Dynamic Wide Area Sever Deployment System with Server Deployment Policies

Yuko Kamiya† Toshihiko Shimokawa†† Fuminori Tanizaki††† and Norihiko Yoshida††††

†Graduate School of Information Science, Kyushu Sangyo University,
††Faculty of Information Science, Kyushu Sangyo University
†††NTT West Fukuoka Blanch
††††Information Technology Center, Saitama University

Summary
Using virtual machine, it is possible to deploy servers in a wide area dynamically. In consequence, it is possible to expand network bandwidth as well as processing power. On deploying servers in a wide area, it is an important issue where to deploy new server. In this paper, we propose policies to decide a location of new server for dynamic wide area server deployment. We call this policy *server deployment policy*. We also propose system architecture of dynamic wide area server deployment system that can use above policies.

Key words:
Virtual Machine, Dynamic Server Deployment, Wide Area Server Deployment, Server Deployment Policy

1. Introduction

In recent years, cloud computing is remarkable as a scalable system. Many cloud-computing systems deploy new virtual or physical machines to expand these processing powers. However, it is difficult to expand network bandwidth. It is because many of existing systems deploy new machine only in a local area typically inside an iDC. We can expand network bandwidth by deploying new machines in a wide area. vMatrix[1], XenoServer[2] and Server Proliferation [3,4] deploy new server in a wide area. These researches deploy physical machines in a wide area in advance, select one physical machine to execute a virtual machine, and they deploy new virtual machine to the physical machine. However, in these researches, there is only simple fixed rule to decide a location of new server. . In this paper, we propose policies to decide location of new server for dynamic wide area server deployment. We also propose system architecture of dynamic wide area server deployment system that can use flexible policies.

We explain related works about dynamic wide area server deployment system in Section 2. In section 3, we propose architecture of dynamic wide area server deployment system with server deployment policies. Section 4 describes a design of dynamic wide area server deployment system with server deployment policies. Finally, we conclude this paper in section 5.

2. Related Works

There are some researches of dynamic wide area server deployment system using virtual machines.

vMatrix is a solution for the distribution and replication of dynamic content. vMatrix is a network of real machines running virtual machine monitor software, hence allowing server virtual machines to be moved among the physical machines.Vmatrix can improve the response time perceived by end users, overall availability and uptime, and on-demand replication to absorb flash crowd requests. vMatrix also can reduce the overall bandwidth consumed in the network.

XenoServer platform focuses on service deployment; this is the step where users, after having selected a number of XenoServers on which the service is to be deployed, proceed to contact the XenoServers to configure and start the virtual machines that will accommodate the service components, and to launch the service components themselves.

However, existing dynamic wide area server deployment system have only simple fixed rule that is about where to deploy the virtual machine.

3. Architecture of Dynamic Wide Area Server Deployment System with Server Deployment Policies

In this section, we describe our two proposals:
- System architecture of dynamic wide area server deployment system that can use flexible policies
- Server deployment policies for dynamic wide area server deployment

In this paper, we call the rule that is used to decide a location of new server *server deployment policy*. In 3.1,

Manuscript received October 5, 2010
Manuscript revised October 20, 2010

we describe system architecture that can use flexible server deployment policies.

3.1 Dynamic wide area server deployment system

We propose system architecture of dynamic wide area server deployment system that can use flexible policies.

Typically, existing server deployment system uses two types of servers. One is *storage server*. This server stores virtual machine image files. The other is *execution server* that executes virtual machines. Virtual Machine Monitor (VMM) is installed in execution servers. New virtual machines are invoked on this server using virtual machine image files received from storage server. As we mentioned above, this architecture has two problems. One is that there is only simple fixed policy to decide location of new servers. The other is that this architecture cannot use flexible server deployment policies.

To be able to use flexible server deployment policies, two functions - collect information and decide a location of a new virtual machine - are needed.

We propose new system architecture, which is shown in Figure 1. This architecture has new two kinds of servers to decide a location of a new virtual machine. One is *observation server* and the other is *control server*. Observation server collects several kinds of information. Control server uses the information that is collected by observation sever to decide the location of new server.

The main components of this architecture are follows:

- Observation server collects several kinds of information of execution server and storage server. The collected data is used for monitoring and deciding a location of new virtual machine (arrow 1).
- System user must do two things. One is that system user sends virtual machine image files to storage server (arrow 2a). The other is that system user tells a desirable policy to control server (arrow 2b).
- Control server selects a suitable execution server based on the system user's policy with information collected by observation server (arrow 3).
- Surrogate server deploys virtual machine image files to the execution server selected by control server and the virtual machine is executed on the execution server (arrow 4).
- Request navigation system navigates requests of clients to servers including a new server.

