DHT based context information disseminating method in autonomic network

Wang Hao
Beijing University of Posts and Telecommunication
Computer Science and technology
Bandbord network research center
Beijing, PRC 100876
wanghao_bupt@163.com

Abstract

In autonomic network, context information given by monitor is the main foundation of decision. In order to make good use of network context information, the dissemination sub-system is needed to be scalable (support large scale of autonomic network) and distributed(deal with issue of single point of failure). In response to these demands, we propose a method to construct a context information dissemination model based on DHT. In our targeted application, an overlay network is built upon the lower network consist of autonomic nodes. The context information can be published and located using DHT lookup mechanism in the structured distributed network environment. To deal with the mismatch between the overlay-level network and physical network, we present a self-adjust scheme to reduce the lookup routing latency during the process agents apply for the context information. In the strategy, we aim to make node themselves adjust node identifiers in a geographically manner. This self-adjust operation only takes place when the autonomic node access to DHT overlay network.

Keywords: *DHT, stretch, context dissemination*

1 Introduction

Decision element is the core function component of control loops which makes decision in autonomic network. Figure 1 shows the framework of the autonomic network system. In order to take decision DE need constant flow information from the network environment. The monitor plays the role of gathering and handling context information, such as link state, topology, service information, QoS characteristics. Traditional method of information management based on central information server cannot afford to support the application of large-scale autonomic network. The context information dissemination sub-system must be of property of robustness, scalable and distributed.

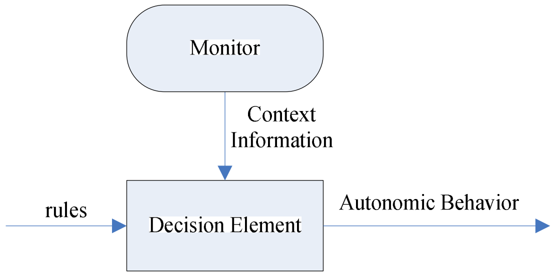

Figure 1: The iris representation with size and rotation invariant

In order to resolve scaling problems, past several years, many research groups have been working on distributed hash table (DHT) technology which can provide a scalable and completely distributed system. In DHT system, both key and value are kept structured in an organized way. All the resources and nodes in DHT network has been mapped to an ID by using hash algorithm. The ID space is large enough to ensure uniqueness in global (always using 160 or 128 bits ID). Each node is responsible for a section of routing information of whole network. In this paper, we proposed a method to construct context information dissemination sub-system which is based on DHT overlay network. We build a DHT overlay layer upon lower network to make autonomic nodes organized in structured form. Each context information record is regarded as a resource in autonomic network system, and be assigned to a ciID(context information identify). Dissemination sub-system uses this ciID to locate context formation.

Several ongoing projects make use of DHT to build overlay network [1], [2], [3], [4], [5], [6], [7]. These projects have made a great progress on DHT lookup performances using the number of application-level hops taken on the path as their measure. But DHT system is constructed in overlay network at the application layer without considering physical topologies. The mismatch between DHT overlay network and physical network could bring extra latency and have a seriously bad influence to context information dissemination sub-system performance. We propose a new self-adjust scheme which can reduce lookup routing latency between autonomic nodes to make dissemination sub-system more efficient and optimize information transmission from source context agent to destination. In our scheme, every autonomic nodes partition themselves into different landmark zones. The lookup cost within a zone is considered far lower than that between different zones. The border information of each zones are recorded. When a new node joins dissemination sub-system, it can execute limited number of “switch” operation with other existed nodes until reaching its suitable location automatically. Using the self-adjust scheme, geographically neighboring autonomic nodes could have familiar dhtIDs.

This paper is organized as follows. In section 2 we describe the model of context information dissemination sub-system. In section 3.1 we describe self-adjust scheme in detail. Simulation methodology and performance evaluation of our scheme are discussed in section 3.2. Finally we conclude our work in section 4.

2 Context information dissemination model

In autonomic network system, control loops and monitor are considered as the basic function components. There has a context agent located in each autonomic nodes. Figure 2 shows that the context agent as the client of context application which is responsible for get context information in network environment for local decision element in control loops to make decision and generate autonomic behavior. Context information application system works in two layers, context application layer and context dissemination layer. Context information are distributed on different nodes instead of central control, context cabin showed in figure 2 is the component as the server of context application that stores context information of the autonomic network. Each autonomic node contains both context agent and context cabin, which means the node play the role of server as well as client of the context application.

