The Effect of Cache Partitioning and Sharing on Named Data Network

Leanna Vidya Yovita, Nana Rachmana Syambas, and Ian Yosef Matheus Edward

Abstract— **The IP network is inefficient to support data communication. The IP network modifications into a Content Distribution Network scheme can not accommodate consumer mobility and consumer demand patterns that is change dynamically. So that, Named Data Network has been proposed to focus on content for data communications. The Named Data Network router comes with a content store for storing data. The content store size is limited so special techniques are required to optimize content store usage and keep NDN performance in good level. In the previous study, the majority choose data to be stored in content stores based on content popularity only. Popular data will be prioritized to keep in content store and the data with less popularity will be deleted from content store when it is full. This basic technique is known as 'sharing-technique'. Sharing-technique is difficult to differentiate the treatment of the data with more than one classification requirement. In fact, it is needed to differentiate the treatment for different requirements classes, not only the content priority. In this research, simulation is conducted to analyze the influence of content storage policy with sharing-techniques and partitioning-techniques. The partitioning-technique accommodates data storage that considers to the different class of content and also popularity of the content. The results of the simulation show that partitioning-technique gives maximum 36% more cache hit ratio than sharing-technique, viewed in network level. Partitioning-technique allows flexibility of class-based settings for content based on the proportion of content store, while in the sharing-technique the difference in cache hit rate for content classes is not significant.**

Keywords—partition, sharing, content store, Named data Network

I. Introduction

THE data exchange using IP network is inefficient. In the IP network, the user will request the required data to a certain server specifically using the IP address [1]. Therefore the response to the user request can be done by a particular server only. Repeated requests from users for the same data will result in data traffic overloading the network. Modifications have been made with the Content Distribution Network (CDN) [2], where a replica server is created which stores a copy of the data from the primary server. The replica server is placed closer to the user. The user request for a data will be responded by this replica server. However, temporary data storage methods on the CDN cannot support user mobility and dynamically change user demands on a data.

In 2009, Jacobson et al. [3] proposed a new content-based network paradigm. This concept replaces the 'where' paradigm into the 'what' paradigm. In this paradigm, the user request is no longer addressed to a particular IP but is intended for a particular content [3], [2], [4]. This concept causes the response to the content requested by the consumer not only can be served by a particular server, but it also can be served by the nearest device that has the requested data [5]. To support this concept, the NDN router nodes are equipped with a content store to store data.

The nodes mobility will be supported because content that is kept in the content store (CS) can be customized to fit the user request pattern on content. The changes of user position cause the content stores have to adjust their CS contents to suit the user requests in the local area. Data storage on NDN is more dynamic than the IP network architecture. CS has limited resources on NDN routers. The size of the content store affects the delay and number of hops that the packets have to through to the consumer [6] and this condition affect the overall network load due to the circulation of data on the network [5], [2]. Previous studies mostly select the data to be saved on the content store by considering the popularity of the content [7], [8], [9] or predicted that the content will provide the expected target value [10], [11]. The data storage setting on CS majority is based on the popularity of a content. Less popular content is not prioritized for saved in CS. In fact nowadays there are various types of services that have different requirements, such as file sharing services, real-time entertainment, web page search, social networking, real-time communication, and more [12]. A content service may be less popular but requires low delay so it should still be considered to be stored on CS. These classes have to accommodated to support NDN performance.

In previous studies, sharing-technique is implemented in content store as a techniques to kept a popular data [13][3][7]. The less popular data in CS will be replaced with more popular data, when CS is full. Sharing-techniques is easy to use with only one classification requirement, for example: popularity. It is need the other technique to accommodate more than one classification requirement, for example: class of content and popularity. So that, partitioning-technique is proposed to accommodate more than one classification requirement.

