

# Intel® Virtualization Technology (VT) in Converged Application Platforms

Enabling Improved Utilization, Change Management, and Cost  
Reduction through Hardware Assisted Virtualization

White Paper

---

*January 2007*

*Revision 1.0*



INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life saving, or life sustaining applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The Intel® Virtualization Technology may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Intel, the Intel logo, Itanium, and Xeon are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

\*Other names and brands may be claimed as the property of others.

Copyright © 2007, Intel Corporation. All rights reserved.



# Contents

---

1	Executive Summary .....	5
2	Converged Application Platform Overview.....	6
3	Intel® Virtualization Technology on Converged Application Platform Systems .....	8
3.1	Intel® Virtualization Technology Background .....	8
3.1.1	Virtualization Challenges on Software-only Virtualization.....	9
3.2	Intel® Virtualization Technology (Intel® VT) - Hardware Assisted Virtualization .....	10
3.2.1	Hardware Enhancement on VT-x.....	11
3.2.2	VMX Operations .....	11
3.3	Intel® Virtualization Technology: CAP Usage Models .....	11
3.3.1	Workload Isolation .....	12
3.3.2	Workload Consolidation.....	13
3.3.3	Workload Migration.....	14
3.3.4	Security .....	15
3.3.5	Limitation of Current Intel® Virtualization Technology (Intel® VT) Architecture.....	15
4	Conclusion .....	17
5	References.....	18

## Figures

Figure 1	Converged Application Platform .....	6
Figure 2	Virtualized vs Non Virtualized Platforms .....	8
Figure 3	T-x Ring Transition Block Diagram.....	10
Figure 4	Intel® Virtualization Technology Generic Usage Model .....	12
Figure 5	Example of OS Fail-over Implementation.....	13
Figure 6	Workload Consolidation on a Sever Supporting Intel® Virtualization Technology .....	14
Figure 7	Guest OS can be Migrated to a New Platform without Application Restart... ..	15

## Tables

Table 1.	Virtualized vs Non Virtualized Platforms .....	8
----------	--	---



## *Revision History*

---

Revision Number	Description	Revision Date
1.0	Initial release.	January 2007

§



# 1 *Executive Summary*

---

The advent of Voice over IP (VoIP) has bridged the thin gap separating data networks and voice networks. Over the years, the high speed last mile services have broadened their reach in tandem with wireless technology such as 3G and Wi-Fi\*, and end users are beginning to experience richer multimedia services in which video, voice, and data are integrated and delivered over the internet. As broadband penetration continues, the trend towards converged networks gains traction in the marketplace.

This document illustrates how Intel® Virtualization Technology (Intel® VT) can enhance Converged Application Platforms (CAP) by creating a secure, reliable, and consolidated environment to host voice, data, and video services, all in a single multi-function device.



