ICNRG
Internet-Draft
Intended status: Informational
Expires: September 6, 2015

M. Arumaithurai
J. Chen
X. Fu
University of Goettingen
K. Ramakrishnan
University of California, Riverside
J. Seedorf
NEC

March 5, 2015

Enabling Publish/Subscribe in ICN
draft-jiachen-icn-pubsub-01

Abstract

Information-Centric Networks (ICN) provide substantial flexibility for users to obtain information without regard to the source of the information or its current location. Publish/subscribe (pub/sub) systems have gained popularity in society to provide the convenience of removing the temporal dependency of the user having to indicate an interest each time he or she wants to receive a particular piece of related information. Such an "information-centric" communication model should be supported in the new ICN network paradigm. This document outlines some research directions for ICN with respect to enhancing the inherently pull-based ICN approaches for achieving efficient pub/sub capability.

Table of Contents

1. Introduction

This document points out the need to support publish/subscribe (pub/sub) capabilities in ICN and the problems with the existing solutions. Further, the document discusses potential directions for enhancing Information Centric Networking (ICN) to achieve efficient pub/sub.

Section 2 describes the pub/sub systems and the challenges of such systems to the current Internet. Section 3 demonstrates the use of pub/sub systems in different scenarios. Section 4 outlines the requirements of an efficient pub/sub architecture and Section 5 discusses the related works and some possible shortcomings. In Section 6 we brief our standardisation considerations.

2. Pub/Sub Communication

Users increasingly desire access to information, ranging from news, financial markets, healthcare, to disaster relief and beyond, independent of who published it, where it is located, and often, when it was published. Typical representation of these usages are microblogs, RSS feed, social network, search engines, etc. A consumer may not wish (or it may even be infeasible) to receive all of the "channels" belonging to a myriad of information providers that disseminate items of interest, either on demand (such as web, twitter, blogs and social networks), or tune to a broadcast channel (e.g., television, radio, newspaper). In these cases, the consumer would rather prefer obtaining the data based on Content Descriptors (CD) such as a keyword, a tag, or a property of the content (publisher identity, published date etc.).

Publish/subscribe (pub/sub) systems are particularly suited for such kind of large scale content-oriented information dissemination, and provide the exibility for users to subscribe to information of interest, without being intimately tied to when that information is made available by publishers. With the use of an appropriate interface, users can select and filter the information desired so that they receive only what they are interested in, often irrespective of the publisher.

Intelligent end-systems and information aggregators (e.g., Google News and Yahoo! News, cable and satellite providers) have increasingly adapted their interfaces to provide a content-oriented pub/sub-based delivery method. However, these mechanisms are built on top of a centralized server based framework and can also result in a waste of network resources as shown in [Ramasubramanian2006][Katsaros2011], since the Internet protocol suite is focused on end-to-end delivery of data. Furthermore, issues of "coverage" and "timeliness" still exist in such forms of dissemination, where the aggregator may be selective in what information is made available.

Information-Centric Networks (ICN) is a new network paradigm that intendeds to achieve large scale data delivery with greater ease for users, greater scalability in terms of the amount of information disseminated as well as number of producers and consumers of

information, and greater efficiency in terms of network and server resource utilization.

It is also desirable for such a network to assist the pub/sub communication model that delivers the information from any of the producers to all subscribers. Moreover, it is desirable for the network to assist in delivering fine-grained information to the subscriber.

Recently, works such as [Schmidt2012],[Carzaniga2011],[Chen2011],[Chen2012] have also highlighted the need for ICN to support a pub/sub like communication model.

3. Scenarios of Pub/Sub Architecture

In this section, we list several use cases of pub/sub architectures in ICN. They help us to understand the requirements of an efficient pub/sub architecture and why the existing solutions fall short.

3.1. Online Social Networks and RSS Feeds

Online social networks (e.g., Twitter, Facebook, etc.) and Rich Site Summary (RSS) feeds are typical use cases for a content-centric pub/sub system. In such systems, the receivers receive messages either from friends, followees, or from some information aggregators. They do not care which exact machine is sending the message (content-centric), nor do they know when and what is the name of the next message they are going to receive (temporal separation).

