1988.
- [31] J. Moy, "Multicast Extension to OSPF", IETF RFC 1584, 1994.

- [32] D. Estrin, D. Farinacci, V. Jacobson, C. Liu, L. Wei, P. Sharma, A. Helmy, "Protocol Independent Multicast-Dense Mode(PIM-DM): Protocol Specification", Internet Draft, January 1996.
- [33] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C. Liu, L. Wei, P. Sharma, "Protocol Independent Multicast - Sparse Mode(PIM-SM): Protocol Specification", IETF RFC 2117, June 1998.
- [34] Thaler, D.G. and Ravishankar, C. V., "Distributed Center-Location Algorithms", IEEE Journal on Selected Areas in Communications, Vol. 15, No. 3, pp. 291-303, April 1997.
- [35] Calvert, K. L., Zegura, E. W., and Donahoo, M. J., "Core Selection Methods for Multicast Routing", Proc. Int. Conf. On Computer Communications and Networks, pp. 638-642, July 1995.
- [36] PARC. CCNx open source platform. <http://www.ccnx.org>.
- [37] IBM lotus sametime, <http://www-01.ibm.com/software/lotus/sametime>.
- [38] Google talk, <http://www.google.com/talk/about.html>.
- [39] Skype Limited Skype video call, <http://www.skype.com/intl/en-us/features/allfeatures/video-call>.
- [40] Polycom telepresence solutions, [http://www.polycom.com/products/telepresence\\_video](http://www.polycom.com/products/telepresence_video).
- [41] 박성철, 이승엽, 박영률, "스마트 TV", 커뮤니케이션북스(주), 2007.
- [42] 김 바 울, 탕 지 아 메 이, 광 초 록, 김 상 욱, "콘텐츠 공유 프레임워크 기반의 다중 디바이스 간 협업 시스템", 정보과학회논문지 : 컴퓨팅의 실제 및 레터, 제 19권 12호, 2013, pp. 618-623.
- [43] 굴람 사와, 파만 올라, 윤 장 우, 이 성 창, "N-스크린 기반 인터랙티브 멀티 비전 시스템 제어 구조", 전자공학회 논문지, 제 50권 6호, 2013, pp. 1388-1397.
- [44] 이은진, 박세훈, 이소영, 김현경, "캠퍼스 CEO 발굴 지원 사업 스포츠 모바일 서비스 플랫폼 개발", 미래창조과학부, 2014.
- [45] 강민구, 홍석찬, 이재훈, 윤병준, 김동오, "N-스크린 미러링용 화면 공유 및 제어

플랫폼 설계”, 2012년도 한국인터넷정보학회 추계학술발표대회 논문집, 제 13권 2호, 2012, pp. 157-158.

- [46] 이동희, 노삼혁, 민상렬, 조유근, “LRFU:무한 참조 정보를 이용하는 블록 교체 기법”, 한국정보과학회, 정보과학회논문지, 제24권 제7호, 1997, pp.632-641.



## 감사의 글

감사합니다. 이 감사한 마음을 어떻게 몇 줄의 글로 표현할 수 있을까요? 그럼에도 모든 분들께 감사드립니다. 부족한 저를 항상 아껴주시고 잘 이끌어 주신 김흥수 교수님 감사드립니다. 교수님의 가르침 잊지 않고 살아가겠습니다. 건강하세요 교수님. 바쁘신 중에 논문 봐주시며 늘 따뜻한 말로 격려해주신 임재운 교수님, 논문을 쓰는 동안 막막한 순간 많은 도움을 주시고 직접 펜을 들고 마지막까지 방향을 잡아 주신 좌정우 교수님, 논문 이 곳 저 곳을 꼼꼼히 봐주신 윤종호 교수님, 제가 할 수 있는 이상을 할 수 있다 믿어주시고 응원해주시는 김도현 교수님. 감사합니다. 그리고 늘 아무 말씀 없이 곁에서 힘이 되어주시던 강진식 교수님. 논문을 쓰는 동안 교수님의 빈자리가 너무나 크다는 것을 느꼈습니다. 스승의 가르침 잊지 않겠습니다. 항상 격려를 아끼지 않으시고 늘 든든한 지원군이 되어주신 김순환 교수님, 자기자리에서 최선을 다하는 사람이 되라는 가르침과 학자의 길이 무엇인지를 보여주시는 양두영 교수님, 늘 자신의 일에 최선을 다하며 통신공학과 후배들의 롤 모델 현승엽 교수님 감사합니다.

영원한 나의 선배님들, 홍성욱 선배님, 이권익 선배님, 강봉수 오빠 저 졸업합니다. 성욱 선배님은 제가 감사의 글을 써야하는 이유인 것 같습니다. 감사드려요. 성욱 선배님의 멋진 반쪽 형미 언니 고맙습니다. 바쁘신 와중에도 후배의 어리광 받아 주시고 시간 내어주신 권익 선배님, 연구실 최근 논문을 작성하신 봉수 오빠 논문 잘 봤습니다. 내 절친이자 멋진 후배 은정아, 지윤아 너희들이 있어 이 긴 시간 잘 견뎠어. 고맙고 사랑한다.

이 길을 포기하지 않고 여기까지 올 수 있게 늘 곁에서 힘이 되어주시고 제가 한 없이 작아질 때 늘 저를 최고라 말씀해주시며 저의 삶의 매 순간을 저보다 더 기뻐하고 슬퍼해주신 내 소중한 부모님 감사합니다. 영원한 내편이자 멋진 남편 김민균, 나의 아들 김승건 힘든 시간 견뎌줘서 고맙고 감사합니다. 그리고 사랑합니다. 아버님, 어머님 부족한 며느리 늘 이쁘게 봐주셔서 감사합니다. 아버님의 눈물 잊지 않겠습니다. 아주버니, 민정언니, 효정언니, 고지혜, 고지훈 감사합니다. 나의 멋진 동생들 은경+α, 지혜와 제부, 원이와 수정이 고맙다. 우리 준우 고모가 사랑한다.



제주대학교 중앙도서관  
JEJU NATIONAL UNIVERSITY LIBRARY