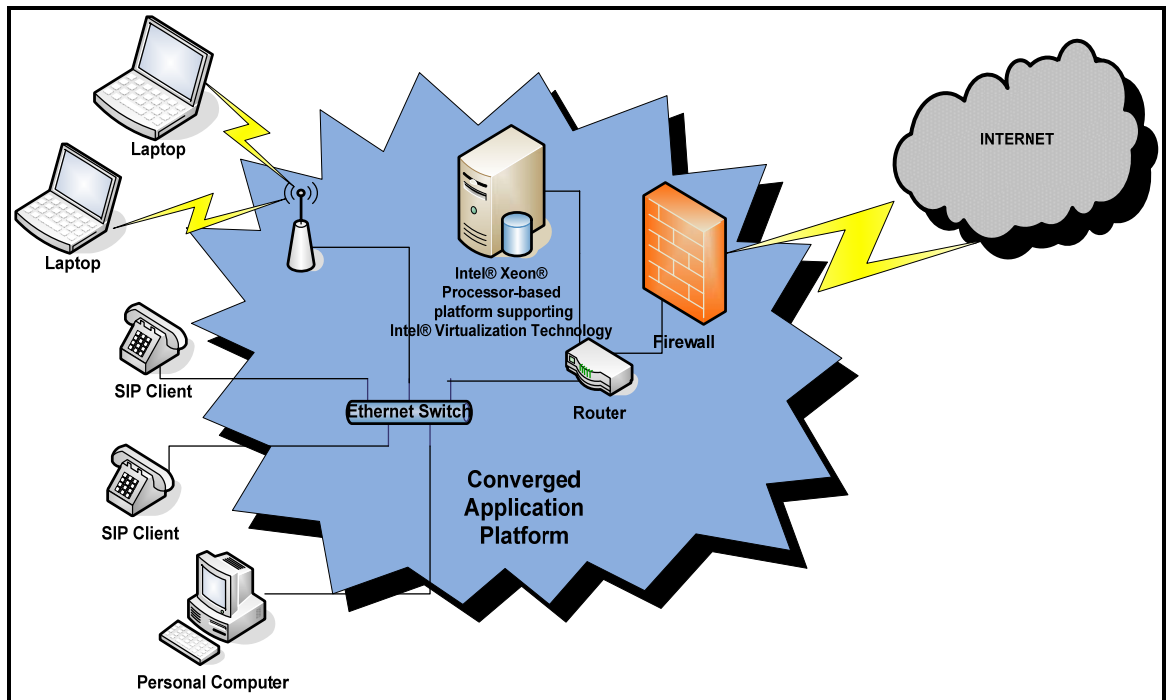
## 2 *Converged Application Platform Overview*

Converged Application Platform (CAP) is a common platform that supports data and multimedia services such as voice and video. The key benefits of CAP architecture include:

- Reduced burden on network administrator and IT staff
- Encrypted voice, data networking, and video streaming.
- Increased user control to manage preferences including call logs, security, and privacy
- Reduced space or footprint requirement for small and medium businesses
- Headroom to enable new services

In addition, CAP hosts many applications such as network routing, IP PBX server, web-hosting, streaming server, VPN, and firewall, which could require multiple servers. The market need for multiple functions in one system, coupled with the rising cost of operation, helped create the latest trend in server consolidation - virtualization technology.

Figure 1 Converged Application Platform





NOTE: \* VT-x enabled Intel® Processor for high performance server platform:

- Dual-Core Intel® Xeon® Processor LV and Intel® E7520 Chipset
- Dual-Core Intel® Xeon® Processor 5100 series and Intel® 5000P Chipset

The Intel® Xeon® Processor-based platforms supporting Intel Virtualization

Technology can be used to host different applications on its guest operating systems:

- Session Initiation Protocol Private Branch Exchange (SIP PBX) server (for example, Asterisk\*)
- Security applications such as firewall and antivirus applications.
- Server for internal applications such as Human Resource Management (HRM), email server and FTP server.
- Video/voice streaming server
- Web-hosting server



# 3 Intel® Virtualization Technology on Converged Application Platform Systems

## 3.1 Intel® Virtualization Technology Background

Virtualization creates a level of abstraction between physical hardware and the OS that manages the computer processor(s) and other platform hardware.

