

A Novel Routing Algorithm for Content-based Publish/Subscribe Systems

Guo Shaohui^{1, a}

¹ School of Law and Political Sciences, Xuchang University, Xuchang, Henan, P.R.China

^axcgsh@126.com

Keywords: Publish/Subscribe, Matching Algorithm, Predicate Cover

Abstract. Publish/Subscribe communication model has the characteristics of asynchronous , multipoint communication ,it can makes the participants of communication decoupled in space and time and control flow completely , and can meet the needs of loose communication that the large distributed system asks effectively. This article makes up the selective routing and content-based publish/subscribe model, and put forward a publish/subscribe data model based on Predicate cover, RABPC, it also gives the algorithm of the subscription and release. Through simulation experiment, compared with other existing algorithms, the method is more efficient in subscribe and release time.

Introduction

The Publish/Subscribe system is a kind of system that all participants in the distributed system can interact in the form of Publish/Subscribe [1]. In the Publish/Subscribe system, information produced during the interaction between producers and consumers is called the "events"[2]. The information producers sent the "events" to the middle ware. The information consumers send "subscription conditions" to the content routing network , which express the events in systems the producers interested in ; While the content routing network guarantee the released events will be transmit to all information consumers who interested in it timely and reliably . compared with other communication paradigm, the publish/subscribe has the advantages of loosely coupled, support more communication , as well as well stability , etc , so the current mainstream middle ware platforms provide support for the publish/subscribe communication model[3].

People have made a lot of research about the key technology of the Publish/Subscribe system. in terms of data model , the existent research are mainly based on three types of systems including the theme of the system, based on the Map system and based on XML , and also puts forward some mixed data models . In terms of matching algorithm, the system based on theme does not need to have the corresponding matching algorithm, to the system based on Map and the system based on XML, the people have designed a lot of kinds of efficient matching algorithms, and for other Publish/Subscribe system with complex data model, people have not to find efficient matching algorithms [4]. In terms of routing protocol, most of the existing research organize the events agency network as a tree structure or acyclic graph structure, and then based on the structure; use some strategies for efficient message forwarding. The Publish/Subscribe system is an ideal choice in a highly dynamic Internet and mobile environment because of characteristics of loosely coupled, anonymity and extensibility, etc [5].

This article obtains the analysis of the content routing, carried on the thorough research to predicate overlay routing, and combining the content covered relationship, design a matching algorithm based on heuristic match rule. At the same time, conduct a further study of the scale of store and subscribe, routing algorithm.

RABPC: Routing Algorithm Based on Predicate Coverage

The following three algorithms for the Publish/Subscribe system based on CRN network describe how the system subscribes content (expression), maintenance, subscribe release and matching. selective subscription content routing is implemented based on covering technology, its

principle is that if the subscription S1 cover S2, the algorithm only needs to send S1 to the upstream router, because S1 can on behalf of S2's interest, the subscription which is covered does not need to be sent to all the content router, thereby reducing the size of the router's subscription and routing table size.

Subscribing Algorithm

The most critical problem is how to maintain the relationship between the subscription in the Publish/Subscribe system reasonably. This will involve the regulatory problems about correctness, efficiency and dynamic properties, etc. The algorithm is any one of the content router nodes in the content routing network (CRN), it was configured and run the following subscription routing algorithm.

Algorithm 1: subscription routing algorithm

- step1: Keep waiting to receive from the downstream port of subscription $Adv(S)$;
- step2: Update the abstract tree structure and routing table of router R;
- step3: Determine whether send to the upstream router R_{up} ? Yes, turn step1.3; otherwise, turn step1.4;
- step4: Revise Subscriptions (expression) , and forward in terms of $Adv(S')$;
- step5: do nothing;
- step6: end;

Publishing Algorithm

If the host C4 send operation $Cal(S_4)$ to the upstream router R2, content router R2 will determine whether it allows to execution and sends message to the upstream router (R3) . If the upstream router still has the subscription S4, R2 must send message to the upstream router again.

