CLAYTON FERREIRA SANTOS

AMBIENTE DE VIRTUALIZAÇÃO: UMA ANÁLISE DE DESEMPENHO

LAVRAS - MG
2011

CLAYTON FERREIRA SANTOS

AMBIENTE DE VIRTUALIZAÇÃO:
UMA ANÁLISE DE DESEMPENHO

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Sistemas de Informação para obtenção do título de Bacharel em Sistemas de Informação.

Área de Concentração:

Virtualização de Servidores

Orientador:

Luiz Henrique Andrade Correia

LAVRAS
MINAS GERAIS - BRASIL
2011

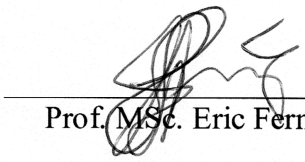
CLAYTON FERREIRA SANTOS

AMBIENTE DE VIRTUALIZAÇÃO: UMA ANÁLISE DE DESEMPENHO

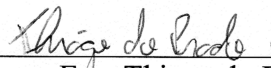
Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Sistemas de Informação para obtenção do título de Bacharel em Sistemas de Informação.

Aprovada em 06 de junho de 2011.

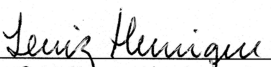
Prof. DSc. Raphael Winckler de Bettio



Prof. MSc. Eric Fernandes de Mello Araújo



Esp. Thiago do Prado Ramos



Prof. DSc. Luiz Henrique Andrade Correia
(Orientador)

LAVRAS
MINAS GERAIS – BRASIL

AGRADECIMENTOS

À minha esposa, Maria da Conceição Andrade Santos, pelo companheirismo, carinho e apoio em cada passo desse projeto.

À minha família, que mesmo distante, apoiava e torcia pela conclusão desse projeto.

Ao professor DSc. Luiz Henrique Andrade Correia pela orientação, paciência, dedicação, ensinamentos transmitidos, atenção, apoio e amizade.

Aos amigos da Diretoria de Gestão de Tecnologia da Informação que a todo o momento torceram e apoiaram cada decisão e atitude tomada para a conclusão desse trabalho.

Aos amigos do Departamento de Ciência da Computação pela torcida dedicada, em particular Eder e Joaquim, pela ajuda na conclusão dos experimentos.

Aos colegas do curso de Bacharelado em Sistemas de Informação, onde juntos sofremos e nos apoiamos até o final desse projeto.

RESUMO

O presente trabalho realiza um estudo de desempenho de ferramentas de virtualização, com o intuito de dar suporte para a escolha da ferramenta mais adequada para a criação de ambientes virtuais em servidores de redes. As ferramentas objeto de estudo foram o *VMware ESXi* e o *Citrix XenServer*. Dessa forma, foram utilizadas ferramentas de *benchmark* para análise do comportamento das máquinas virtualizadas, gerando uma sobrecarga na demanda dos mesmos recursos básicos, tais como, processador, memória e disco. Os testes foram distribuídos em categorias baseadas nos recursos analisados, sendo aplicados ensaios em dois sistemas operacionais hóspedes, *Windows 2003 Server* e *Fedora* 14. Para cada ferramenta de virtualização, foram criadas cinco máquinas virtuais, onde foram realizados testes de desempenho, primeiramente, com uma máquina virtual, depois com duas máquinas, a seguir com três, quatro e finalmente cinco máquinas executando simultaneamente. Com os testes concluídos, pode-se observar particularidades de cada ferramenta de virtualização e realizar uma avaliação sobre seu desempenho, suas limitações e comportamento. Observou-se que a escolha do sistema operacional hóspede influencia no desempenho da ferramenta de virtualização escolhida. Assim, para máquinas virtuais com sistema operacional hóspede *Windows 2003 Server*, a ferramenta que apresenta melhor desempenho comparativo é o *Citrix XenServer*. Para máquinas virtuais com sistema operacional hóspede *Fedora* 14 a escolha mais adequada seria o *VMware ESXi*.