Figure 2 Virtualized vs Non Virtualized Platforms

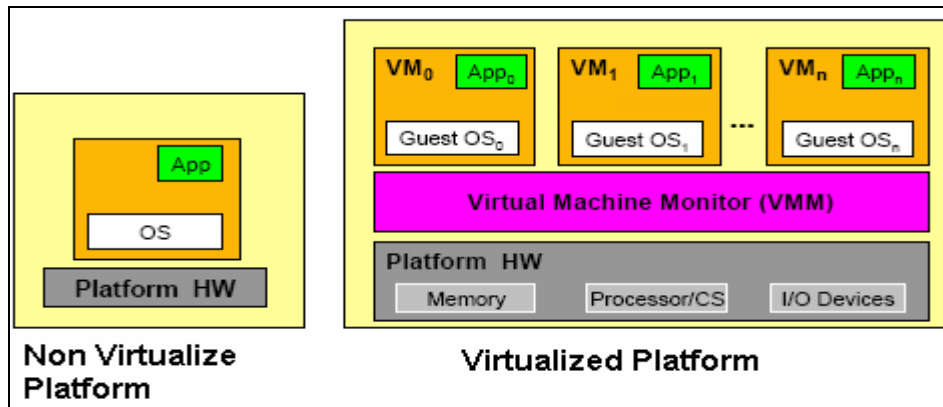


Table 1. Virtualized vs Non Virtualized Platforms

Non Virtualized Platform	Virtualized Platform
<ul style="list-style-type: none"> <li>IA-32 architecture requires all software to run in one of the four privilege levels called “rings”</li> <li>OS typically runs on Ring0 – privileged access to the widest range of processor and platform resources</li> <li>Individual applications run in Ring3 – Limited privilege access to hardware resources</li> </ul>	<ul style="list-style-type: none"> <li>The four privilege levels (rings) are still employed on Virtualization Technology based platforms, but now, instead of an OS, the VMM runs on Ring 0. Note that, typically, the OS is programmed to run on Ring0 in a non virtualized environment.</li> <li>VMM runs on Ring0 and guest OS runs on Ring1 or Ring3.</li> </ul>



Conceptually, without virtualization technology, a single operating system controls all hardware resources. The VMM presents each guest OS a virtual machine (VM) environment that emulates the hardware environment needed by the guest OS. Virtual-machine extensions define processor-level support for virtual machines on IA-32 processors. Two principal classes of software are supported under the virtual machine architecture:

- Virtual-machine monitor (VMM): A VMM acts as a host and has full control of the processor(s) and other platform hardware. VMM presents guest software with an abstraction of a virtual processor and allows it to execute directly on a logical processor. A VMM is able to retain selective control of processor resources, physical memory, interrupt management, and I/O.
- Guest software: Each virtual machine is a guest software environment that supports a stack consisting of the OS and application software. Each operates independently of other virtual machines and uses the same interface to processor(s), memory, storage, graphics, and I/O provided by a physical platform. The software stack acts as if it were running on a platform with no VMM. Software executing in a virtual machine must operate with reduced privilege so that the VMM can retain control of platform resources.

There are two options for software-only virtualization solution:

1. Runtime Modification of the guest OS: In this case the VMM monitors operation during runtime and takes control of the processor. When any of the 17 instructions controlling critical platform resources arises in the guest OS, the VMM manages the conflict and returns control to the guest OS.
2. Static modification on guest OS (Para-virtualization): In this case the guest OS is modified prior to runtime.

### 3.1.1 Virtualization Challenges on Software-only Virtualization

- When any of the 17 instructions controlling critical platform resource arises, but the OS is not running in Ring0, this could cause conflict resulting in system fault of wrong response.
- Runtime modification forces the VMM to provide complex workarounds during operations, which can impact performance and system reliability.
- Para-virtualization prevents VMM from hosting unmodified guest operating system
- Both runtime modification and para-virtualization require extensive software modification efforts from the VMM and the OS vendors. This increases the cost and complexity of IT support.

Today's virtualization solutions mainly involve virtual machines, which are implemented in software using techniques like ring compression and binary translation. This allows unmodified guest OS to run, at a slightly lower performance in the virtual machine. Para-virtualization requires changes to the guest operating system so it can surrender delicate system operations like page table memory and interrupt management to the VMM.



### 3.2 Intel® Virtualization Technology (Intel® VT) - Hardware Assisted Virtualization

Hardware support for processor virtualization enables system vendors to provide simple, robust, and reliable VMM software. VMM relies on hardware support to set policy and operational details for handling events, exceptions, and resources allocated to virtual machines. A hardware assisted processor must be able to avoid conflict caused by many guest operating systems running on top of the VMM software. This can be achieved if the processor can ensure that the VMM maintains control of critical platform resources and hands off limited control to each guest OS as appropriate. This efficiency and integrity of the hardware control switching between the VMM and guest OS are critical for optimal performance and reliability.

