Technology to Mitigate Impacts of Running Virtual Operating System (OS)

Yasuhiro Otake

Keywords Existing system, Life extension, Virtualizing function, Expansion board

Abstract

When a Personal Computer (PC) in current use is replaced with a cutting-edge new model due to life extension of control systems, the Operating System (OS), in some cases, may be forced to upgrade to the most up-to-date OS. If the OS is upgraded, existing application programs must also be updated and they often be required additional costs.

If virtual technology is utilized in such a case, an adverse effect on applications can be relieved because an OS of former generation can work with the latest PC models. In such a case, however, not just any expansion board can be used from the guest OS running Virtual Machines (VMs). This can be an obstacle for the replacement of a PC which uses the expansion boards.

Against such background, we tried to verify the feasibility of the latest PC with a guest OS running VMs. We realized this by making a redirect mechanism to use an expansion board from an application on the guest OS.

1 Preface

Control systems using our products tend to be used for a long period of time. For this reason, when maintaining existing Personal Computers (PCs) or deploying a system to another project, it is often necessary to replace the existing PCs with the latest models. At this time, if the Operating System (OS) used in a PC model currently used is not compatible with the latest PC model, an OS upgrade to the latest generation OS will be required. Although the latest OS maintains some compatibility to the older generation OS to some extent, it is often necessary to modify the application programs. For this reason, an OS upgrade to the latest generation incurs additional costs. By using Virtualization Machines (VMs), however, we make it possible to run older-generation guest OSes using VMs on the latest PC model.

Additional costs are kept low because the impact on the application is reduced.

There are, however, limitations on the virtual hardware configuration that can be provided by virtualization, and there is also a weakness that expansion boards cannot be handled.

For example, many of the control systems in which our products are involved use a legacy internal infrastructure of the factory. When replacing the PC in such a system, it is necessary to replace the legacy internal infrastructure of the factory, such as communication. The same is true when using a General Purpose Input/Output (GPIO) expansion board.

The availability of expansion boards in the VMs is, therefore, a major barrier to a PC upgrade to latest model which uses the VMs.

As such, we constructed a mechanism that allows application programs that use our own communication transmission line system at work for the PC currently in use to work under a guest OS without modification.

As a result of the actual measurements, we could confirm that the overhead remained at around 1 ms. The sum of all delay elements is called “overhead”. It involves not only communication overhead but also the processing time of the virtualization engine, and the Central Processing Unit (CPU) processing time. This paper introduces the feasibility of PC upgrade to the latest generation model using the guest OS running VMs.

2 Support of Expansion Board in Virtualized Environments

2.1 Types of Virtualization Technology

There are two main types of virtualization technology, the host type and the hypervisor type. There are two types of hypervisor type: the monolithic kernel type and the microkernel type.

The host type is a method that provides a virtual PC environment as one of the applications and has been widely used since the early days of virtualization due to its simple structure. It is easy to introduce, but the overhead is so large that it is not suitable for use in control systems.

The monolithic kernel type is a method that provides a virtual PC environment only with a heavy-duty virtualization engine which is rarely adopted due to its complicated structure. Although the overhead is very small, it is difficult to introduce due to many operational constraints.

The microkernel type uses the host OS driver to reduce the mass of the virtualization engine, and is used in many OSes. It also has a small overhead, making it suitable for use in control systems.

In particular, the microkernel-based Microsoft Hyper-V has embedded OSes like Windows 10 and Windows Server 2012 or later as standard, and as a guest OS, it can run Windows 7 or Linux as a guest OS, making it a prime candidate for PC replacement case.

2.2 Redirect Mechanism

To overcome the weakness of the expansion board not be handled from the guest OS, we devised a redirect mechanism that handles input/output requests as a mediator. Fig. 1 shows the redirect mechanism. The redirect driver is a simulated driver that replaces the original internal communication infrastructure driver and provides the same application interface as before.

A redirect service is a service that handles incoming and outgoing requests received by a redirect driver to the gateway service via Ethernet. A gateway service is a service that issues incoming and outgoing requests passed from a redirect service to the original communication system driver. This mechanism enables applications to use the legacy internal communication infrastructure with the same application interface as before, eliminating the need for modification.

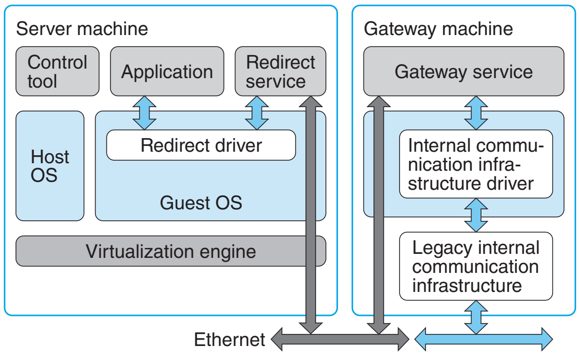

Fig. 1 Redirect Mechanism

A block diagram of key elements of the redirect mechanism is shown.