In this research it is conducted the testing to analyze the effect of sharing-techniques and partitioning-techniques to

Manuscript received June 15, 2018. L. V. Yovita, PhD student at School of Electrical Engineering and Informatics, Bandung Institute of Technology, (e-mail: leanna@telkomuniversity.ac.id). N.R. Syambas, Professor, School of Electrical Engineering and Informatics, Bandung Institute of Technology, 10 Ganesha Road, Lb. Siliwangi, Coblong, Bandung 40132, West Java, Indonesia (e-mail: nana@stei.itb.ac.id). Ian Yosef M.E., School of Electrical Engineering and Informatics, Bandung Institute of Technology, 10 Ganesha Road, Lb. Siliwangi, Coblong, Bandung 40132, West Java, Indonesia (e-mail: ian@stei.itb.ac.id).

save the data in the content store. In sharing-techniques, the content store is share for every kind of content. The more popular content is prioritized for storage in CS. As long as CS is not full, content will be saved. When CS is full, less popular content will be removed and replaced with more popular content. In partitioning-techniques, CS is divided into 2 parts. Each section only stores certain class of content. In every section, the content will be kept based on the popularity. The caching performance is seen from the cache hit ratio on each type of content service.

II. NAMED DATA NETWORK TAXONOMY

Named Data Network Taxonomy can be divided into 3 components, namely system architecture, service system and application [14] as in Fig.3.

A. *NDN System Architecture*

The NDN architecture system includes a content naming scheme that must be unique within the NDN system. This naming structure must be unique to each requested package. The naming structure follows formats such as URIs using the "/" sign to separate each part of the name.

In NDN, there are two types of packets circulating in the network, namely Interest Packet and Data Packet [3], [4]. Interest Packet contains content requested by the consumer and Data packet is a packet containing data from the producer or other nodes in response to the previously sent Interest packet from the consumer. The type of packet in NDN as shown in Fig. 1.

INTEREST PACKET	DATA PACKET
Content Name	Content Name
Selector	Signature
Nonce	Signed Info
	Data

Figure 1. The type of packet in NDN[14]

Every router in NDN has 3 components as shown in Fig. 2, namely Content store (CS), Pending Interest Table (PIT) and Forwarding Information Based (FIB) [3][14]. When the consumer wants an information from the producer, the consumer will send a request for certain content using the Interest Packet. This Interest Packet will be delivered from the consumer, via the NDN router until it arrives at the producer node. During this forwarding process, each node will record in their pending interest table (PIT) information from which face the request came. The information stored on the PIT is useful for sending the requested data packets back from the producer or another router to the consumer. If the NDN router node has stored the requested content by the consumer in their content store, the packet interest sent by the consumer will not be forwarded to the producer, but directly responded by the router node by sending the requested content. If there is no content in the NDN router node, the router will check in the Forwarding Information Base (FIB) table where the request should be forwarded.

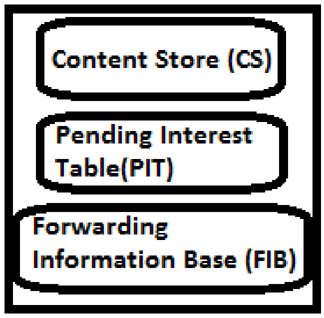

Figure 2. The NDN routers components [3], [4]

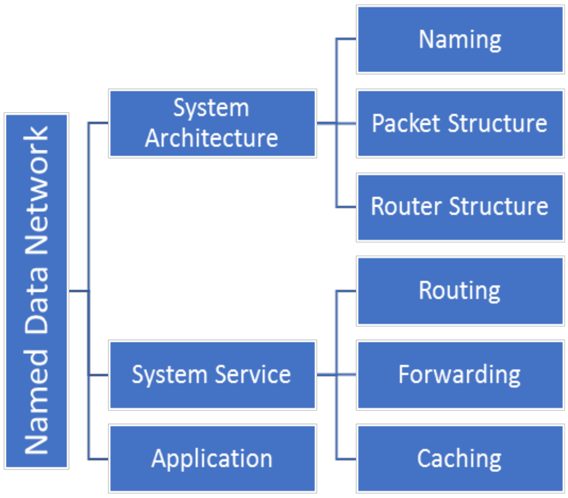

Figure 3. NDN Taxonomy [14]

B. *NDN System Service*

Related to the service system, NDN takes care of 3 things, namely routing, forwarding and caching. In the NDN routing, the network is required to determine the best path at the beginning of the formation of FIB table, then the forwarding task is to set the mechanism of packet delivery from node to node according to data in FIB table. The forwarding decision can use several parameters, such as link conditions, round trip time and others. FIB can set interface priorities for use in sending packets[15]. In addition to supporting the temporary data storage in the router, there is a caching mechanism on NDN to store content so that it can be as soon and as close as possible accessed by the consumer if needed [3][14]. The content store on the NDN node store the data temporarily. This causes consumer request does not have to always respond by the original server that could be located far from the consumer. The closest node of the consumer that stores the requested data can respond to the consumer requesting. There are 3 strategies related to setting up the Content Store, as shown in Fig.4, i.e. cache placement, cache content selection and cache policy design. Cache placement determines where a content will be placed in the network. The cache content selection determines which content will be stored and what should be removed from the content store. Cache policy design, determines the rules that run on the content store, for example, the various storage portions on the content store in a node or on the several nodes in the network. In this study, an analysis is related to cache policy design strategy to keep the data the content store of a node router, either partitioning or by sharing techniques.