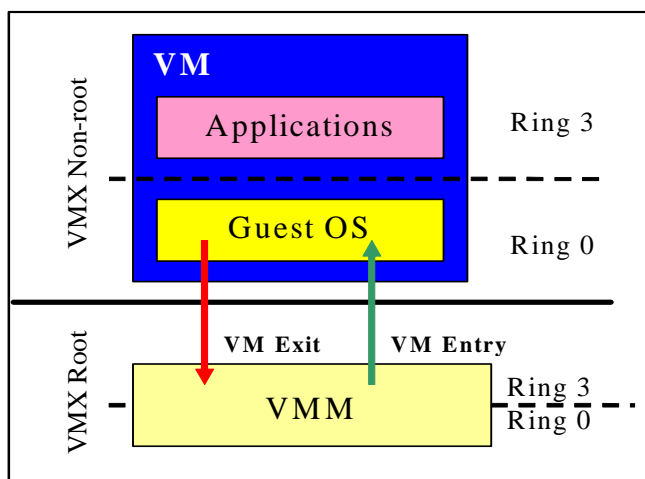
Intel VT provides support for IA-32 hardware assisted processor virtualization (VT-x) and directed IO virtualization (VT-d). VT-x consists of a set of virtual machine extensions (VMX) that support virtualization of processor hardware for multiple software environments using virtual machine. An equivalent virtualization technology to VT-x for Intel® Itanium® processor architecture is defined and commonly referred to as VT-i. The scope of this whitepaper will focus solely on VT-x and will not cover VT-d and VT-i implementation.

A VMM written to take advantage of the Intel Virtualization Technology runs the monitor in a new CPU mode called “VMX Root” mode and the guest OS in the “VMX Non-root” mode. The VMM will manage the virtual machines through the VM Exit and VM Entry mechanism.

Intel Virtualization Technology is designed to enable high performance VMM without the need for para-virtualization changes or binary translation techniques. This enables the implementation of VMM that can support a broad range of unmodified guest operating systems.

VT-x introduces IA-32 architecture with two new forms of CPU operations: VMX root and VMX non-root operation. The following figure illustrates the software model for the VT-x architecture.

Figure 3 T-x Ring Transition Block Diagram





### 3.2.1 Hardware Enhancement on VT-x

1. Higher Privilege Ring for the VMM: This allows guest OS and applications to run on a reprioritized ring they were designed for, while ensuring VMM has privilege control over platform resources. This helps to eliminate potential conflicts, simplify VMM complexity and improve compatibility with unmodified operating systems.
2. Hardware based Transitions: Handoffs between the VMM and guest OS are supported in hardware, which reduces the need for complex software transitions.
3. Hardware based Memory Protection: Processor state details are retrained for the VMM and each guest OS in dedicated address spaces. This helps to accelerate transitions and ensure reliability of the process.

### 3.2.2 VMX Operations

Processor support for virtualization is provided by a new form of processor operation called VMX operation. There are two kinds of VMX operation: VMX root operation and VMX non-root operation. In general, a VMM will run in VMX root operation and guest OS will run in VMX non-root operation. Transitions between VMX root operation and VMX non-root operation are called VMX transitions.