Context application layer mainly resolve the problem of end-to-end communication between context agent and context cabin. Context in autonomic network system are described by XML. Context agent map context record to a ciID using hash algorithms based on information key, and use “ContextStore” operation to send context information to context cabin for storing, while use “GetContext” operation to get context information stored in context cabin.

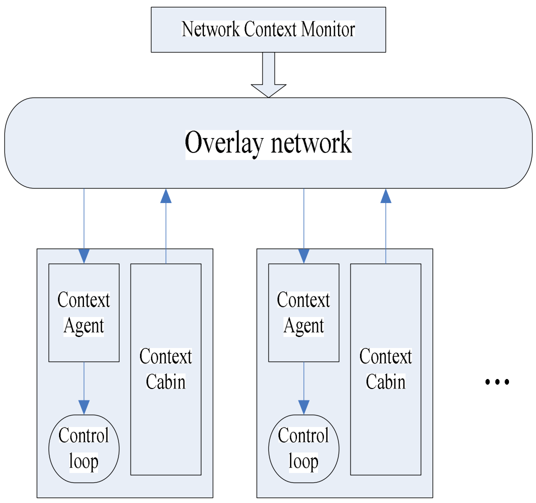

Fig 2 context agent and context cabin in autonomic node

We mainly explore the problem of context dissemination layer. Dissemination layer provide a distributed context store environment for application layer to locate information. We construct an overlay-level network to connect all the autonomic nodes in dissemination sub-system. Here we use DHT to work as the overlay-level network. DHT overlay network has been proved to be available for large-scale network application containing several millions of nodes. The path from context agent to context cabin is routed over DHT network, context application layer is only concerned about problem on end-to-end context information application (problem about communication format, context information expressing, handling, store and transmission) requiring no form of knowledge about where context cabin located. Autonomic nodes connect with an existed node in the dissemination sub-system to join and are assigned a dhtID randomly as their unique identify in the system. The interface between context dissemination layer and context application layer is “GetLocation” operation with ciID as input and destination node’s value (always be ip address) as result. This operation afford location of the context cabin where the information stored when context agent apply for context information, and choose suitable context cabin to receive when publishing context information. The “GetLocation” operation follows DHT lookup mechanism. Take Chord [1] as an example, each autonomic nodes maintain a routing information table. Node x record the successor information of the nodes ($x+2^i$), $i\in[1, n]$. Lookup message routed by overlay-level hop recursively until converged to destination.

In order to deal with the problem of context information unavailable caused by node’s failure for different reasons, we modify the process of publishing information to make the dissemination sub-system more robust. In DHT system, resource always located on successor node in the id ring and when a node quit from the ring, all of the lookup message will be converged to its successor node. Make use of the features above, we keep additional copy of information as a backup in successor nodes. The copies will not be accessed while the node which keeps original information is alive.

Once the context agent gets a context record marked with “backup” symbol, which means the original context cabin has changed to be unavailable, another copy of context record will be sent and stored on the next successor node. This strategy can be supported by most of DHTs and take effect without modifying current lookup mechanism.

3 Self-adjust scheme and Experiment

3.1 Self-adjust scheme

In most of DHT algorithms, identifies of nodes are chosen complete randomly, and the structured relation between nodes are established based on these identifies. Our scheme aims to make physical neighboring nodes be arrayed continuously in the identifier ring instead of dhtID choosing independent of geographically location. Note that the uneven distribution of nodes in different regions and resources are published based on random ciID, as a result , binding identifies to the given interval in identify space artificially will bring problem of load unbalance among the different nodes, which means that the nodes in low-density areas will hold more <key, value> information of context than the nodes in high-density areas. To avoid the problem above, we make nodes to choose their dhtIDs randomly at the beginning and then self-adjust their logical position by switching their dhtIDs with other nodes already existed in the network. In the paper, we take Chord as an example.

We make use of landmark [11] to partition autonomic nodes to different zones. Each landmark zone hold a continuous interval in identifier ring and is consist of the nodes which belong to the landmark zone according to round-trip time to the landmark nodes. The border nodes of the landmark zones are held globally. It is sufficiently to record the nodes of max number clockwise in identifier ring. Figure 3 shows that landmark zone 2($Lzone_2$) holds the interval (bn_1, bn_2]. $Lzone_i$.bn is defined as the border node of the landmark zone i, while $Lzone_i$.interval is defined as the interval of the landmark zone i and $Lzone_i$.id is defined as the identifier of Lzone i. Our scheme assumes that latency within the same zone is considered far lower than that cross different zones. It is clearly that reducing the occurrence of routing cross different zones and taking path zone-by-zone will be helpful for significantly lowing down “GetLocation” response time.

