PSoCCN: Publish/Subscribe Support in Content-Centric Networking

Nischay Bodapati
University of Toronto

Kaiwen Zhang
University of Toronto

Hans-Arno Jacobsen
University of Toronto

ABSTRACT

Content-Centric Networking (CCN) is a proposed model for Future Internet Architecture (FIA) which replaces traditional host-oriented interactions with named data networking. In this demo, we present PSoCCN, our model for supporting a software-level publish/subscribe middleware over CCN. Our model demonstrates how CCN features can be leveraged to support publish/subscribe routing. We first demonstrate a simple model where CCN is used a straightforward replacement for one-to-one messaging between brokers using sockets. We then extend our model to support multicasting using CCN.

PSoCCN is implemented using the PADRES publish/subscribe prototype over CCNx. As a guiding application, we present the Connected Vehicle and Smart Traffic (CVST) alert tool. This CVST application allows users to subscribe to location-based alerts on the road conditions in the Greater Toronto Area (GTA).

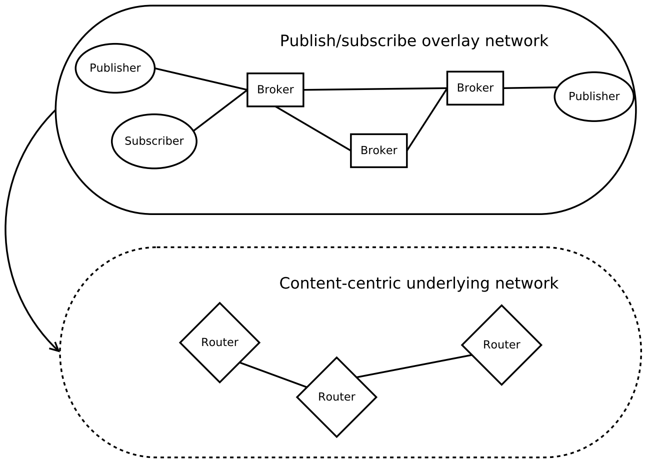

Figure 1: PSoCCN Architecture

Categories and Subject Descriptors

C.2.4 [**Computer-Communication Networks**]: Distributed Systems—*Distributed applications*; H.3.4 [**Information Systems**]: Systems and Software—*Distributed systems*

General Terms

Design, Algorithms, Performance

Keywords

Publish/Subscribe, Content-centric networking, Traffic monitoring

1. INTRODUCTION

Content-centric networking is touted as a blank slate approach to redesigning the Internet, eschewing outdated host-centric concepts of IP in favor for named data networking [5]. This new Internet architecture puts the focus on flexibility, allowing interest in a piece of content to be accessible from multiple sources, and security, which requires authentication per piece of content. This named data routing model is in line with the publish/subscribe paradigm, which supports many-to-many interaction patterns through its matching model. Therefore, efforts have been made to introduce publish/subscribe features in the CCN layer [2, 6]. Unlike the aforementioned, we are interested in developing the PSoCCN model, which allows current publish/subscribe middleware systems (which currently function over IP) to be adapted to operate over CCN. We first introduce a simplified model which allows only for point-to-point communication, and then extend it to leverage the multicast nature of CCN interest packets to support many-to-many communication.

Applications such as teleconferencing, have been shown to leverage the multi-point capabilities of CCN [4]. We show how the same features benefit PSoCCN through our example application, the Connected Vehicles and Smart Transportation (CVST) Alert application. In this application, users wish to be notified of traffic alerts based on location. We show how the sample application operates over our PSoCCN implementation using PADRES, a publish/subscribe system written in Java [3], and CCNx, an open-source implementation of CCN maintained by PARC [1].