To prevent the receivers from polling all the possible providers, existing systems use web servers as rendezvous points: the publishers send new messages to the servers and the receivers/subscribers poll the server periodically. This still causes great wastage for the (HTTP) servers answering "304 - Not Modified" repeatedly since the message update frequency is usually lower than the polling frequency.

3.2. Online Gaming and Audio/Video Conferencing

Massively multiplayer online role-playing games (MMORPGs, e.g., Counter-Strike, Quake, World of Warcraft, etc.) and audio/video conferencing (e.g., Skype meeting, Web Whiteboard, Etherpad, etc.) is another kind of content-centric pub/sub systems. Similar to the social network scenario, users in such systems only care about the content, either the area of interest (AoI) or the conference partners, and they do not know when and from where the next message will come. But different from the previous scenario, such systems

require real-time update (message) delivery and these messages are usually smaller in size compared to the online social networks.

Many of these systems choose to use HTTPS or direct TCP connection between the server and the users to enable the capability of server "pushing" the updates to the user. But maintaining such links are costly. MMORPGs usually limit the number of players in a same game which greatly reduces the interesting of these games.

3.3. Notification Systems in Disaster

Disasters have often disrupted communications because of damages to critical infrastructure. For instance in the aftermath of the Japanese Earthquake in 2011, approximately 1,200,000 fixed telephone lines and 15,000 base-stations were not functioning. On average, 22% (with peaks up to 65% in some areas) of the base-stations had to shut down due to the lack of power or damages to the infrastructure.

Contradictory to the loss of available hardware capacity, during and in the aftermath of a disaster, there is a substantial increase in the amount of traffic generated because of the natural anxiety and panic among people and the need to organize rescue and emergency services. Many of these traffic are in the form of a pub/sub communication model, e.g., the government needs to publish some notifications (recovery status, new shelter locations, etc.), the refugees need to notify their friends about their safety, or people needs to ask for help from ambulances or fire brigade. In the Japanese case, the congestion caused by such traffic resulted in restrictions in voice traffic up to 95%, including emergency priority calls.

4. Requirements of an Efficient Pub/Sub Architecture

Given a pub/sub communication model as described in Section 2, on a high-level one can derive the following (incomplete) list of basic requirements:

- Decouple publishers and subscribers: In an ideal pub/sub environment, publishers only focus on their core task of publishing while not having to maintain membership status, and subscribers receive content from a multitude of sources without having to worry about maintaining a list of publishers and frequently polling them for the availability of fresh data. Moreover, a consumer may not wish (and it may even be infeasible) to subscribe to all of the channels belonging to a myriad of information providers that disseminate items of interest, either on demand (such as web, twitter, blogs and social networks), or tune to a broadcast channel (e.g., television, radio, newspaper).

In these cases, support should be provided to the consumer who would prefer obtaining the data based on descriptors such as keywords, tags, or other properties of the published data.

- Push enabled dissemination: The ability to exploit push-based delivery is a key to achieving timeliness and to avoid wasting server and network resources because of redundant polls. Therefore, an efficient pub/sub architecture must provide the capability for publishers to push information to online subscribers interested in it. Such timely dissemination is necessary in many scenarios such as disaster (e.g., Tsunami) warnings, stock market information, news and gaming.

- Scalability: The target architecture should be able to accommodate a large number of subscribers as well as publishers (often subscribers are also publishers as user-generated content becomes common). Therefore, it should minimize the amount of states maintained in the network, ensure the load on the publisher grows slowly (sublinearly) with the number of subscribers. The load on the subscribers should also grow slowly with the number of publishers (e.g., dealing with the burden of duplicate elimination). Importantly, the load on the network should not grow significantly with the growth in the number of publishers and subscribers. There is also a need to accommodate a very large range in the amount of information that may be disseminated, and the need for all elements of the pub/sub framework in a content-centric environment to scale in a manageable way.