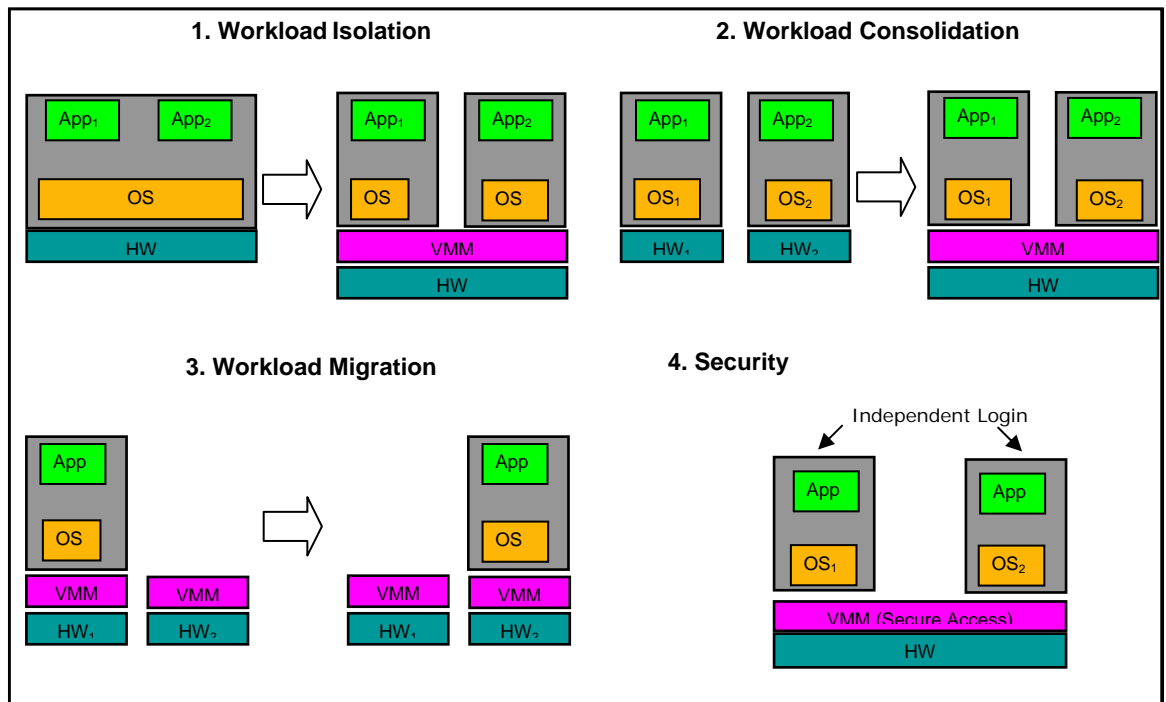
There are two kinds of VMX transitions. Transitions into VMX non-root operation are called VM entries. Transitions from VMX non-root operation to VMX root operation are called VM exits. Processor behavior in VMX root operation is very similar to its behavior outside VMX operation. The principal differences are that a set of new instructions (VMX instructions) is available and that limits values that can be loaded into certain control registers. Processor behavior in VMX non-root operation is restricted and modified to facilitate virtualization. Instead of their ordinary operation, certain instructions (such as the new VMCALL instruction) and events cause VM exits to the VMM. Because these VM exits replace ordinary behavior, the functionality of software in VMX non-root operation is limited. It is this limitation that allows the VMM to retain control of processor resources.

## 3.3 Intel® Virtualization Technology: CAP Usage Models

There are various Intel VT usage models which could be implemented to existing CAP systems to enhance the value proposition.