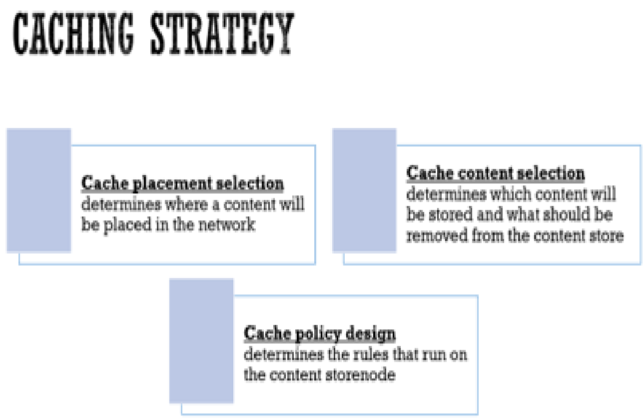

Figure 4. Caching strategy [16]

III. Research Model

A. Sharing and Partitioning Technique

In this research, the simulation is conducted to analyze the performance of caching with the sharing-technique and the partitioning technique. In this simulation, the data is generated with 2 class, that is A class and B class. Every content in A and B class is has different popularity too.

As in Fig.5, in sharing-technique, the content store is shared to kept the various of data from all of the consumer. As long as the content store is not full, the data will be saved. If the content store is full, the data with the lowest popularity will be deleted. In partitioning-technique, the content store is divided into 2 part. Every part is only kept the specific class of data. One part for store the A class of data and the the other part is only store the B class of data. Each class has the content with various popularity. The distribution model of popularity is generated following the Zipf-like distribution [17]. Zipf-like distribution describe the number of content with various popularity. There is a number of data that is more popular than others.

B. Simulation

In this simulation, the performance of content store is observed from cache hit ratio parameter related to the effect of different treatment of data class and also based on content popularity in the content store. In more detail, there is 3 scenario in this test aims to:

1. First scenario : Analyze the effect of cache size changes on NDN using content store partitioning and sharing in the content store
    a. Randomly generated interest
    b. Interest generation with Zipf-like distribution
2. Second scenario : Examine the effect of changes in Zipf-like exponent factor values for generating interest packets on NDN, using the partition and sharing techniques on the content store.
3. Third scenario: Examine the effect of the data storage proportion to cache on NDN

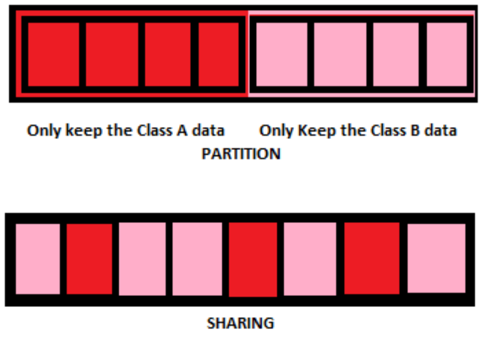

Figure 5. Partition and sharing on the content store

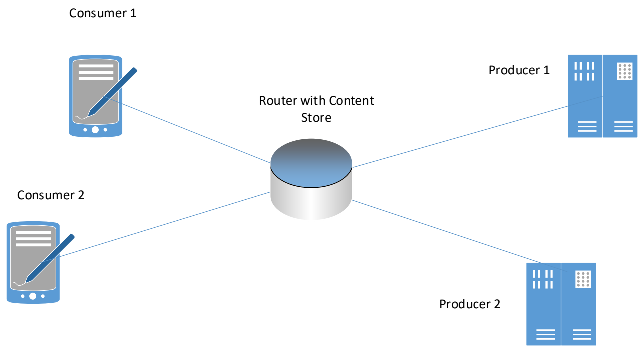

Figure 6. Simulation topology

The network model used for simulation in accordance with Fig. 6, where there are two consumer nodes that generate different interest to request a content. Content store on the network is used to support this communication by storing data. The cache rules on the content store are partitioning-technique and sharing-technique. A partition-technique separate to store content based on the class of data in the content store. Each class can only fill the part that belongs to it only. The second technique is sharing-techniques where all types can be stored in the content store without any limitations as long as there is still an empty space in the content store. If CS is full, less popular content will be removed and replaced with more popular content. The simulation parameter as in Table 1. Result and Analysis