- Efficiency: The architecture should enable a nearly unlimited amount of information being generated by publishers, allow for delivery of information related to subscriptions independent of the frequency at which that information is generated by publishers. The architecture must utilize network and server resources efficiently. It is desirable that content is not transmitted multiple times by a server or on a link. Furthermore, the overhead on publisher and subscriber end-points to query unnecessarily for information must be minimized.

- Dynamicity: The architecture should be able to deal with the substantial churn in subscription state, allowing a large number of users to join, leave and frequently change their subscriptions. The topics of interest may change frequently as well (e.g., in a Twitter-like publishing environment, where the popular topics change frequently).

Additionally, to support a full-fledge pub-sub environment, it is desirable that the target system support the following additional features:

- Support hierarchies and context in naming content: It is desirable to be able to exploit both context and hierarchies in identifying content. Hierarchical naming has been recognized by NDN as well. Exploiting context enables a richer identification of content (in both subscriptions and published information), as noted in the database community.

- Support two-step dissemination for policy control and user interest: There is a need for pub/sub environments to support a two-step dissemination process both for reasons of policy and access control at the publisher as well as managing delivery of large volume content. In such a scenario, the pub/sub framework would be designed to publish only a snippet of the data (containing a description of the content and the method how to obtain it) to subscribers. The subscribers then request for the content based on their interest and allowance.

- Subscriber offline support: Another typical characteristic of pub-sub environments is that subscribers could be offline at the time the data is published. There is clearly a need for asynchronous delivery of information in a pub/sub environment in an efficient, seamless and scalable manner. The system needs to allow users who were online to retrieve the data that they have missed. It should also allow new subscribers to retrieve previously published content that they are interested in. We envisage a server that stores all the content published.

- Prevent Spam/DoS: Spam and DoS attacks are security issues that concern push based pub/sub mechanisms. Efforts to mitigate this at the network layer as well as at the application layer should be considered.

Additionally, it will be desirable to have the following features to support a (limited) pub-sub environment in a disaster affected scneario:

- TBD

- TBD

5. Related Work

5.1. IP/Overlay Multicast

IP multicast [RFC1112] is a candidate solution for efficiently delivering content to multiple receivers. A sender sends data to a multicast group address that subscribers could join. Multicast routing protocols such as PIM-SM [RFC4601] construct and maintain a

tree from each sender to all receivers of a multicast group. However, IP multicast isn't an efficient pub/sub delivery mechanism for several reasons: 1) IP multicast is designed for delivery of packets to connected end-points. Dealing with disconnected operation (when subscribers are online) would have to be an application layer issue. Overlay multicast solutions such as [Jannotti2000][Chu2002][Banerjee2002] are agnostic of the underlying network topology, usually relying on multiple unicasts in the underlay path and are therefore also inefficient as a pub/sub delivery mechanism. 2) The somewhat limited multicast group address space makes it difficult to support a direct mapping of CDs to IP multicast addresses. 3) Current IP multicast is not able to exploit relationships between information elements, such as CDs. CDs may be hierarchical or may have a contextual relationship, which enables multiple CDs to be mapped to a group. For example, consider a publisher that sends a message to all the subscribers interested in football, and subscribers who are interested in receiving messages about all sports. The message from the publisher will have to be sent to two distinct IP multicast groups. If there happens to be a subscriber of messages on sports and football, (s)he will receive the same message twice and will have to perform redundancy elimination in the application layer. The result is a waste in network traffic and processing at both ends.

5.2. Named-Data Networking (NDN)

NDN has limited intrinsic support for pub/sub systems, a critical need in a content centric environment. The aggregation of pending Interests at routers achieves efficient dissemination of information from NDN nodes. But this aggregation is similar to a cache hit in a content distribution network (CDN) cache, which occurs only if subscribers send their Interests with some temporal locality. Thus it avoids multiple Interest queries having to be processed directly by the content provider. Note however that this is still a pull-based information delivery method and depends both on temporal locality of interests and a large enough cache to achieve effective caching in the (content centric) network. On the other hand, native multicast support allows for a much more scalable push-based pub/sub environment, since it is not sensitive to issues such as the cycling of the cache when a large amount of information is disseminated.