Palavras-chave: Virtualização. Servidores. Desempenho. VMware. XenServer

ABSTRACT

The present work conducts a performance study of virtualization tools with the aim of giving support to choose the most appropriate tool for creating virtual environments on network servers. The tools were the subject of study VMware ESXi and Citrix XenServer. Thus, benchmark tools were used to analyze the behavior of virtualized machines, creating an overload on the demand of the same basic resources such as processor, memory and disk. The tests were divided into categories based on the resources analyzed, and applied testing two guest operating systems, Windows Server 2003 and Fedora 14. For each virtualization tool, was created five virtual machines, which were conducted performance tests, first with a virtual machine, then with two machines, then three, four and finally five machines running simultaneously. With the tests completed, was observed characteristics of each virtualization tool and make an evaluation on their performance, their limitations and behavior. Where it was observed that the choice of guest operating system influences the performance of virtualization tool chosen. Thus, for virtual machines guest OS Windows 2003 Server, the tool that performs best is the comparative Citrix XenServer. For virtual machines guest OS Fedora 14 would be the best choice VMware ESXi.

Keywords: Virtualization. Servers. Performance. VMware. XenServer.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Atualmente a capacidade de processamento dos computadores tem aumentado significativamente devido ao desenvolvimento de novos *hardwares* e às novas técnicas de programação (BARUCHI, 2008). Entretanto, toda a capacidade adquirida não tem sido plenamente aproveitada. Dessa forma, em boa parte do tempo, a capacidade computacional não se encontra em um nível de utilização desejado. Para tentar minimizar esse problema, muitos administradores de sistemas vêm recorrendo às técnicas de virtualização. A virtualização é um termo geral usado para descrever várias tecnologias que dividem os recursos de *hardware* em múltiplos ambientes, aplicando um ou mais conceitos ou tecnologias (AGOSTINHO, 2009). A virtualização foi utilizada nos *mainframes* e, atualmente, está disponível para todos os aspectos da computação.

De acordo com Pollon (2008), com a crescente onda ecológica e a tendência mundial de diminuir o lixo eletrônico, existe um incentivo à utilização da virtualização para a criação dos centros de processamento de dados “verdes”, onde conceitos como redução do consumo de energia, melhor uso de espaço físico e recursos computacionais, entre outros, são fatores decisivos nas organizações atuais.

A virtualização possibilita um melhor uso dos recursos, eliminando a necessidade de grandes ambientes climatizados e cheios de máquinas, que com o passar do tempo, seriam sucateadas. Além disso, é possível uma redução do consumo de energia e de manutenção (POLLON, 2008).

O problema da manutenção demandada pelo ambiente de servidores virtualizados, ainda não apresenta consenso na comunidade de tecnologia da informação (TI). Alguns pesquisadores defendem que num ambiente virtualizado existem todos os problemas do mundo real mais os problemas gerados pela virtualização. Todavia, outros pesquisadores argumentam que os benefícios oriundos da virtualização sobrepõem aos problemas que possam vir a surgir (POLLON, 2008).

A virtualização é uma solução viável quando empregada como consolidadora de servidores e/ou serviços para muitas empresas, pois permite aumentar a utilização dos servidores, além de permitir diversos sistemas operacionais em uma mesma plataforma física (JUNIOR, 2008). Isso possibilita benefícios na manutenção, disponibilidade, confiabilidade e mobilidade dos sistemas, tolerância a falhas, além de rápida recuperação de problemas (POLLON, 2008). Visando reduzir custos e aumentar sua capacidade de TI

com qualidade, as empresas realizam estudos para a escolha da melhor ferramenta e da melhor técnica de virtualização a ser empregada.

O objetivo de aplicação das técnicas de virtualização é, segundo Baruchi (2008), aumentar a quantidade de serviços de um ambiente de TI com o mínimo de manutenções físicas. Com esse intuito, no mercado, existem diversos *softwares* que fazem gerenciamento de Máquinas Virtuais (VM), tais como *VMware* (VMWARE, 2010), *Xen* (XEN, 2010), *Microsoft Virtual Server* (MICROSOFT, 2010), *VirtualBox* (ORACLE, 2010), *FreeBSD Jails* (FREEBSD, 2010) e outros. Os *softwares* possuem vantagens e desvantagens quando comparados entre si, como custo, desempenho, confiabilidade, estabilidade, entre outros fatores. A escolha da ferramenta adequada é essencial para o sucesso do processo de mudança.

A utilização de ferramentas de virtualização para criação de ambientes virtuais pode trazer uma série de benefícios para as organizações tais como: facilitar os testes de otimização, configuração, novas instalações de servidores e aplicações, pois podem ser feitos testes em máquinas virtuais simulando o ambiente de trabalho; reduzir o tempo e custo de recuperação de desastres, pois todo o ambiente virtualizado pode ser replicado para outro *hardware* sem que seja necessária a instalação e configuração do servidor ou computador em questão (AGOSTINHO, 2009).