The simulation results show that the interest generation pattern influences the cache hit ratio. In a 1st scenario, the simulation is conducted with random distribution and with Zipf-like distribution for interest packet generation. Random distribution means that consumer generate the interest of content randomly. Zipf-like distribution means that consumer generate the interest to request packets following the Zipf-like function. There are some content that is requested more frequently than the other. The popularity of content can be grouped into certain ranks, as shown in Fig. 11. The size of content store is modified from 10, 20, 50 and 100. For partitioning-technique, arranged the same proportion for the A class and B class of content. For sharing-technique there is no proportion arrangement. In a randomly generated interest pattern, the sharing-technique provides better results than the partitioning-technique.

ISSN: 1942-9703 / CC BY-NC-ND

The larger the content store size, the greater the cache hit ratio for each consumer. In the interest generation pattern using the Zipf-like distribution, the cache hit ratio for partitioning-techniques in the content store gives better results, maximum 5% more tha sharing-technique. This is because the selection of cached content begins to pay attention to popularity. A Storage of content in the content store in partitioning-techniques does not interfere with another content types. The bigger size of content store, the bigger cache hit ratio because content store can keep more data. This simulation results as in Fig.7 and Fig. 8

Table 1. Simulation Parameter

Parameter	Value
Cache Size	10, 20, 50, 100
Interest packet distribution	random dan Zipf-like $\alpha1=\alpha2=0.7$ (for 1st and 3rd scenarios) Zipf-like $\alpha1=0.7$; $\alpha2=0.7$, 0.8, 1, 1.2 (for 2nd scenario)
Proportion of content store	50:50 (for 1st and 2nd scenarios), 50:50, 20:80, 70:30, 80:20 (for 3rd scenario)
Number of content	100
Consumer	2
Producer	2
Replacement	LFU

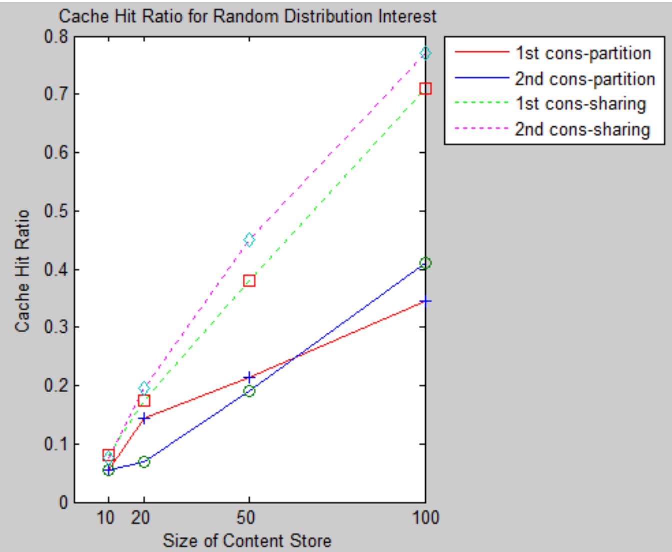

Figure 7. Cache hit ratio for Random Distribution

The 2nd Scenario aims to analyze the effect of exponent factor changes on Zipf-like, using partitioning-techniques and sharing-techniques. The greater the value of the exponential factor in the Zipf-like distribution, it means that the probability of consumer request for more popular content will be greater, as shown in Fig. 11. It causes the cache hit value is also greater. With the same proportion of content stores for 2 different content class using partitioning-techniques gives the better result than with sharing-techniques. In testing, the exponent constant value of user 1 is 0.7 and user 2 changes from 0.7, 0.8, 1 and 1.2. In partitioning-techniques, differences in content popularity lead to different cache hit rates, but on sharing-techniques, the cache hit rate difference is not significant, as compared to partitioning-techniques. The partition gives a larger cache hit rate, seen from the network level. This value is a total cache hit rate for each content class. The simulation results as in Fig. 9 and Fig. 10. It shows that the partitioning technique gives maximum 36% more cache hit ratio compare to sharing technique.