In order to supporting functions better , when implement a system , each subscription sent to the upstream router are required to do a tag flag ,which can indicate whether the subscription is exist in the upstream router . as situation shown in the above figure, after executing release operation in this router, we still need to check (directed by the flag) upstream router to see if there are any same copies and has not been post .if the same copy still exist in upstream router, $Cal(S')$ must be sent to the upstream router to make the upstream router can still perform and publish . Otherwise, the upper router will store amount of "junk" message which is already be released but be detained due to incorrect perform; the phenomenon will produce in the following situation:

Algorithm 2: release routing algorithm

- step1: Keep waiting to receive publish subscribe message $Cal(S)$ from downstream port;
- step2: Query the structure of subscription's relationship and conduct heuristic matching $match(Pub_e, Tree_s) ?$;
- step3: Calculate and query routing pt_null , and get the collection of ports C_{port} will be forward by Pub_e ;
- step4: Judge if $C_{port} = \emptyset$, Yes , do not do any forward ,No ,turn step1.4 ;
- step5: Forward Pub_e to all the ports of C_{port} ;
- step6: end.

REALISTIC SIMULATION

In order to avoid occupying a lot of bandwidth of local area network (LAN) , each client was designed to send message at a speed of 20 packets, the length of each packet is 128 bytes, among

them ,the data hold 100 bytes , UDP hold the first 8 bytes , IP hold first 20 bytes .each client can convenient to observe and record the data in a network flow , so it let each client send 10000 packets , and let the host virtual client access into the system every 5 s, until all 20 virtual clients have accessed to . The peak of active router port packet in the system can reach 400 packets, in the condition of all the 20 host have accessed to the system and at the rate of 20 packets sent. However, this value is not absolute; it will be affected by the network congestion. Active router side will save the number of packet that every second it process, can draw data graphs according to need to sample.

The result of the experiment and analysis

In this paper, we will conduct two sets of experiments, we will analyzes and contrast from the following four methods, include RABPC, Navie[6],Neon[7] and Tree[8]. The contexts of compare are the subscription size 's impact on the efficiency of the algorithm , the coverage ratio $\alpha\beta$'s influence on the subscription scale of content router , the situation of subscription scale distributed in each router , the experiment about matching event message to the routing 's time , the subscription dimension 's impact on the algorithm .

As we know, in general, if the larger the scale of the subscriptions to be dealt with, the longer time the algorithm spends in processing. Different way that the subscriptions will be stored, different time that the new subscription will be inserted, and event will have different efficiency in the process of matching.

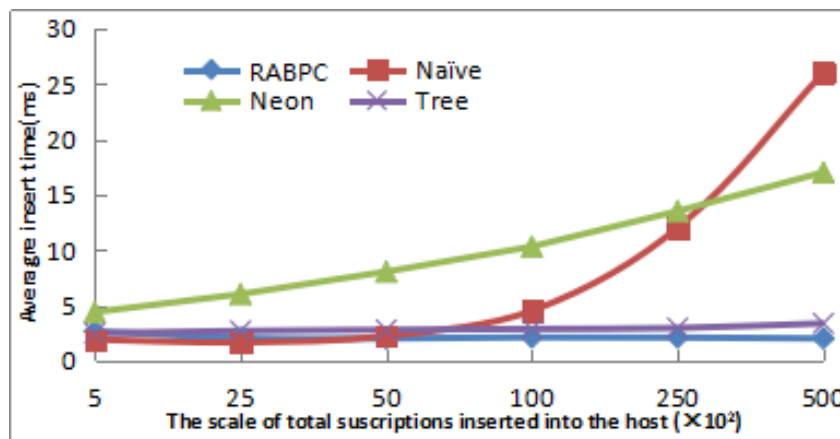


Figure 1: The comparison among the average insertion time

Figure 1 shows that, the average insertion time of RABPC algorithm's subscription maintain between 2.0 ± 0.5 ms, Aguilera's Tree algorithms close to the RABPC algorithm in terms of the average insertion time and property. The Naïve algorithm and Neon were affected by the scale of subscription largely, to ensure there is no repeat in subscription; the two algorithms should spend a lot of time in query, just because the experiment considered about the situation that there is the same state among subscriptions.