As Máquinas Virtuais (VMs) podem executar em ambientes isolados uma das outras, podem ser criados servidores virtuais para cada tipo de aplicação. Assim, se um servidor virtual apresentar algum problema de segurança, os demais servidores não serão afetados. Dessa forma, a virtualização de servidores, a criação de ambientes virtuais de teste, a configuração e a consolidação de servidores e serviços está cada vez mais popular como “boa prática” de segurança e continuidade de negócio, devido aos benefícios proporcionados por essa técnica. Esses benefícios podem ser aplicados em instituições que tenham por objetivo aumentar sua capacidade de serviços ofertados, além de proporcionar um maior potencial de flexibilidade no gerenciamento, manutenção e controle de serviços e servidores.

Assim, os objetivos e justificativas deste trabalho são apresentados nas seções seguintes.

1.1 Objetivos

O objetivo geral deste trabalho é a criação, instalação, configuração e migração de servidores para ambientes virtualizados, de forma a possibilitar uma análise adequada ao seu emprego, desempenho e uso.

Os objetivos específicos a serem alcançados com a utilização da virtualização dos servidores são:

- criação, instalação, configuração e migração de servidores um ambiente virtualizado;
- melhor aproveitamento na utilização dos recursos de TI disponíveis;
- possibilidade de centralização e simplificação do processo de gerência e manutenção dos recursos de rede;
- economia de espaço físico disponível;
- oportunidade de criação de ambientes seguros para servirem de alvo para ataques externos à organização;
- facilidade de recuperação de incidentes;
- melhoria na qualidade do atendimento e do conhecimento da equipe técnica;
- alinhamento com as boas práticas internacionais de qualidade e segurança da informação;
- Análise comparativa das soluções de virtualização, indicando qual o uso adequado para cada ferramenta.

Assim, esses objetivos para o desenvolvimento deste trabalho estão justificados na seção seguinte.

1.2 Justificativa

A maioria das organizações possui um número limitado de recursos computacionais, de pessoal técnico e espaço físico para atender às crescentes demandas oriundas de seu público-cliente. As organizações necessitam da centralização da gerência de seus recursos de TI e da garantia da disponibilidade, de segurança das informações e de

serviços oferecidos. Dessa forma, são sempre desejáveis soluções que possibilitem um ganho ou uma melhora dos ambientes e serviços atualmente oferecidos, pela aquisição de ferramentas que ajudem o gerenciamento e manutenção de seus sistemas de TI.

A aplicação de ferramentas de virtualização pode trazer benefícios de grande valia para a equipe responsável pela gerência de recursos de TI, ao possibilitar um ambiente de testes e configuração seguro, melhorando a qualidade do serviço e a experiência dos técnicos envolvidos. Essas soluções são comumente associadas a ações de “boas práticas” em continuidade de negócios e segurança da informação.

Com intuito de aumentar a segurança e a continuidade de negócio da organização, é desejável que esta se adeque às normas internacionais de segurança da informação. A segurança da informação compreende a proteção das informações, sistemas, recursos e demais ativos contra desastres, erros (intencionais ou não) e manipulação não autorizada, objetivando redução da probabilidade de impacto. A norma ABNT NBR ISO/IEC 27002 (2007) afirma que a segurança da informação é a proteção da informação de vários tipos de ameaças para garantir a continuidade de negócio, que deve ser obtida como resultado da implementação de um conjunto de controles, compreendendo estruturas organizacionais e funções de *hardware* e *software*.

Atualmente, por causa da ausência da criação de ambientes virtualizados de servidores, muitas das soluções implantadas são realizadas em máquinas de produção, gerando muitos transtornos e desgastes entre os técnicos responsáveis e a sua população cliente. A busca por soluções que mitiguem a ocorrência desses problemas e aumentem a qualidade e a segurança da informação, devem ser consideradas como objetivos diários da equipe de gerência e manutenção de rede.

1.3 Motivação

A virtualização permite a execução de múltiplas instâncias de máquinas virtuais sobre um mesmo equipamento, o que permite a instalação de servidores de arquivos, e-mail, páginas web, entre outros, dentro do ambiente da organização. Dessa forma, essa tecnologia também cria uma interessante padronização de *hardware*, abstraindo as complexidades do sistema operacional e das camadas de aplicações (CRUZ, 2008).