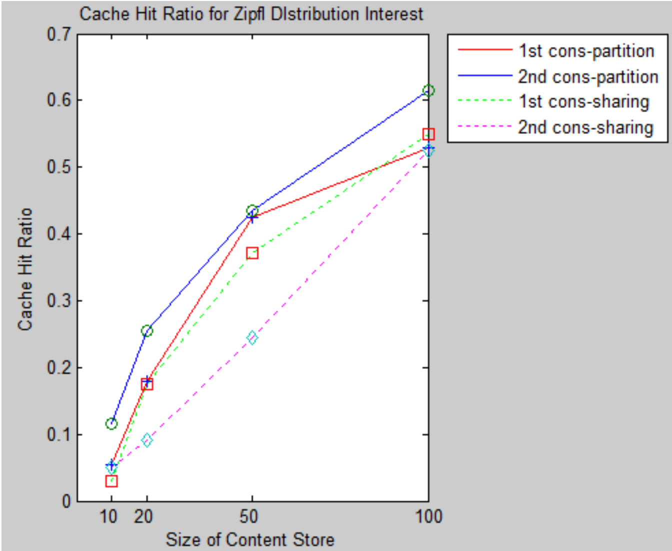

Figure 8. Cache hit ratio for Zipf-like Distribution

Scenario 3 aims to analyze the effect of changes in the proportion given for different content types. X-axis in Fig. 12 is a proportion of content store for 1st content class and 2nd content class. For example, 20:80 means that the proportion for 1st class is 20 unit packets to cache and proportion of 2nd class is 80 unit packets to cache. The whole content store size is 100 unit packets to cache. The simulation results as in Fig. 12 shows that in partitioning-techniques, content that gets a large proportion will have a larger cache hit ratio as well. In sharing-techniques, the difference in cache hit rate is not significant. Partitioning-technique allows flexibility of class-based settings for content based on the proportion of content store.