TBD: Update it based on recent modifications by the NDN team

5.3. CCN

TBD: Update it based on recent modifications by the CCN team

5.4. Content-Oriented Publish/Subscribe(COPSS)

COPSS enhances CCN/NDN with a push-based delivery mechanism using multicast in a content-centric framework. It is designed to satisfy the requirements mentioned above, especially to provide temporal separation between subscription (or expression of Interest) and publication. At the content-centric network layer, COPSS uses a multiple-sender, multiple-receiver multicast capability, in much the same manner as PIM-SM.

5.5. PSIRP Project

TBD

5.6. NetInf Project (http://www.netinf.org)

TBD

5.7. Pursuit Project (http://www.fp7-pursuit.eu/)

TBD

5.8. Other Related Works

Here we list the other related works we are considering. The list might not be complete and we intend to add to it based on feedback received in further revisions.

- A. Carzaniga, M. Rutherford, A. Wolf, A routing scheme for content-based networking, in: INFOCOM, 2004.
- B. Segall, D. Arnold, J. Boot, M. Henderson, T. Phelps, Content Based Routing with Elvin, in: AUUG2K, 2000.
- C. Esteve, F. Verdi, M. Magalhaes, Towards a new generation of information-oriented Internetworking architectures, in: ReArch, 2008.
- G. Chockler, R. Melamed, Y. Tock, R. Vitenberg, SpiderCast: a scalable interest-aware overlay for topic-based pub/sub communication, in: DEBS, 2007.
- H. Eriksson, Mbone: the multicast backbone, Commun. ACM 37 (8) (1994) 54-60.
- M. Ott, L. French, R. Mago, D. Makwana, Xml-based semantic multicast routing: an overlay network architecture for future information services, in: GLOBECOM, 2004.

- P. T. Eugster, P. A. Felber, R. Guerraoui, A.-M. Kermarrec, The many faces of publish/subscribe, ACM Comput. Surv. 35 (2) (2003) 114-131.

- R. Baldoni, R. Beraldi, V. Quema, L. Querzoni, S. Tucci-Piergiovanni, TERA: topic-based event routing for peer-to-peer architectures, in: DEBS, 2007.

- R. V. Renesse, K. P. Birman, W. Vogels, Astrolabe: A Robust and Scalable Technology for Distributed System Monitoring, Management, and Data Mining, ACM TOCS 21 (2001) 66-85.

- S. Voulgaris, E. Riviere, A.-M. Kermarrec, M. Van Steen, Sub-2-Sub: Self-Organizing Content-Based Publish and Subscribe for Dynamic and Large Scale Collaborative Networks, Research report, INRIA (December 2005).

- T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, I. Stoica, A data-oriented (and beyond) network architecture, in: SIGCOMM, 2007.

- V. Ramasubramanian, R. Peterson, E. G. Sirer, Corona: a high performance publish-subscribe system for the world wide web, in: NSDI, 2006.

- V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, R. L. Braynard, Networking Named Content, in: CoNEXT, 2009.

- Y. Cui, B. Li, K. Nahrstedt, ostream: asynchronous streaming multicast in application-layer overlay networks, JSAC 22 (1) (2004) 91-106.

- Y. Diao, S. Rizvi, M. J. Franklin, Towards an internet-scale XML dissemination service, in: VLDB, 2004.

6. Standardisation Considerations

Future versions of this document will outline a concrete protocol specification for pub/sub support for ICN. Below some initial standardisation considerations are outlined.

An initial list of details that need to be specified is the following:

- Pub/Sub related interfaces/APIs

- Pub/Sub related data structure modification to existing ICN proposals

We are also considering to write a survey paper that accumulates all the Pub/sub related work.