Segundo Strianese (2010), há um alto potencial de crescimento para o mercado de virtualização para consolidação de servidores, estimando que 42% das médias e grandes

empresas no Brasil já adotaram algum grau de virtualização e que esse mercado irá movimentar algo em torno de US$ 11,7 bilhões até 2011.

A virtualização permite o encapsulamento de ambientes e criação de pontos de restauração e recuperação, muito útil para criação de ambientes de testes e configuração. Essa capacidade oferecida por esse tipo de ferramenta permite uma rápida recuperação de estados indesejáveis do ambiente durante sua configuração e ainda um aumento na segurança deste mesmo ambiente, pois qualquer incidente ocorrido fica restrito àquela máquina virtual, não afetando o funcionamento das demais.

As organizações em geral, tentam manter a disponibilidade de seus serviços a níveis aceitáveis pela sua população cliente. Porém não são raros os casos em que é necessário realizar interrupção de seus serviços, seja programada por sua equipe técnica, ou por ocasião de algum incidente. No caso das paradas programadas por sua equipe técnica, a maioria poderia ser evitada, caso estivesse em conformidade com as normas técnicas de segurança da informação e continuidade de negócio, as NBR ISO 27000.

Com o uso da virtualização, a criação de ambientes seguros de testes de *software* e de sistemas aumentaria a segurança, a disponibilidade e o tempo de recuperação e/ou restauração dos serviços, sem comprometer o ambiente de produção. Além disso, possibilita alinhamento da organização com as boas práticas de segurança e continuidade de negócio.

1.4 Organização do Trabalho

Este trabalho está estruturado da seguinte forma: no Capítulo 2 são fundamentados os principais conceitos sobre virtualização e seus tipos e métodos de aplicação. No Capítulo 3 são descritas as principais ferramentas de virtualização e suas características. No Capítulo 4 é apresentada a metodologia aplicada, bem como, as ferramentas selecionadas. No Capítulo 5 são apresentados os resultados obtidos. Finalmente, no Capítulo 6, as considerações finais do trabalho, sua conclusão e os trabalhos futuros.

2 REFERENCIAL TEÓRICO

Este capítulo apresenta uma visão geral da tecnologia de virtualização descrevendo seu conceito, as três principais técnicas de virtualização e a terminologia utilizada nesta área do conhecimento.

2.1 Virtualização

A virtualização de servidores, tão conhecida e difundida atualmente nos servidores da plataforma x86, tem a sua origem e seus conceitos diretamente relacionados a descobertas e pesquisas da IBM (*International Business Machines*). A tecnologia como apresenta-se hoje vem sendo preparada desde os anos 90, mas somente agora ganhou grande projeção no meio tecnológico. A *VMware,* empresa responsável pelo desenvolvimento do primeiro *software* de gerenciamento de *hardware* (*hypervisor*) para arquitetura x86 na década de 90, é a responsável pelo "boom" da virtualização de servidores, afirma Favacho *et al.*(2008).

Muitas vezes a virtualização é confundida com emulação e simulação, porém são conceitos diferentes. De maneira simples, simulação é fazer algo se parecer ou funcionar como outra coisa. Na emulação um *software* é responsável por simular um computador real traduzindo todas as instruções. No caso da virtualização ocorre a multiplexação do *hardware* real, possibilitando assim a criação de diversas máquinas virtuais que podem ser consideradas uma cópia idêntica e isolada do *hardware* real (JUNIOR, 2008).

O conceito de virtualização não é novo, segundo os autores Williams e Garcia (2007), essa tecnologia baseia-se numa técnica que adiciona entre as aplicações e o *hardware* uma camada de abstração, podendo melhorar os níveis de serviços e qualidade dos mesmos. Em sua essência, segundo Carissimi (2009), a virtualização consiste em estender ou substituir um recurso ou uma interface existente por um outro de modo a imitar um comportamento. Isso é feito através de uma camada de *software* responsável por transformar ações de um sistema *A* em ações equivalentes em um sistema *B* (isomorfismo). Dependendo de como e onde essa transformação é feita, é possível classificar os *softwares* de virtualização em três grandes categorias (ROSEMBLUM, 2004):

a) *Nível de hardware:* é aquela em que a camada de virtualização é posta diretamente sobre a máquina física e a apresenta às camadas superiores como um *hardware* abstrato similar ao original. Corresponde à definição original de máquina virtual dos anos 60, onde máquina virtual é um conceito de sistemas operacionais para indicar uma abstração em *software* de um sistema computacional em *hardware* (CARISSIMI, 2009). Oferece compatibilidade de *software*. Isso significa que todo *software* desenvolvido para uma máquina virtual deve executar nela independente de onde ela esteja sendo usada.