3.2 Typical server deployment policies

In this section, we describe five typical server deployment policies.

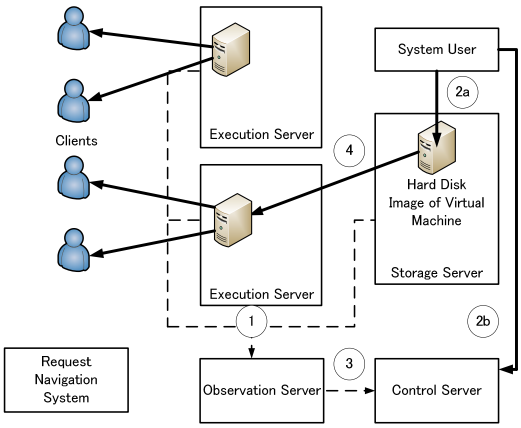

Fig. 1 Architecture of Dynamic Wide Area Server Deployment System

3.2.1 Largest remaining capacity of processing power

When service capacity is insufficient due to increasing number of requests, control server adds new virtual machine. Thus, the processing power can be expanded. In this case, suitable server selection policy is to select an execution server that has the largest remaining capacity of processing power.

3.2.2 Largest remaining capacity of network bandwidth

When network bandwidth is insufficient, control server expands new network bandwidth. To expand network bandwidth, VMs are deployed on the network that differed from existing VMs are on. In this case, suitable server selection policy is to select an execution server that has the largest remaining capacity of network bandwidth.

3.2.3 Shortest network distance between storage server and execution server

By adding new virtual machine, it can expand processing power and network bandwidth. However, in case which load of the server expands rapidly, the new virtual machine should be added as soon as possible. In this case, suitable server selection policy is to select a nearest execution server from storage server for rapid deployment of virtual machine.

3.2.4 Region where largest number of clients

By deploying servers in a wide area, it is possible to expand network bandwidth. However, network bandwidth shortage may be occurred in some cases, e.g., network between new virtual machine and clients are congested. Control server identifies locations of clients from their source IP addresses. Then, control server identifies the region where largest number of clients is. After that, control server selects the execution server that is located the nearest from the region. We can use two type of distance to decide nearest region. They are network distance and geographical distance. An advantage of shortening a network distance between clients and server is obvious. On the other hand, an advantage of shortening a geographical distance between clients and server is an expectation that the network distance between clients and server is near also. For example, it is rational to select the execution server inside Europe but not on North America and/or Africa toward the access from Europe. In this case, suitable server selection policy is to select a nearest execution server from clients.

3.2.5 System user's opinion

This policy uses system user's opinion as a criterion. For example, if economical cost is not same among execution servers, there is a possibility that not only the performance but also the cost is a better criterion. At this time the system users wants to choose execution servers whose costs are low. On the other hand, some system users want to do traffic engineering. Accordingly, the system user selects execution server manually.

4. Design of Dynamic Wide Area Server Deployment System with Server Deployment Policies

In this chapter, we describe the design of dynamic wide area server deployment system with server deployment policies based on the architecture shown Section 3. To implement the system, we use Server Proliferation we proposed as a basis. We describe about Server Proliferation in section 4.1. Section 4.2 proposes design of the system. In section 4.3, we describe how to implement policies.

4.1 Server Proliferation

Server Proliferation realizes expanding and shrinking processing power and network bandwidth of server system dynamically. To realize this, it increases and decreases a number of servers in a wide area dynamically. Figure 2 shows architecture of Server Proliferation. We introduce two types of the servers in Server Proliferation. One is *Execution Server* (ES) and the other is *Distribution Server* (DS). Server Proliferation deploys physical machines that are installed virtual machine monitor all over the Internet in advance. These physical machines execute virtual machines on them. We call these physical machines *Execution Server*. The other server is *Distribution Server*. DS stores hard disk images of virtual machines. In Server Proliferation, services (ex. Web server, Streaming server and so on) are executed inside virtual machines. When a new virtual machine is required, a hard disk image of virtual machine is distributed from DS to one of the ES. The distributed virtual machine is executed on the ES. In this architecture, DS can become a bottleneck. However, it is easy to use multiple DS. Therefore, it does not become a bottleneck.