Figure 4 Intel® Virtualization Technology Generic Usage Model

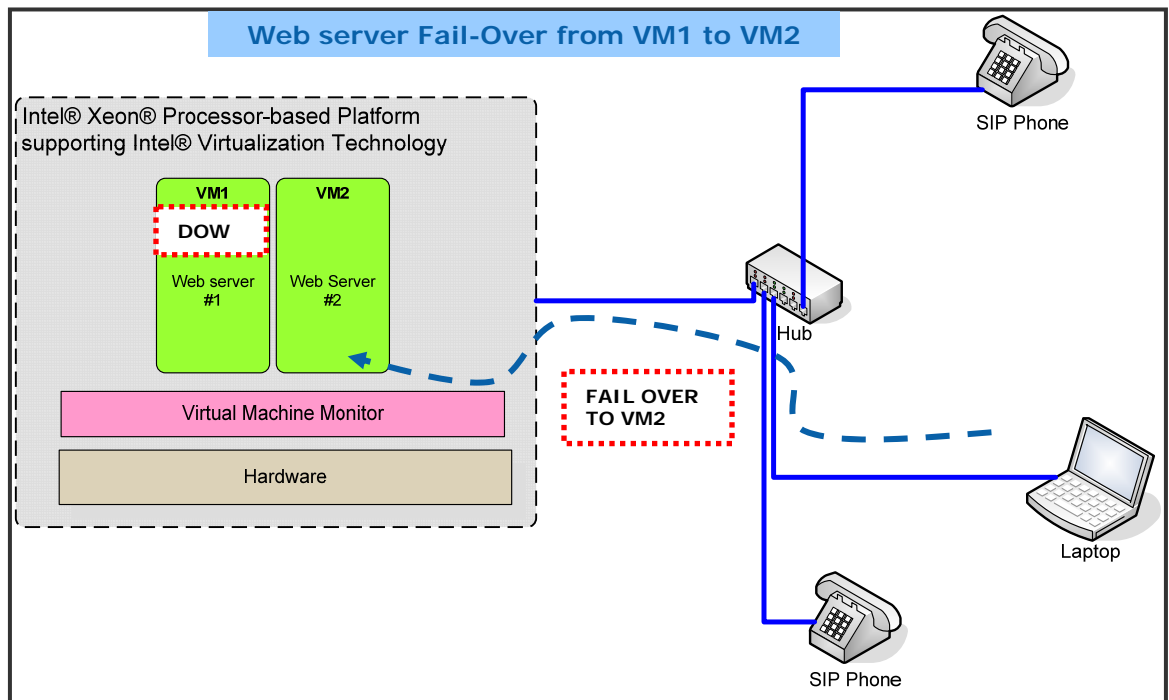


### 3.3.1 Workload Isolation

Workload isolation implies that each VM is independent of the other. What this means is that:

- Each guest OS on the VMM can be used to host different operating systems and applications depending on their criticality and functionality. For instance, security applications (such as firewall applications) and PBX SIP servers can be hosted on Linux\*-based guest OS, while Windows\* streaming servers can be hosted on Microsoft\* Windows-based operating systems simultaneously on a single platform.
- Guest OS could also be replicated to provide fail-over functionality. For critical application such as the PBX SIP servers, downtime could translate into a showstopper. If one VM installed with the server application goes down, it could be programmed to immediately switch to the second separate VM.

Figure 5 Example of OS Fail-over Implementation



### 3.3.2 Workload Consolidation

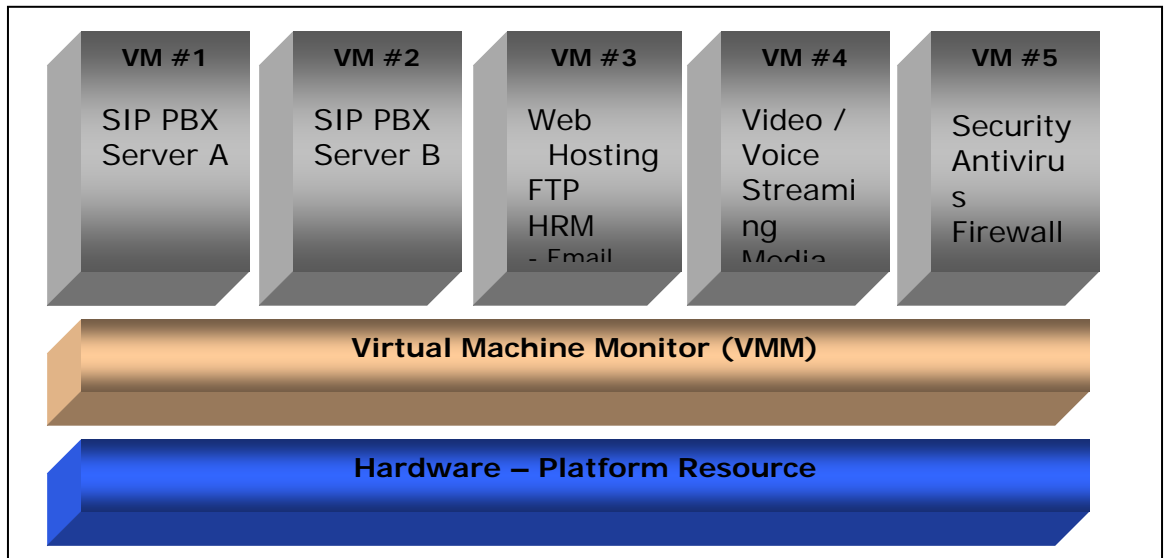
A few factors have driven the trend towards workload consolidation, primarily, the increasing cost of operations that comes with the higher number of servers required. This includes an increasing number people, space, power consumption and cooling solutions. Server consolidation is a key strategy for reducing these costs and today's virtualization software solutions make it easy to run multiple applications safely and securely on an Intel processor-based server. Companies are using virtualization software and associated management applications to:

- Consolidate workload by utilizing a physical server as several VMs, each capable of hosting its own OS and application stack.
- Manage and implement physical and VM resources efficiently from a common interface.
- Allocate server resources (CPU, memory and I/O) dynamically, and move running applications, workloads, and sessions very quickly from one VM to another. Initially this capability was used for zero-downtime maintenance. It is now beginning to be used as a method to automatically provision new capacity when a system fails or when the workloads threaten to exceed available resources.

Shown below are some of the applications that could be consolidated into Intel Xeon processor-based platform supporting Intel VT:



**Figure 6 Workload Consolidation on a Server Supporting Intel® Virtualization Technology**



- The SIP PBX Server is replicated to provide automatic fail-over to the other VMs. The replicated server VMM can also be hosted on separate platforms. Other methods of utilizing the fail-over model is to ensure that the client could detect the IP address of VM#1 and VM#2, and switch over to the healthy server if either server is down.
- Note that the other VMs can be configured to host a variety of applications such as SIP PBX Server, web hosting, FTP server, email server, firewall, antivirus, and video streaming hosted on various operating systems.
- Media server can be hosted on a VM intended to stream video or voice over IP.

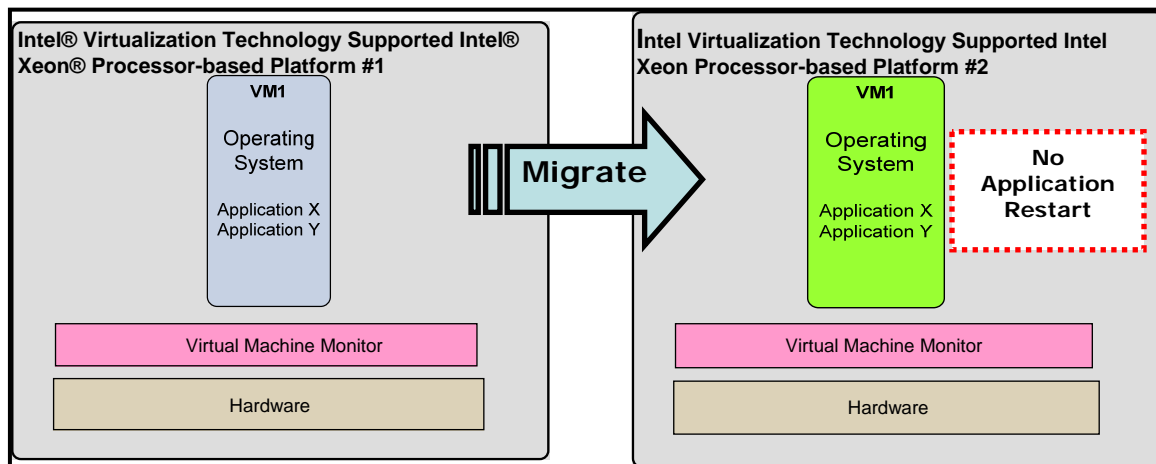
### 3.3.3 Workload Migration

Migrating VMs is akin to migrating an entire OS and all of its applications as one unit. This allows us to avoid many of the difficulties faced by process-level migration approaches. For instance, the narrow interface between a virtualized OS and the VMM makes it easy to avoid the problem of dependencies in which the original host machine must remain available and network-accessible in order to service certain system calls or even memory accesses on behalf of migrated processes. With virtual machine migration, on the other hand, the original platform that is hosting the VM may be shut down once migration has completed. This is particularly valuable when migration is occurring in order to allow maintenance of the original platform.