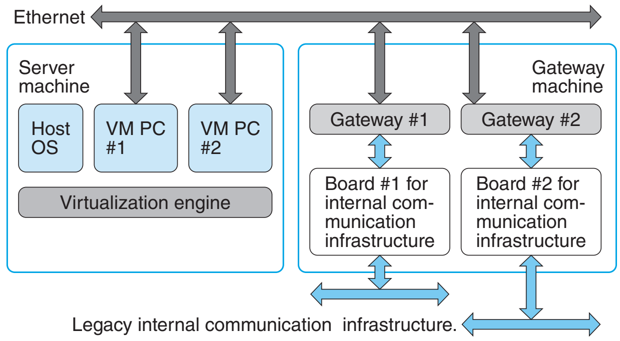

Fig. 2 Integration of Multiple PCs

An example is shown for the integration of multiple VM PCs replacing the existing PCs.

2.3 Alternative Configuration Variations

A typical configuration is a combination of a high-performance server machine without an expansion slot for running a guest OSes and an inexpensive gateway machine with an expansion slot. This replaces a single existing PC. As an extension of this configuration, we can consider an alternative configuration in which multiple guest OSes are installed on the server machine, multiple expansion boards are mounted on the gateway machine, and multiple existing PCs are integrated as one set of an alternative PC. Fig. 2 shows the integration of multiple PCs.

Conversely, we could think about a minimal configuration in which the host OS and guest OS are integrated as a single alternative PC with an expansion slot. Fig. 3 shows the minimum configuration of a substitutive PC. Since the OS of the host OS or the OS for the gateway machine may be different from the OS of the currently used PC to be replaced, it is possible to use Linux or the like for either or both.

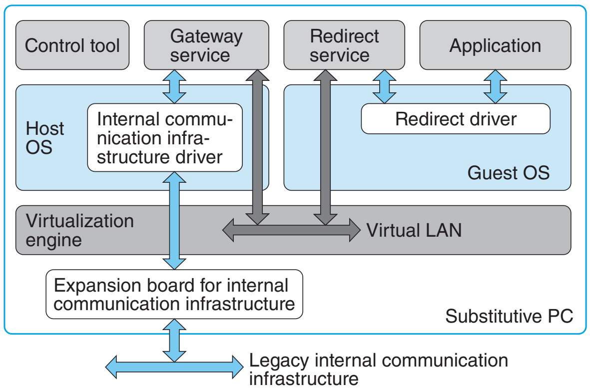

Fig. 3 Minimal Configuration of Substitutive PC

An example of a host OS integration is shown. The gateway service, communication transmission line system driver, and expansion board are integrated in the host OS.

2.4 Performance Measurement

We built a prototype redirect mechanism with the following configuration and measured its actual performance.

(1) Gateway machine (using existing PC): μPORT M5 model 200, Windows 7 Ultimate

(2) Server machine: PS7000, Windows 10 IoT

(3) Internal communication infrastructure: ADA7000

As a result of performance measurement, the following findings were obtained.

(1) Requests that do not involve communication have a large impact on the overhead. A simple set request delayed 0.5 ms to 1.0 ms (+50%).

(2) If the transmitted data is large, the overhead becomes inconspicuous. 1 byte transmission request was delayed from 2.0 ms to 2.7 ms (+30%). A request to transmit 4096 bytes was delayed from 5.7 ms to 6.3 ms (+10%).

2.5 Evaluation

We have confirmed that the overhead of the redirect mechanism remains at around 1 ms. Therefore, we conclude that our virtualization technology is useful as an alternative to the existing PCs that use the expansion board for our legacy internal communication infrastructure.

In fields where quick response is required, however, such as detection of input signal status changes using a GPIO expansion board, it is necessary to evaluate the effects of delay on a system-by-system basis.

3 Postscript

Since CPUs, chipsets, and OSes each have their own product lifetimes, replacement or upgrade to the latest generation model is unavoidable, but it is also a weak point in terms of system maintenance.

By using virtualization technology, however, we can expect to minimize the impact.

Going forward, we intend to enrich these variations of the redirect mechanism to market demands according.

・Microsoft and Windows are registered trademarks of Microsoft Corporation in U.S. and/or other countries.
・Ethernet is a registered trademark of FUJIFILM Business Innovation Corp.
・Linux is a registered trademark of Linus Torvalds in the U.S. and other countries.
・All product and company names mentioned in this paper are the trademarks and/or service marks of their respective owners.