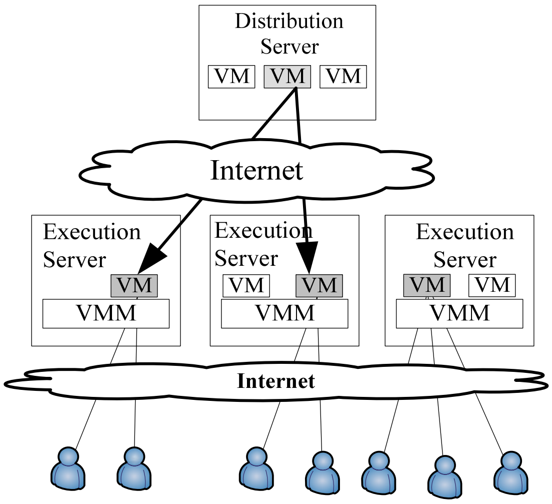

Fig. 2 Architecture of Server Proliferation

Cloud computing makes it possible to increase a number of servers dynamically. By expanding servers, it is possible to use the CPU and the network of the expanded server. Thus processing power and network bandwidth is expanded. Therefore cloud-computing systems can expand the processing power and network bandwidth. However, compared with processing power, it is difficult to expand network bandwidth. This is due to network bottleneck. Typical cloud-computing system is constructed in an iDC (Internet Data Center). The uplink network of the iDC may become network bottleneck of the cloud-computing system.

By contrast, Server Proliferation can expand both of processing power and network bandwidth. It is because it can deploy servers in a wide area; therefore, deployed servers can use different uplink network each other. As it turned out, it is possible to expand network bandwidth of the system.

Server Proliferation uses virtual machine as a basis. It is because virtual machine is easy to increase and decrease dynamically. Moreover using virtual machine can reduce cost since physical machines can be shared with other system that uses virtual machines. It is possible to execute another virtual machine besides virtual machine executed by Server Proliferation. We can say that Server Proliferation is high-cost performance.

4.2 Design of dynamic wide area server deployment system using Server Proliferation

As we described in Section 4.1, Server Proliferation provides the way to deploy a virtual machine dynamically. However it lacks capability to decide timing and location of deploying virtual machine. Therefore, we introduce Observation Server (OS) and Control Server (CS) in addition to Execution Server (ES) and Deployment Server (DS) in Server Proliferation. OS collects several kinds of metrics using server deployment policy. For example, it collects the CPU load and network traffic of ESs, calculates the distance from ESs to DSs. CS controls all over our system. To keep up with the deploying a new virtual machine, flexible request navigation system is required. Our system uses Tenbin [5] as a request navigation system.

4.3 Design of typical server deployment policies

4.3.1 Largest remaining capacity of processing power

To realize the policy described 3.2.1, observation server collects CPU load average of each execution server by using hrProcessorLad MIB [6] periodically. In addition to this, observation server collects processing power (e.g. MIPS value of CPU) of each execution server. Control server calculates remaining capacity of processing power from these information.

4.3.2 Largest remaining capacity of network bandwidth

To realize the policy described 3.2.2, observation server collects the number of input and output packets of external network link of each execution server by using SNMP[7] periodically. In addition to this, observation server collects network interface speed of external network link of each execution server. Control server calculates remaining capacity of network bandwidth from these information.

4.3.3 Shortest network distance between storage server and execution server

The policy described 3.2.3 uses the network distance between storage server and execution server as a criterion. Some metrics can be used to calculate the distance. Typically, Round Trip Time (RTT) of ICMP echo and echo reply or TCP syn and ack packet can be used. The distance between storage server and execution server may change at any time depending on the network situation. Therefore storage server measures the distance from each execution server periodically.

4.3.4 Region where largest number of clients

To realize the policy described 3.2.4, control server identifies locations of clients from their source IP addresses. Then, control server identifies the region where largest number of clients is. After that, control server selects the execution server that is located the nearest from the region. We can use access log and flow data to identify IP addresses of clients. In this research, we use flow data. There are three reasons. First one is that there are some de fact standard ways to retrieve flow data from route remotely. For example, sFlow[10] and IPFIX[11] are available. Second one is that sFlow and IPFIX can sample flow data. Therefore we can reduce access data. The last one is that when ES face to heavy load, it is hard to send access log data to OS. On the other hand flow data can retrieve from router. Therefore it is easy to retrieve flow data even while heavy load situation.

To identify network location of clients, we use AS[8] information of clients' IP addresses. We can identify AS where the clients are from its IP address by BGP routing information, e.g. looking glass [9]. To identify geographical location of clients, we use country information of clients' IP addresses. We can identify country where the IP address is by IP address allocation information of Regional Internet Registry (RIR). RIR is an organization that manages the allocation and registration of IP addresses within a particular region of the world. There are five RIRs: ARIN, RIPE NCC, APNIC, LACNIC and AfriNIC. ARIN manages North America, RIPE NCC manages Europe, the Middle East, and Asia. APNIC manages Asia and the Pacific area, LACNIC manages Latin America and the Caribbean Area, and AfriNIC manages Africa. As we said above, we use country as a unit of region where clients and execution servers are. Therefore, we have to estimate distance between two other countries. We can use two type of information to estimate the distance. One is geographical distance. The other is whether client's and execution server's IP addresses are allocated from same RIR or not. If they are allocated from same RIR, they may locate same or near region. Moreover, this information is very easy to use.