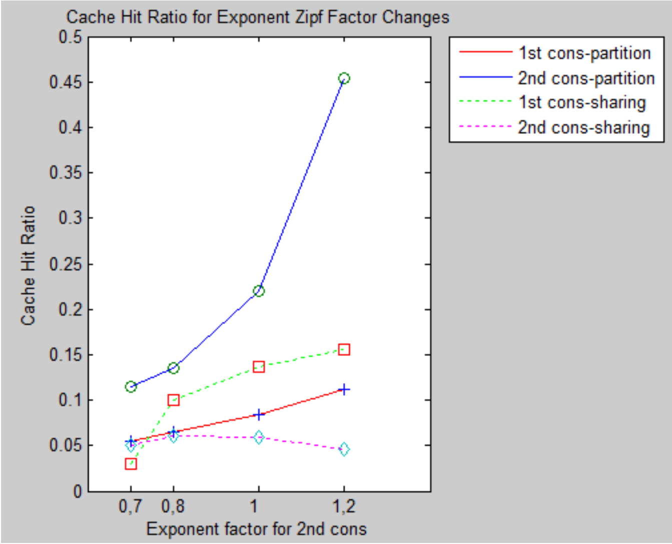

Figure 9. Cache Hit Ratio for Exponent Zipfl Factor Changes

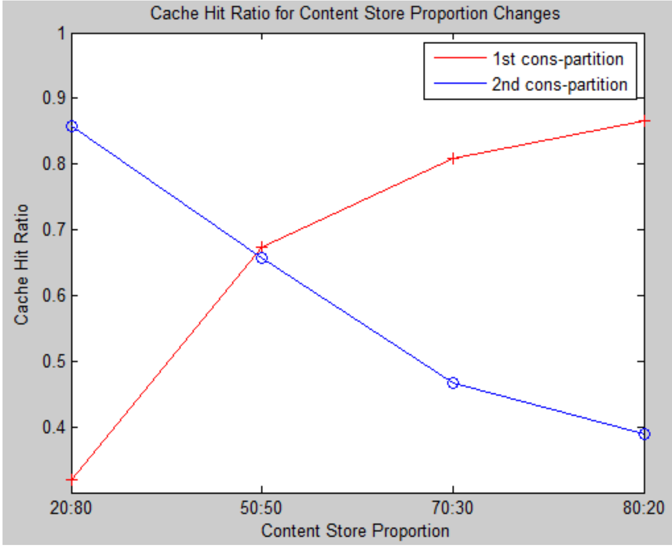

Figure 12. Cache hit ratio for CS Proportion Changes

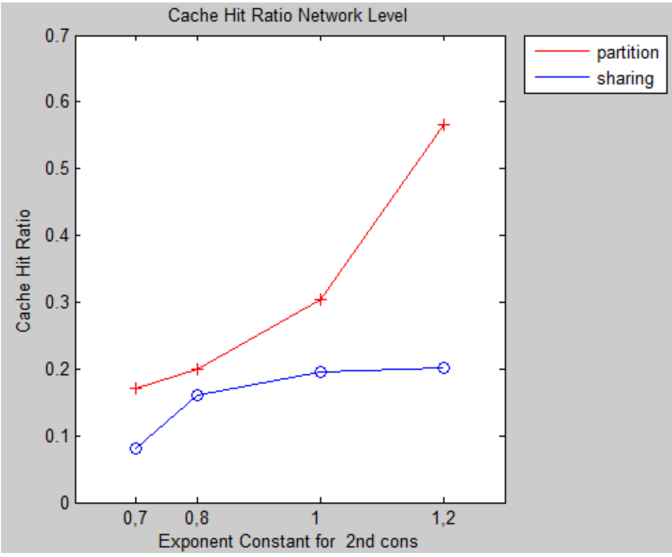

Figure 10. Cache hit ratio for network level

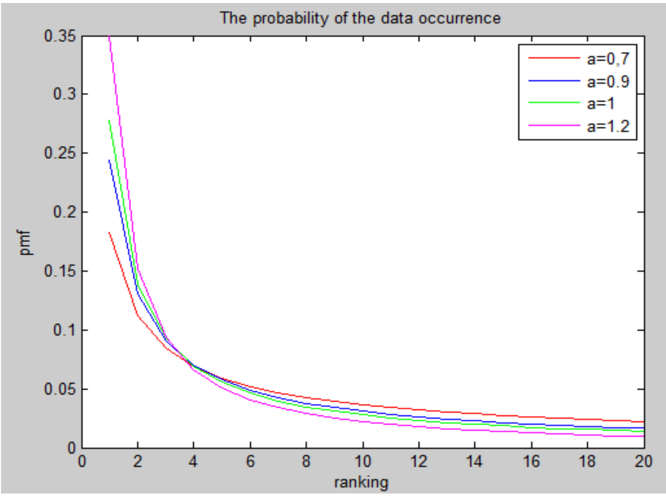

Figure 11. The probability of request for the data according to the data rank

IV. Conclusion and Future Research

From the simulation, can be concluded that the partitioning- technique gives the greater cache hit rate than the sharing- technique for interest generation conditions based on the popularity of the content. The value of exponent constants in the generation of interest using Zipf-like affects consumer cache hit ratio, The greater the constant value, the greater the cache hit ratio. The Sharing-technique is not enough to distinguish the cache hit ratio despite the popularity of different class of content so that it is required differentiation techniques content treatment with the partitioning-technique on the cache to improve the performance of NDN system. Partitioning-technique allows flexibility of class-based settings for content based on the proportion of content store.

For the future research, it is important to accommodate the different treatment for different classes of content and more than one classification of requirement in NDN according to current needs, to improve NDN performance

Acknowledgment

This research was supported in part by Telkom University and Ministry of Research Technology and Higher Education Republic of Indonesia.

References

[1] L. Kleinrock, "IEEE communications magazine," *An Early History of the Internet*, pp. 26–36, 2010.

[2] M. Mangili, F. Martignon, and A. Capone, "Performance analysis of Content-Centric and Content-Delivery networks with evolving object popularity," *Comput. Networks*, vol. 94, pp. 80–89, 2015.

[3] V. Jacobson, D. K. Smetters, N. H. Briggs, J. D. Thornton, M. F. Plass, and R. L. Braynard, "Networking Named Content," in *The 5th ACM International Conference on emerging Networking EXperiments and Technologies*, 2009, pp. 1–12.

[4] Anjali, "Components of Named Data Networking," *Int. J. Innov. Eng. Technol.*, vol. 7, no. 3, pp. 543–552, 2016.