Migrating the virtual machine allows in-memory state to be transferred in a consistent and efficient fashion. This translates into a fast migration while the processes are still running on the operating system. This applies to kernel-internal state (for example, the TCP control block for a currently active connection) as well as application-level state, even when this is shared between multiple cooperating processes. In practical terms, for example, this means that it is possible to migrate an on-line advertisement via streaming media server without requiring clients to reconnect, using application level restart. This could also be implemented for the SIP PBX server migration, whereby connection established before the migration will still be alive even after migration to a new platform is completed.



Figure 7 Guest OS can be Migrated to a New Platform without Application Restart



### 3.3.4 Security

Intel Virtualization Technology manages VMX transitions in hardware rather than software. This helps to strengthen the logical isolation of virtual partitions. Less complex VMMs also provide fewer opportunities for software based attacks.

In addition, each guest OS can be password protected separately, although they reside on the same VMM. This would be especially useful to prevent unauthorized access, since the consolidated server is shared between different functional groups in an organization.

### 3.3.5 Limitation of Current Intel® Virtualization Technology (Intel® VT) Architecture

Despite the overwhelming benefits that come with virtualizing CAP platforms, there are also some areas of concern. VMMs consume valuable processing resources to manage operations. This is a permanent overhead that needs to be accepted by end users and system designers. In addition, system performance may take a hit from the context switching that occurs when the VMM switches between VMX root mode and VMX non-root mode. Lastly, I/O performance of present day VMM and VT platforms are limited by the architecture, causing greater latency and lower throughput I/O performance. Current VT technology (only VT-x is supported) will only be able to access the virtual I/O devices, which are mapped to physical I/O devices, instead of mapping directly to physical I/O devices.

Enhancements are already underway to fix these setbacks. VMM context switching and computing resource overhead is improving as VMM vendors discover how hardware assisted virtualization of Intel processors can enhance performance. I/O bottleneck is being addressed by chipsets that support I/O Virtualization (VT-d). Virtualization technology for directed I/O provides VMM with the following capabilities:

- Assign I/O devices across VMs: Flexibly assign I/O to VMs and extends protection & isolation properties of VMs for I/O accesses.
- Remap DMA: Direct Memory Access from devices can be directly address translated.
- Record and report DMA errors



These features allow VMs to provide better I/O performance through a new software interface, which has less overhead compared to emulation; and a direct assigned physical I/O device, which provides improved performance for I/O intensive applications. In addition, VMM can also support device assisted I/O sharing, which provides multiple functional interfaces, each of which may be independently assigned to a VM, allowing more virtual devices than physical devices in a platform.



## 4 *Conclusion*

---

The CAP architecture is paving the way to dramatically reducing capital and operational expenditure by consolidating multiple applications into a single system. It also simplifies management and improves efficiency by enabling a single network for voice and data.

Intel Xeon multi-core processor-based platforms supporting Intel VT provides processor level support for today's virtualization software solutions, making them more robust, secure, supportable, and interoperable when used to consolidate applications on the high performance platforms.

Intel Xeon processor-based platforms supporting Intel VT enables businesses to be at the forefront of innovation so that they can continue to drive down their total cost of ownership and create more flexible and manageable network infrastructure.



## 5 *References*

---

For more information on Intel Virtualization Technology enabled platforms, please visit the links below:

- Dual-Core Intel® Xeon® Processor LV 2.0 GHz for Dual-Processor Embedded Computing and Communications Applications  
<http://developer.intel.com/design/intarch/dualcorexeon/overview.htm>
- Intel® E7520 Chipset for Dual-Core Intel® Xeon® Processor LV 2.0 GHz for Embedded Computing  
[http://developer.intel.com/design/chipsets/embedded/e7520\\_dcxeon\\_docs.htm](http://developer.intel.com/design/chipsets/embedded/e7520_dcxeon_docs.htm)
- Dual-Core Intel® Xeon® Processor 5100 Series for Dual-Processor Embedded Computing  
<http://developer.intel.com/design/intarch/dualcorexeon/5100/index.htm>
- Intel® 5000P Chipset for Dual-Core Intel® Xeon® Processor 5100 Series  
<http://developer.intel.com/design/chipsets/embedded/5000P.htm>
- Intel® Virtualization Technology Documentation  
<http://www.intel.com/technology/virtualization/index.htm>