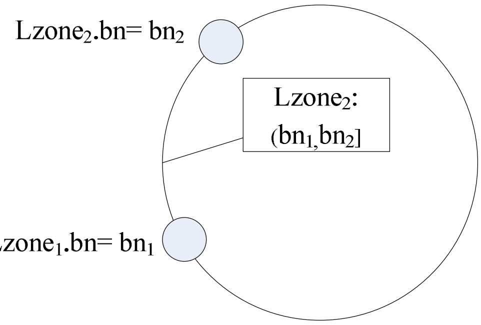

Fig3. interval of the landmark zone in identifier ring

Under steady statement of network, autonomic nodes locate logically in the landmark zone they belonged and the path taken by “GetLocation” follow the sequence of Lzone. At this point, we extend the join algorithm to gatherround the nodes neighboring geographically. Figure 4 shows the pseudocode of join algorithm. Node n waits for join the network. At first the node n get a dhtID randomly (hash algorithm should help to map node’s information to identifier). Then prob the round-trip time between current node and landmark nodes to measure the landmark zone n.lzone_id node n belonged. $Lzone_{cur}$ is current holds the interval where node n locates. If $Lzone_{cur}$ is not the one node n belonged, which means nodes in $Lzone_{cur}$ have long distance to node n in physical topologic, the self-adjust scheme take effect. Node n will switch identifier with $Lzone_{cur}.bn$. the “switch” will relocate node’s position in identifier ring. Whereupon the routing information will also be switched each other, and <key,value> information stored before will move to other side. Note that $Lzone_{cur}.bn$ is also moved to new position without leaving $Lzone_{cur}$, so it is necessary to choose $Lzone_{cur}.bn$’s directly processor node as the border node of $Lzone_{cur}$ instead (update_bn($Lzone_{cur}$)), while finding the processor node can be support by almost all of the DHTs. Since the movement of border node, $Lzone_{cur}.bn$ will lose a section of interval. Self-adjust operation will be proceeded recursively until node n locates itself in the scope of $Lzone_{n.lzone\_id}.interval$. It can also be concluded that $Lzone_{n.lzone\_id}$ extends its interval followed by join of node n.

```
//node n wait for join the network
node_join(n)
    n.id = getrandom_id();
    while (n.lzone_id !∈ Lzone_cur.interval)
        switch(n, Lzone_cur.bn);
        update_bn(Lzone_cur);
        next_lzone(Lzone_cur);
    return;
```

Fig4.The pseudocode for node to join the overlay network

Different from some other algorithms, in our self-adjust scheme, the times of switch is foreseeable and the self-adjust will be taken with only a few number of times. It is obviously that if $Lzone_{start}$ and $Lzone_{n.lzone\_id}$ have a distance of t in clockwise, switching with times of t will bring network to steady statement, in the other words, the node n reaches the correct position. Suppose the number of landmark nodes is l, as a result the number of landmark zones y=l!. Assume that the autonomic nodes belong to each landmark by the same probability of 1/y. It can be inferred that the average switching number of times is

$$ans = \sum_{i=0}^{y-1}\left(i*\frac{1}{y}\right) = \frac{y-1}{2}.$$

3.2 Simulation

We use GT-ITM tool which is used to build Transit-Stub topologic to construct the physical topologic . The transit area is considered as backbone network, while the stub area as the edge of the network. Each DHT node is mapped to an node on the transit-stub network. We take Chord as an example and use C++ to build Chord and the optimize algorithms.

The stretch is used to evaluate the match level between overlay network and physical topologic. The stretch is equal to the average logical link latency normalized by actual average physical link latency. Figure 5 shows the latency stretch for increasing numbers of DHT nodes. The different lines with different colors also shows the latency stretch when different number of landmark nodes exists in the system.

