Wide Area Network-scale Discovery and Data Dissemination in Data-centric Publish/Subscribe Systems

Kyoungho An* and
Aniruddha Gokhale
EECS Dept, Vanderbilt Univ
Nashville, TN, USA
{kyoungho.an,a.gokhale}@vanderbilt.edu

Sumant Tambe
Real-Time Innovations
Sunnyvale, CA, USA
sumant@rti.com

Takayuki Kuroda
Knowledge Discovery
Research Lab, NEC Corp
Kawasaki, Kanagawa, Japan
t-kuroda@ax.jp.nec.com

ABSTRACT

Distributed systems found in application domains, such as smart transportation and smart grids, inherently require dissemination of large amount of data over wide area networks (WAN). A large portion of this data is analyzed and used to manage the overall health and safety of these distributed systems. The data-centric, publish/subscribe (pub-/sub) paradigm is an attractive choice to address these needs because it provides scalable and loosely coupled data communications. However, existing data-centric pub/sub mechanisms supporting quality of service (QoS) tend to operate effectively only within local area networks. Likewise broker-based solutions that operate at WAN-scale seldom provide mechanisms to coordinate among themselves for discovery and dissemination of information, and cannot handle both the heterogeneity of pub/sub endpoints as well as the significant churn in endpoints that is common in WAN-scale systems. To address these limitations, this paper presents *PubSubCoord*, which is a cloud-based coordination and discovery service for WAN-scale pub/sub systems. *PubSubCoord*, which builds upon the ZooKeeper coordination primitives, realizes a WAN-scale, adaptive, and low-latency endpoint discovery and data dissemination architecture by (a) balancing the load using elastic cloud resources, (b) clustering brokers by topics for affinity, and (c) minimizing the number of data delivery hops in the pub/sub overlay.

Categories and Subject Descriptors

C.2.4 [**Computer Systems**]: Distributed Systems—*publish/subscribe*; D.2.11 [**Software Engineering**]: Architectures—*middleware*

Keywords

Publish/subscribe, Distributed Systems, WAN

*Author is currently with Real-Time Innovations

Middleware '15 Vancouver, BC, Canada
Copyright 2015 ACM X-XXXXX-XX-X/XX/XX ...$15.00.

1. PUBSUBCOORD CONTRIBUTIONS

PubSubCoord is a cloud-based coordination service for geographically distributed publish/subscribe (pub/sub) brokers to transparently connect heterogeneous endpoints and realize Internet-scale data-centric pub/sub systems. PubSubCoord addresses the following challenges in the context of scalable, reliable, and dynamic pub/sub systems:

- **Scalability and Availability:** To address the scalability and availability needs of data dissemination across WAN-scale systems despite NAT/firewall issues and failures, PubSubCoord uses the separation of concerns principle to decouple local area-based brokers called *edge brokers* that handle local pub/sub issues from cloud-based brokers called *routing brokers* that handle routing between edge brokers.
- **Dynamic Discovery and Dissemination:** To support dynamic discovery and data routing between brokers, PubSubCoord provides efficient coordination among the brokers by building pub/sub-specific event notifications using the basic primitives provided by ZooKeeper coordination service [2]. This solution helps to synchronize the dissemination paths over the dynamic overlay network of brokers and heterogeneous endpoints.
- **Overload and Fault Management:** To manage topic and dissemination overload, PubSubCoord uses cloud-based elasticity to balance the load. Load balancing and broker failures are handled by an elected leader.
- **Performance Optimizations:** For those dissemination paths that need both low latency and reliability assurances, PubSubCoord trades off resource usage in favor of deadline-aware overlays that build multiple, redundant paths between brokers.

Our PubSubCoord design can easily be adopted by industrial systems because its design is based on proven software engineering design patterns. We have favored maximal reuse of proven industrial-strength solutions wherever possible instead of reinventing the wheel. A key guiding principle for us was to ensure a *non-invasive and extensible design* which preserves the endpoint discovery and data dissemination model of the underlying pub/sub messaging system by tunneling discovery and dissemination messages across the hierarchy of brokers. Using this approach, it is possible to support multiple concrete pub/sub technologies without breaking their individual semantics. We have empirically validated our claims.

2. PUBSUBCOORD ARCHITECTURE

Figure 1 shows the layered PubSubCoord architecture depicting three layers: a coordination layer, a pub/sub broker overlay layer, and the physical network layer. The pub/sub broker overlay comprises a broker hierarchy based on a separation of concerns representing the logical network of brokers and endpoints in a system. An *edge broker* is directly connected to individual endpoints in a local area network (LAN) (*i.e.*, which represents an isolated network) to serve as a bridge to other endpoints placed in different networks. A *routing broker* serves as a mediator to route data between edge brokers according to assigned and matched topics that are present in the global data space. The coordination layer comprises an ensemble of ZooKeeper servers used for coordination between the brokers.