[5] L. V. Yovita and N. R. Syambas, "Content Storage Effect on the Named Data Network Traffic Load," in *The International Conference on Telecommunication Systems, Services, and Applications 2017*, 2017.

[6] H. Situmorang, N. R. Syambas, and T. Juhana, "The Effect of Scaling the Size of Topology and Content Stored on the Named Data Networking," in *International Conference on*

Telecommunication Systems, Services, and Applications 2016, 2016, pp. 16–21.

[7] X. Hu, J. Gong, G. Cheng, and C. Pant, “Enhancing In-network Caching by Coupling Cache Placement , Replacement and Location,” in *IEEE ICC 2015 - Next Generation Networking Symposium*, 2015, pp. 5672–5678.

[8] M. Dehghan, L. Massoulie, D. Towsley, D. Menasche, and Y. C. Tay, “A Utility Optimization Approach to Network Cache Design,” in *IEEE INFOCOM 2016*, 2016.

[9] W. Zhang, R. Fan, F. Liu, and P. Lai, “Energy-Aware Caching,” in *IEEE 21st International Conference on Parallel and Distributed Systems*, 2015, pp. 473–480.

[10] H. Chao, W. J. Jian, H. Cho, C. Tsai, and J. Pan, “Prediction-Based Cache Adaptation for Named Data Networking,” *J. Comput.*, vol. 27, no. 1, pp. 45–55, 2015.

[11] J. Ren *et al.*, “MAGIC : a Distributed MAx-Gain In-network Caching Strategy in Information-Centric Networks,” in *2014 IEEE INFOCOM Workshop on Name-Oriented Mobility*, 2014, pp. 470–475.

[12] Sandvine Intelligent Broadband Network, “Identifying and Measuring Internet Traffic : Techniques and Considerations,” 2015.

[13] K. Cho, M. Lee, K. Park, T. T. Kwon, and Y. Choi, “WAVE : Popularity-based and Collaborative In-network Caching for Content-Oriented Networks,” in *2012 IEEE Conference on Computer Communications Workshops*, 2012, pp. 316–321.

[14] D. Saxena and I. I. T. Roorkee, “Named Data Networking: A Survey,” *Comput. Sci. Rev.*, vol. 19, pp. 15–55, 2016.

[15] T. Ariefianto W. and N. Rachmana Syambas, “Routing in NDN Network : a Survey and Future Perspectives,” in *The International Conference on Telecommunication Systems, Services, and Applications 2017*, 2017.

[16] H. Jin, D. Xu, C. Zhao, and D. Liang, “Information-centric mobile caching network frameworks and caching optimization : a survey,” *EURASIP J. Wirel. Commun. Netw.*, vol. 33, pp. 1–32, 2017.

[17] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, “Web Caching and Zipf-like Distributions : Evidence and Implications,” in *Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, 1999, vol. XX.

Leanna Vidya Yovita, a Ph.D. student at the School of Electrical Engineering and Informatics, Bandung Institute of Technology. She got her Bachelor and Master degree from Telkom University. Currently working as a lecturer at Telkom University, S1 Telecommunication Engineering program. She and the team have produced a textbook, titled “Jaringan Komputer” for the students of Telkom University. Research interests include Computer Networks, Content Centric Network, and Network Engineering.

Nana Rachmana Syambas. Professor Nana Rachmana Syambas was graduated with his bachelor degree in Electrical Engineering Department, ITB in 1983. He got his Master by Research degree from Royal Melbourne Institute of Technology, Australia in 1990 and a doctoral degree from School of Electrical Engineering and Informatics, ITB in 2011. He has been a lecturer at School of Electrical Engineering and Informatics, ITB since 1984. His research interest includes Telecommunication Networks, Telematic Services, Content Centric Network (CCN), Software Defined Network (SDN), Protocol engineering and Tele-traffic engineering. He has authored or coauthored over 80 published articles.

Ian Yosef Matheus Edward, graduated with his Bachelor degree in Electrical engineering, Bandung Institute of Technology in 1992. He got his Master in Bandung Institute of Technology, majoring Electrical engineering in 1996, and his Doctoral degree from University of Indonesia in 2007. He has been a lecturer at School of Electrical Engineering and Informatics, ITB since 1994. He is active in some research in telecommunication area. His research interest includes telecommunication technology, transmission technology, information system and multimedia technology.