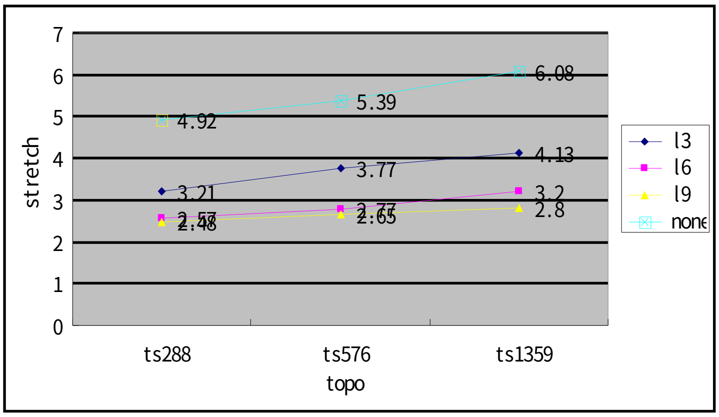

Fig5.The latency stretch

4 Conclusion

This paper express a new idea for context dissemination in autonomic network based on DHT. This method can support the large scale of the network, and provide a distributed dissemination mechanism to deal with the single server unavailable problem. Meanwhile, this paper propose a self-adjust scheme over DHT network. This scheme make a obviously improvement of the stretch when using DHT technology, mismatch between the overlay network and physical topologic , and reduce the latency on actual application system.

References

[1] Chord: A scalable peer-to-peer lookup protocol for Internet applications Stoica, Ion (Computer Science Division, University of California); Morris, Robert; Liben-Nowell, David; Karger, David R.; Kaashoek, M. Frans; Dabek, Frank; Balakrishnan, Hari Source: IEEE/ACM Transactions on Networking, v 11, n 1, February, 2003, p 17-32

[2] A scalable content-addressable network Ratnasamy, S. (Dept. of Elec. Eng. and Comp. Sci., University of California, Berkeley); Francis, P.; Handley, M.; Karp, R.; Shenker, S. Source: Computer Communication Review, v 31, n 4, 2001, p 161-172

[3] Kademlia: A Peer-to-peer Information System Based on the XOR Metric Petar Maymounkov and David Mazi`eres Electronic Proceedings for the 1st International Workshop on Peer-to-Peer Systems (IPTPS '02)

[4] Rowstron A, Druschel P. Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems. In: Guerraoui R, ed. Proc. of the 18th IFIP/ACM Int'l Conf. on Distributed Systems Platforms (Middleware 2001). Heidelberg: Springer-Verlag, 2001. 329?350.

[5] B．Zhao，J．D．Kubiatowicz，A．D．Joseph．Tapestry: An Infrastructure for Fault-tolerant Wide-area Location and Routing ．Technical report UCB／CSD 0l l 141．UC Berkeley．April 2001

[6] Malkhi D, Maor M, Ratajczak D. Viceroy: A scalable and dynamic emulation of butterfly. In: Ricciardi A, ed. Proc. of the 21st Annual Symp. on Principles of Distributed Computing. New York: ACM Press, 2002. 182?192.

[7] Shen HY, Xu CZ, Ghen G. Cycloid: A constant-degree and lookup-efficient P2P overlay network. In: Panda DK, Duato J, Stunkel C, eds. Proc. of the 18th Int'l Parallel and Distributed Processing Symp. (IPDPS 2004). New York: IEEE Press, 2004. 26?30.

[8] Routing Algorithms for DHTs Some Open Questions I. Stoica, D. Adkins, S. Ratnasamy, S. Surana, S. Shenker and S. Zhuang Proceedings of the First International Workshop on Peer-to-Peer Systems (IPTPS) 2002, March 2002

[9] M. Castro, P. Druschel, Y. C. Hu, and A. Rowstron. Exploiting network proximity in distributed hash tables. In InternationalWorkshop on Future Directions in Distributed Computing (FuDiCo), 2002.

[10] Ratnasamy S, Handley M, Karp R, Shenker S. Topologically-Aware overlay construction and server selection. In: Proc. of the IEEE INFOCOM. New York: IEEE Press, 2002. 1190?1199.

[11] NG, E., AND ZHANG, H. Towards global network positioning. In Proceedings of ACM SIGCOMM Internet Measurement Workshop 2001 (Nov. 2001).

[12] Ren S, Guo L, Jiang S, Zhang X. SAT-Match: A self-adaptive topology matching method to achieve low lookup latency in structured P2P overlay networks. In: Proc. of the 18th Int'l Parallel and Distributed Processing Symp. (IPDPS 2004). New York: IEEE Press, 2004. 83?91.

[13] A generic approach to making P2P overlay network topology-aware Qiu, Tong-Qing (State Key Laboratory for Novel Software Technology, Nanjing University); Chen, Gui-Hai Source: Ruan Jian Xue Bao/Journal of Software, v 18, n 2, February, 2007, p 381-390

[14] http://wiki.efipsans.org/