Data curve in Analysis figure 2 shows, the average matching time of RABPC algorithm and Aguilera's Tree algorithm is influenced by the scale rarely. when the subscriptions' number less than 10000 articles , Naïve algorithm has the best property ; when the subscriptions' number greater than 10000 articles, the growth rate of Naïve algorithm's and Neon algorithm's average matching time become faster . Conclude from the analysis of data curve, the bigger the subscription's scale is, the greater the property of PCRT.

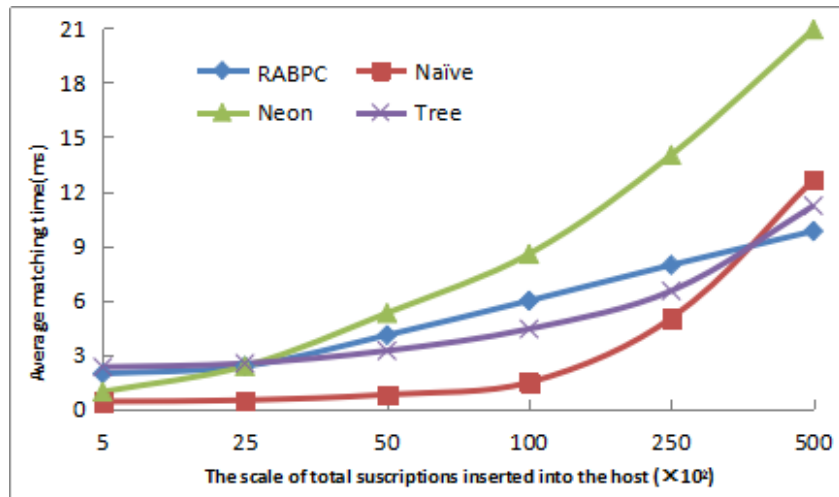


Figure 2: The comparison among the average matching time

CONCLUSION

This paper put forward and realized subscription algorithm, Publish algorithm, heuristic matching algorithm and routing algorithm based on the predicate covering technology, those algorithms makes the Publish/Subscribe system has a high efficiency. At the same time, after make a comprehensive consideration of the weigh among the network performance, time and space, this paper adopted the subscription-based sub-prime covering technology, sub-prime predicate covering techniques, and has a obviously improvement in computational complexity, high space utilization and network property.

Acknowledgements

This work was financially supported by the National Natural Science Foundation of China (6100503).

References

- [1] T. Balikhina, F. Ball, D. Duce. Distributed Virtual Environments-An Active future? In: Proc. of the 20th Eurographics UK Conference. Washington, DC: IEEE Computer Society, 2002. 33-37.
- [2] G Li, S Hou, Jacobsen H. A unified approach to routing, covering and merging in publish/subscribe systems based on modified binary decision diagrams. In Proceedings of 25th ICDCS. Columbus, USA, 2005: 447-457
- [3] M.Petrovic, I.Burcea, H.-A.Jacobsen. S-ToPSS: Semantic Toronto Publish/Subscribe System. In Proceedings of the 29th International Conference on Very Large Databases(VLDB 2003), pages 1101-1104, 2003.
- [4] P.T.Eugster, Pascal A.Felber, Rachid Guerraoui, Anne-Marie Kermarrec. The many faces of publish/subscribe. ACM Computing Surveys, 35(2), pages 114-131, Jun.2003.
- [5] Christian Esposito, Domenico Cotroneo, Setfano Russo. On-reliability-in-publish-subscribe-services[J]. Computer-Networks, 2013, 57: 1318-1321
- [6] M.K.Aguilera, R.E.Strom, D.C.Sturman, M.Asteley, and T.D. Chandra, Matching events in a content-based subscription system. In Proceedings of the Eighteenth ACM Symposium on Principles of Distributed Computing(PODC'99), pages 53-61, 1999.
- [7] Li J, Ji X Q, Liu X, et al. Topic-based resource allocation for real-time publish/subscribe communication systems[C]. Communications and Networking in China (CHINACOM), 2010 5th International ICST Conference on. IEEE, 2010: 1-3
- [8] Françoise Fabret, François Llibat, João Pereira. Efficient Matching for Content-based Publish/Subscribe Systems[R]. Technical report, INRIA, 2000.