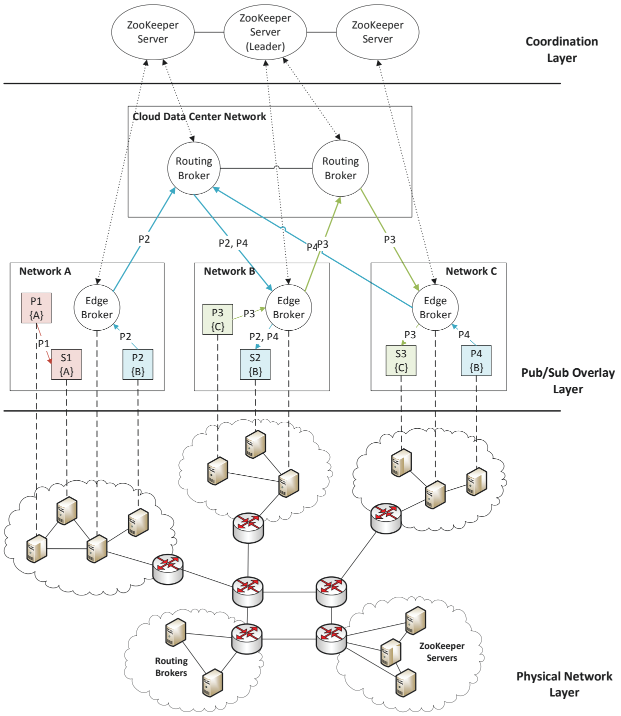

Figure 1: Layering and Separation of Concerns in the PubSubCoord Architecture

The data dissemination in PubSubCoord is explained using an example from Figure 1. $Pi\{T\}$ denotes a publisher i that publishes topic T (similarly for a subscriber $Sj\{T\}$). In the example, since publisher $P1$ and subscriber $S1$ are the only endpoints interested in topic A, they communicate within their local network A only using the techniques provided by the underlying pub/sub technology. On the other hand, $P2$, $P4$, and $S2$ are interested in topic B but are deployed in different isolated networks. So their communications are routed through a routing broker that is responsible for topic B. The network transport protocol between brokers is configurable, but TCP is used as a default transport to ensure reliable communication over WANs. As seen from this example, a maximum of 2 hops on the overlay network are incurred by data flowing from one isolated network to another (*e.g.*, network A to B).

3. CONCLUDING REMARKS

Emerging WAN-scale distributed systems found in domains, such as transportation and smart grid, must disseminate large volumes of data between a large number of heterogeneous entities that are geographically distributed, and require a variety of QoS properties for data dissemination from the publishers of information to the subscribers. Many disparate solutions that handle individual aspects of the problem space exist but seldom have these techniques been brought together holistically to solve the broader set of challenges. There is a need to systematically integrate these proven solutions while also providing new capabilities. This is challenging, particularly when the sum of the parts itself must be made reusable and applicable across a variety of pub/sub technologies.

To address some of these broader challenges, this paper presents the design, implementation, and evaluation of PubSubCoord, which is a cloud-based coordination and discovery service and middleware for geographically dispersed pub-/sub applications. PubSubCoord supports scalability in terms of data dissemination as well as coordination, dynamic discovery, and configurable QoS properties. Our experimental results validate our claims.

Several open research problems exist in the context of PubSubCoord. For example, we have not fully explored the fault tolerance and security dimensions in current work. Similarly, the edge broker bottleneck remains to be resolved. Furthermore, our work uses overlay networks; thus we do not have control over the QoS over the network links. It is possible to use software defined networking (SDN) to control the network QoS for pub/sub traffic and provide differential services to different pub/sub flows.

PubSubCoord is designed with reuse in mind and can easily be adopted in industrial settings. A detailed description of PubSubCoord including experimental validation appears in [1]. The PubSubCoord middleware and test harness can be downloaded from:

`www.dre.vanderbilt.edu/~kyoungho/pubsubcoord`.

Acknowledgments

This work is supported in part by NSF CAREER CNS 0845789. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NSF.

4. REFERENCES

[1] K. An, A. Gokhale, S. Tambe, and T. Kuroda. Data-centric Publish/Subscribe Discovery and Dissemination in WAN-scale Distributed Systems. Technical Report ISIS-15-120, Institute for Software Integrated Systems, Nashville, TN, USA, 2015.

[2] P. Hunt, M. Konar, F. P. Junqueira, and B. Reed. Zookeeper: wait-free coordination for internet-scale systems. In *Proceedings of the 2010 USENIX conference on USENIX annual technical conference*, volume 8, pages 11–11, 2010.