b) *Nível de sistema operacional*: é um mecanismo que permite a criação de partições lógicas em uma plataforma de maneira que cada partição seja vista como uma máquina isolada, mas que compartilham o mesmo sistema operacional. Nesse caso, a camada de virtualização se insere entre o sistema operacional e as aplicações. Dessa forma, um *software* em execução em uma máquina virtual não deve ver, afetar ou ser afetado por outro *software* em execução em outra máquina virtual.

c) *Nível de linguagens de programação*: a camada de virtualização é um programa de aplicação do sistema operacional. O objetivo é definir uma máquina abstrata sobre a qual executa uma aplicação desenvolvida em uma linguagem de programação de alto nível específica. Assim ocorre o encapsulamento, permitindo a qualquer momento a captura do estado completo do ambiente virtual parando a sua execução. Isso deve ser feito de forma a possibilitar que, posteriormente, a execução seja retomada a partir desse estado.

Segundo Agostinho (2009), em virtude do crescimento dessa técnica, assim como em qualquer outra área, surgem as várias características resultantes que são apresentadas a seguir.

- A virtualização reduz o tempo de indisponibilidade dos servidores e elimina a necessidade de compra de equipamentos, restringindo somente para o caso de ser necessária a reposição.
- A maioria dos produtos de criação de cópias de segurança/restauração suportam a recuperação dos Sistemas Operacionais (SOs) e aplicações dos servidores físicos instalados como VMs.

- O uso de *Virtual Desktops* pode reduzir custos, enquanto permite manter o controle do ambiente cliente e fornece camadas adicionais de segurança sem custos adicionais.
- As máquinas virtuais (VMs) são ideais para criação e distribuição de ambientes de teste para viabilidades de soluções (*Proof of Concept* – POC) de forma segura e confiável. Caso esse ambiente criado obtiver sucesso, sua transferência para um ambiente produtivo torna-se fácil e relativamente rápida, sem ter que reconstruir o ambiente.
- A facilidade de gerenciamento oferecido pelas soluções de virtualização diminui o número de máquinas físicas, facilitando o trabalho dos técnicos responsáveis.
- A utilização eficiente dos recursos de processamento, memória e espaço de armazenamento e físico. Desta forma, pode-se criar vários sistemas heterogêneos ou não, isolados em um único *hardware*, obtendo-se uma melhor utilização de seus recursos.
- Possibilitar a criação de vários ambientes em um único *hardware*, permitindo que tarefas como consolidação de aplicações, consolidação de servidores e migrações entre ambientes sejam executadas sem riscos, e, em sua grande maioria, eliminando a necessidade de aquisição de um novo *hardware*.

Essas características são decorrentes das propriedades básicas da virtualização, que segundo Pollon (2008) são:

- ✓ *Particionamento*: é a capacidade de partilhar o *hardware* físico, geralmente executado pelo *hypervisor*.
- ✓ *Isolamento:* representa a separação entre máquinas virtuais em execução e uma máquina física. Um processo de máquina virtual não pode interferir em outra máquina virtual e também não pode interferir no monitor de máquina virtual (VMM).
- ✓ *Encapsulamento*: na virtualização, uma máquina virtual é implementada na forma de arquivo ou conjunto de arquivos. Esse arquivo, ou conjunto de arquivos, contém o *hardware* virtual, o sistema operacional e as aplicações instaladas. Desta forma, a máquina virtual pode ser movida entre máquinas físicas e transportada em qualquer dispositivo de armazenamento (CD, DVD, discos removíveis entre outros). Essa propriedade em particular é uma das

responsáveis pela implementação de soluções de recuperação de desastres e continuidade de negócios.

- ✓ *Desempenho*: por inserir uma camada extra de *software*, pode haver um comprometimento no desempenho de um Sistema Operacional (SO), porém este é compensado pelos benefícios adquiridos com o uso da virtualização.
- ✓ *Gerenciabilidade*: capacidade de gerenciar uma máquina virtual independente das outras máquinas virtuais.
- ✓ *Compatibilidade de software*: todo *software* escrito para executar em uma determinada plataforma deve ser capaz de rodar em um máquina virtual que virtualiza essa plataforma.
- ✓ *Eficiência*: instruções que não comprometam o hospedeiro podem ser executadas diretamente no *hardware*.
- ✓ *Inspecionabilidade*: o VMM deve ter acesso a todas as informações sobre os processos que estão rodando em suas máquinas virtuais, bem como o controle sobre os mesmos.
- ✓ *Interposição*: o VMM tem de ser capaz de inserir instruções de operação de máquinas virtuais.