5. Conclusion

There are two problems in existing dynamic wide area server deployment system. One is that there is only simple fixed rule to decide a location of new servers. The other is that architecture of existing system cannot use flexible server deployment policies. We tackled these problems. In this paper, we proposed typical policies to decide location of new virtual machines for dynamic wide area server deployment. We called this policy *server deployment policy*. We also propose system architecture of dynamic wide area server deployment system that can use above policies.

In the future, we plan to implement and evaluate the system we proposed. At the same time, we plan to consider other policies.

Acknowledgments

This research was supported in part by MEXT in Japan under Grants-in-Aid for Scientific Research on Priority Area 21013008, and by JSPS in Japan under Grants-in-Aid for Scientific Research (B) 20300024.

References

[1] A.A.Awadallah and M.Rosenblum. "The vMatrix: A network of virtual machine monitors for dynamic content distribution." In Seventh International Workshop on Web Content Caching and Distribution, August 2002.

[2] Kotsovinos, E., Moreton, T., Pratt, I., Ross, R., Fraser, K., Hand, S., Harris, T.. "Global-Scale Service Deployment in the XenoServer Platform", 1st Works. On Real, Large Distrib. Sys.—WORLDS'04, December 2004.

[3] Yuko Kamiya, Toshihiko Shimokawawa, "Scalable Server Construction Method Based on Virtual MachineTransfer and Duplication." Proceedings of International Multi-Conference on Engineering and Technological Innovation IMETI 2008, Vol.2, pp273-278, June 2008.

[4] Yuko Kamiya, Toshihiko Shimokawa, Norihiko Yoshida, "Scalable Server System Based on Virtual Machine Duplication in Wide Area", Proeedings of The 3rd International Conference on Ubiquitous Information Management and Communication, pp.432-436, January 2009.

[5] Toshihiko Shimokawa, Norihiko Yoshida, Kazuo Ushijima, "Server Selection Mechanism with Pluggable Selection Policies", Electronics and Communications in Japan, Part III (Fundamental Electronic Science), Vol.89, No.8,pp.53-61, July 2006

[6] P.Grillo, and S.Waldbusser(Eds.). "Host Resources MIB." RFC 1514, 1993.

[7] K.McCloghrie and M.Rose, "Management Information Base for Network Management of TCP/IP-based Internets: MIB-II" , RFC 1213, 1991

[8] J.Hawkinson and T.Bates," Guidelines for creation, selection, and registration of an Autonomous System(AS)", RFC1930, 1996

[9] looking glass http://www.lookinglass.org/

[10] P.Phaal, S.Panchen, and N.McKee. "InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks", RFC3176, 2001

[11] B. Claise, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information", RFC5101, 2008

Yuko Kamiya received the B. Information Science and the M. Information Science from Kyushu Sangyo University in 2006 and 2008 respectively, where she is currently working toward the Ph.D degree. Her research interests include load balancing and wide area networking. She is a member of Information Processing Society of Japan.

Toshihiko Shimokawa received the B.E. and M.E. degree in computer science and communication engineering and the Ph.D. degree in computer science from Kyushu University in 1990, 1992 and 2001. He is a professor of information science in the Department of Information Science at Kyushu Sangyo University since 2007. His research interests include parallel and distributed computation, and wide area networking. He is a member of Information Processing Society of Japan and The Institute of Electronics, Information and Communication Engineeers.

Fuminori Tanizaki joined NTT West in 1986. Since he joined NTT West, he has been engaged in Network & Server Operation. His research interests include IPv6, Video Streaming, DNS, and BSD UNIX.

Norihiko Yoshida received the M. Eng. degree in mathematical engineering and information physics in 1981 from the University of Tokyo, and the Dr. Eng. degree in computer science and communication engineering in 1990 from Kyushu University. He is Professor of computer systems at the Department of Information and Computer Sciences since 2002, and Director of Information Technology Center since 2008, both of Saitama University . His research interests include system design methodologies, and parallel and distributed computation. He has been served on program committees and organizing committees of various international conferences and symposia. He is a member of IPSJ, IEICE, JSSST, ACM and IEEE.