

Fujitsu Develops Technology Employing 10 Gbps Virtual Switch to Substitute for On-Server Virtual Switch Functions

11 June 2010

Makuhari Messe in Chiba, Japan.

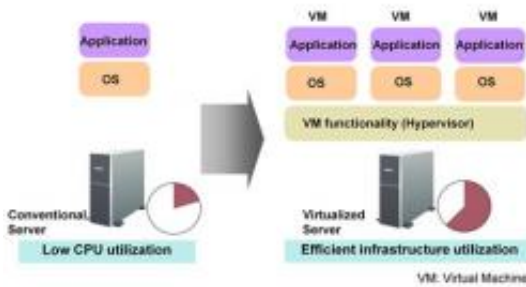


Figure 1: Infrastructure optimization through server virtualization

Fujitsu Laboratories today announced the development of technology for use in virtual cloud computing networks that manages virtual switching functionality - which has traditionally been handled using server-level software - using dedicated external switching hardware. This new technology is the world's first such technology to run at 10 Gbps while supporting standard protocols.

The technology enables users to lighten the CPU processing loads of servers that deliver virtual switching functionality, thereby allowing a higher number of virtual machines to operate on a single server than was previously possible. In addition, by automatically detecting virtual machine migration and preemptively configuring network settings, this technology enables virtual servers to be migrated without impacting the end-user's business operations. These technologies help cloud service providers to reduce hardware costs by up to half, and reduce operating costs as well, thus contributing to a lower total cost of ownership (TCO).

This technology will be exhibited as reference at INTEROP TOKYO 2010, held from June 9 - 11 at

Server virtualization has emerged as one of the core technologies for implementing cloud computing. With server virtualization, a single physical server runs multiple virtual machines, allowing each virtual machine to run its own operating system (OS) and business system applications simultaneously. By consolidating tasks previously performed by multiple servers onto a single server, users can effectively utilize each server's CPU, therefore leading to lower hardware costs.

While virtual machines and physical machines use networks in the same way, each virtual machine needs to be connected to a virtual switch on the server. Because the virtual switch is a switch that has been implemented using software, the server creates a virtual network within itself that transmits packets between virtual machines and also transmits packets between itself and a physical switch.

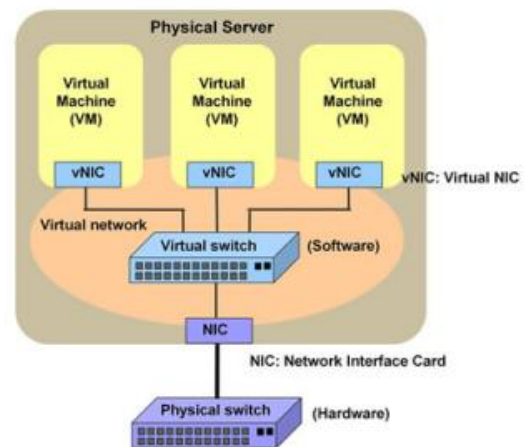


Figure 2: Communications between virtual machines over

virtual switch

Hereon, in order for cloud computing continue to evolve with higher reliability and security, it is anticipated that there will be higher demand for increased functionality in virtual switches, such as functions to prevent users from impairing the performance of other users who share a single physical server, to constantly monitor the behavior of virtual networks, and to manage server performance and failures.

In managing the operations of cloud computing systems, the management of conventional physical networks and the management of virtual networks that employ virtual switches have been considered separate tasks - currently there is no way to centrally manage the two. For example, even during a "live migration", in which a virtual machine is migrated without shutting down running applications, network settings for the virtual switch and physical switch need to be configured as part of the virtual machine's migration. Using conventional methods, this results in a temporary break in communications after the migration.

In addition, quality-of-service (QoS) functions and security functions - which were conventionally implemented on existing physical switches - impose high processing loads. Attempting to implement these QoS and security functions through software on a virtual switch imposes a high load on the server's CPU, thereby reducing CPU resources available for allocation to virtual machines.

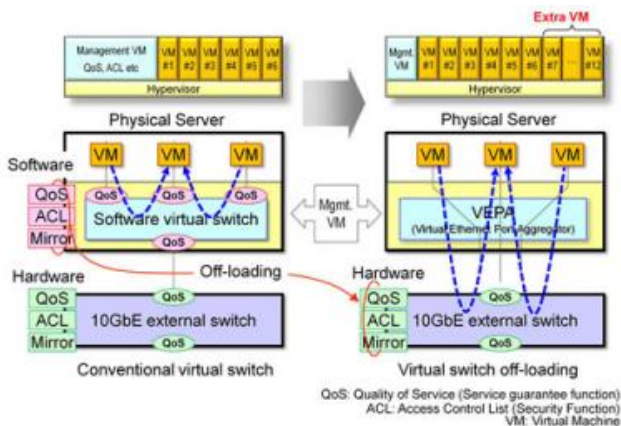
Figure 3: Effects of virtual-switch offloading

The IEEE 802.1Qbg working group has begun to develop standards to address the aforementioned issues. The following two technologies were implemented using the offloading function of Fujitsu's 10Gbps switch chip, and for the first time in the world have been proven to be effective. Following is an overview of the novel technologies:

1. Virtual switch offloading technology: Technology to bypass virtual switching and substitute it with an external hardware switch. By reducing the CPU load associated with running the virtual switch as software, the server can run more virtual machines. Furthermore, the virtual switch can be centrally managed by the physical switch, thereby simplifying network management.
2. Network settings data management technology: Technology that automatically configures network settings information (port profiles) in conjunction with virtual-machine live migration. By adding to the management VM Fujitsu's proprietary functions that detect the initiation of a [virtual machine](#) migration, network information settings can be configured before the migration completes (patent pending), allowing for live migrations with no interruption in communications.

Because the new technology reduces CPU burden of the virtual switch, the CPU can run more virtual machines (up to two times as many compared to conventional virtual switches, Figure3), thus resulting in lower server hardware costs.

In addition, by employing technology that handles server management in conjunction with network settings data, operators are able to automate switch provisioning in which the arrangement of virtual machines is modified, thereby simplifying system operations management. The technology also enables live migrations of virtual machines without affecting the end-user's business operations.



Provided by Fujitsu

APA citation: Fujitsu Develops Technology Employing 10 Gbps Virtual Switch to Substitute for On-Server Virtual Switch Functions (2010, June 11) retrieved 24 January 2021 from <https://phys.org/news/2010-06-fujitsu-technology-gbps-virtual-substitute.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.