2. MODEL

Middleware '14, December 08 - 12 2014, Bordeaux, France
Copyright 2014 ACM 978-1-4503-3220-0/14/12$15.00
http://dx.doi.org/10.1145/2678508.2678516

Figure 2: CVST Application

Figure 1 shows the architecture of our proposed PSoCCN model. Our publish/subscribe system operates on a broker overlay structure, where publication routing and matching is performed by the brokers. In a typical system, this broker overlay operates on a underlying network consisting of IP routers. PSoCCN replaces this underlying network using CCN routers. In other words, communication links between two nodes in the overlay (broker to broker, or pub/sub client to broker) are now carried through CCN. We therefore modify the network layer of our pub/sub system in order to transmit data via CCN packets. The advantage of such approach is that the other pub/sub components are agnostic to the choice of underlying communication protocol.

Our first protocol is a replacement of one-to-one communication (eg. using TCP sockets). Each link is assigned a name for both directions. For example, the link between a broker A and a broker B has two labels associated: */BrokerA/BrokerB* and */BrokerB/BrokerA*. The first one is used by Broker A to communicate to Broker B and vice versa. These labels are defined based solely on the topology and the nodes must maintain the corresponding interests as long as the links are still up.

Our second protocol leverages the multicasting capability of CCN to efficiently propagate pub/sub messages to multiple destinations without duplicating packets. Upon receipt of a pub/sub message, a broker determines its set of next hop destinations via regular routing/matching. It then initiates a three-way protocol with the targeted next hops to determine a label which will be used to disseminate the message. For example, a broker A wants to send a publication to both brokers B and C. It does so by telling B and C (via their respective channels) to send an interest packet for a new name (eg. */Message1*). Upon acknowledgment receipts, broker A can then send the new message via this label instead of sending it individually to brokers B and C.

3. CVST APPLICATION

The demo application is the Connected Vehicle and Smart Transportation Alerts, an extension of the Android smartphone application presented in [7]. Figure 2 shows a sample screen of the alert page of the application. We allow users to be notified of traffic alerts in the Greater Toronto Area (GTA). Notifications can be specified based on location (eg. the name of a street, or using geographical boundaries), as well as the type of alerts (eg. traffic conditions, transit, etc.). The degree of flexibility in the subscriptions suit content-based publish/subscribe matching, while the mobile nature of its users operates well over CCN.

We therefore employ PSoCCN as the communication layer between the various sources of traffic data and the end-users of our alerting system.

4. IMPLEMENTATION AND LIVE DEMO

We implemented PSoCCN using PADRES, a publish/subscribe system written in Java, over CCNx, an open-source implementation of CCN. Most of the implementation work is located within the network engine component of PADRES, making our design modular.

The demo will showcase the use of PSoCCN using Padres/CCNx in our CVST alert application. Metric visualization will be provided using the PADRES monitor tool to compare PSoCCN with the standard TCP implementation. We will operate PSoCCN using both protocols: one supporting one-to-one communication and the other with extended multicast capabilities.

5. REFERENCES

[1] CCNx | Content Centric Networking. *https://www.ccnx.org/*, 2014.

[2] J. Chen, M. Arumaithurai, L. Jiao, X. Fu, and K. Ramakrishnan. Copss: An efficient content oriented publish/subscribe system. In *Architectures for Networking and Communications Systems (ANCS), 2011 Seventh ACM/IEEE Symposium on*, ANCS '11, pages 99–110, 2011.

[3] E. Fidler, H.-A. Jacobsen, G. Li, and S. Mankovski. The PADRES distributed publish/subscribe system. In *ICFI'05*, pages 12–30, 2005.

[4] V. Jacobson, D. K. Smetters, N. H. Briggs, M. F. Plass, P. Stewart, J. D. Thornton, and R. L. Braynard. Voccn: Voice-over content-centric networks. In *Proceedings of the 2009 Workshop on Re-architecting the Internet*, ReArch '09, pages 1–6, 2009.

[5] 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*, CoNEXT '09, pages 1–12, 2009.

[6] A. Nigam. Information centric data collection and dissemination fabric for smart infrastructures. Master's thesis, University of Toronto, Canada, 2013.

[7] Y. Yue, K. Zhang, and H.-A. Jacobsen. Smart phone application for connected vehicles and smart transportation. In *Proceedings Demo & Poster Track of ACM/IFIP/USENIX International Middleware Conference*, MiddlewareDPT '13, pages 12:1–12:2, 2013.