7. References

7.1. Normative References

[RFC1112] Deering, S., "Host extensions for IP multicasting", STD 5, RFC 1112, August 1989.

[RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.

7.2. Informative References

[Banerjee2002] Banerjee, S., Bhattacharjee, B., and C. Kommareddy, "Scalable application layer multicast", SIGCOMM, 2002, .

[Carzaniga2011] Carzaniga, A., Papalini, M., and A. Wolf, "Content-based Publish/Subscribe Networking and Information-centric Networking", Proceedings of the ACM SIGCOMM workshop on Information-centric networking, ACM, 2011, .

[Chen2011] Chen, J., Arumaithurai, M., Fu, X., and K. Ramakrishnan, "COPSS: An Efficient Content Oriented Publish/Subscribe System", ACM/IEEE 7th Symposium on Architectures for Networking and Communications Systems (ANCS), 2011, .

[Chen2012] Chen, J., Arumaithurai, M., Fu, X., and K. Ramakrishnan, "G-COPSS: A Content Centric Communication Infrastructure for Gaming Applications", IEEE 32nd International Conference on Distributed Computing Systems (ICDCS), 2012, .

[Chu2002] Chu, Y., Rao, S., Seshan, S., and H. Zhang, "A case for end system multicast", IEEE Journal on Selected Areas in Communications 20, no. 8 (2002): 1456-1471, .

[Fenner2005]
Fenner, W., Rabinovich, M., Ramakrishnan, K., Srivastava, D., and Y. Zhang, "XTreeNet: Scalable overlay networks for XML content dissemination and querying (synopsis)", 10th International Workshop on Web Content Caching and Distribution (WCW), 2005, .

[Jannotti2000]
Jannotti, J., Gifford, D., Johnson, K., and M. Kaashoek, "Overcast: reliable multicasting with on overlay network", Proceedings of the 4th conference on Symposium on Operating System Design & Implementation-Volume 4, pp. 14-14. USENIX Association, 2000, .

[Katsaros2011]
Katsaros, K., Xylomenos, G., and G. Polyzos, "MultiCache: An overlay architecture for information-centric networking", Computer Networks 55.4 (2011): 936-947, .

[Ramasubramanian2006]
Ramasubramanian, V., Peterson, R., and E. Sirer, "Corona: A High Performance Publish-Subscribe System for the World Wide Web", NSDI. Vol. 6. 2006, .

[Schmidt2012]
Schmidt, T. and M. Waehlisch, "Why We Shouldn t Forget Multicast in Name-oriented Publish/Subscribe", arXiv preprint arXiv:1201.0349 (2012), .

Appendix A. Acknowledgment

This document has been supported by the GreenICN project (GreenICN: Architecture and Applications of Green Information Centric Networking), a research project supported jointly by the European Commission under its 7th Framework Program (contract no. 608518) and the National Institute of Information and Communications Technology (NICT) in Japan (contract no. 167). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the GreenICN project, the European Commission, or NICT.

Authors' Addresses

Mayutan Arumaithurai
University of Goettingen
Goldschmidt Str. 7
Goettingen 37077
Germany

Phone: +49 551 39 172046
Fax: +49 551 39 14416
Email: arumaithurai@informatik.uni-goettingen.de

Jiachen Chen
University of Goettingen
Goldschmidt Str. 7
Goettingen 37077
Germany

Phone: +49 551 39 172051
Fax: +49 551 39 14416
Email: jiachen@informatik.uni-goettingen.de

Xiaoming Fu
University of Goettingen
Goldschmidt Str. 7
Goettingen 37077
Germany

Phone: +49 551 39 172023
Fax: +49 551 39 14416
Email: fu@informatik.uni-goettingen.de

K. K. Ramakrishnan
University of California, Riverside
900 University Ave
Riverside CA 92521
USA

Email: kkramakrishnan@yahoo.com

Jan Seedorf
NEC
Kurfuerstenanlage 36
Heidelberg 69115
Germany

Phone: +49 6221 4342 221
Fax: +49 6221 4342 155
Email: seedorf@neclab.eu