A NBR ISO 27002 (ABNT, 2005) possui uma seção referente ao gerenciamento das operações e comunicações (seção 10), na qual indica a organização que convém criar controles de segurança para proteção de códigos maliciosos e móveis e cópias de segurança para garantir a integridade da informação. Em relação à proteção de códigos maliciosos, com o uso da virtualização é comum a criação de máquinas servidoras com o intuito de serem alvos de ataques externos a instituição, denominadas de *honeypots* ou *honeynets*, permitindo um estudo e avaliação da segurança do ambiente.

Essa mesma norma possui outra seção específica referente à gestão da continuidade de negócio (seção 14). Ela ressalta a importância de criar controles que não permitam a interrupção das atividades do negócio e que protejam os processos críticos contra efeitos de falhas ou desastres significativos, de forma a assegurar a sua retomada em tempo hábil. A virtualização permite isso através da criação de cópias das máquinas virtuais criadas, além da criação de pontos de restauração. Em alguns casos, dependendo da solução de ferramenta adotada, é possível que o próprio monitor de máquina virtual, ao perceber que uma máquina está indisponível (*off-line*), restabeleça essa máquina em outro ambiente virtualizado controlado por outro monitor de máquina virtual automaticamente.

A seguir são apresentados os principais conceitos de virtualização.

2.1.1 Máquina Virtual

Máquina Virtual (*Virtual Machine* – VM) refere-se à instância de um *hardware* virtualizado e um sistema operacional também virtualizado. Normalmente estão sob a forma de simulação, ou seja, uma interface com o ambiente, diferentemente da emulação que refletiria todos os estados internos do ambiente ao mesmo tempo. Uma máquina virtual pode executar qualquer tipo de *software* como um servidor, um cliente ou um *desktop*. Esta máquina virtual também é chamada de computador virtual, convidado, `domain U`, `domU` ou domínio sem privilégios. Pode-se dizer que máquina virtual é uma duplicata eficiente e isolada de uma máquina real (LAUREANO, 2006).

Uma máquina virtual pode ser definida como a abstração em *software* de uma máquina física real. Esta abstração possibilita a divisão de uma única plataforma física de *hardware* em duas ou mais plataformas virtuais, tendo cada plataforma virtual os seus próprios recursos e dando aos usuários a ilusão de estarem acessando diretamente a máquina física (JUNIOR, 2008).

Entre os vários conceitos envolvidos no estudo de máquinas virtuais, o monitor de máquina virtual é um dos principais, sendo abordado na seção a seguir.

2.1.2 Monitor de Máquina Virtual

O monitor de máquina virtual (também chamado de *hypervisor*, VMM, ou *Virtual Machine Monitor*) é uma camada de *software* introduzida entre o sistema visitante (*guest system*) e o *hardware* onde o sistema visitante executa. Essa camada faz uma interface entre os possíveis sistemas visitantes (virtuais) e o *hardware* que é compartilhado por eles. Ele é responsável por gerenciar todas as estruturas de *hardware*, como MMU (*Memory Management Unit*), dispositivos de E/S, controladores DMA (*Direct Memory Access*), criando um ambiente completo (máquina virtual), onde os sistemas visitantes executam. O VMM é o centro da virtualização de servidores, que gera recursos arbitrários de *hardware* e os múltiplos pedidos dos hóspedes dos sistemas operacionais e das aplicações (WILLIAMS & GARCIA, 2007).

O *hypervisor* ou VMM é uma plataforma de virtualização que possibilita a execução concomitante de vários sistemas operacionais em um único *hardware* (FAVACHO *et al.*, 2008). Para realizar tudo isto, o VMM pode atuar de duas maneiras distintas para promover a virtualização:

a) *Tipo 1*: é aquele que é executado diretamente no *hardware* da máquina, como se fosse um sistema operacional, também chamado de *bare metal*, como pode ser visto na Figura 2.1 (ANDRADE, 2006). Os sistemas operacionais virtualizados são executados em um segundo nível acima do *hardware*, logo acima do *hypervisor*. Alguns exemplos ferramentas de virtualização são: *VMware* ESX, Xen, dentre outros.

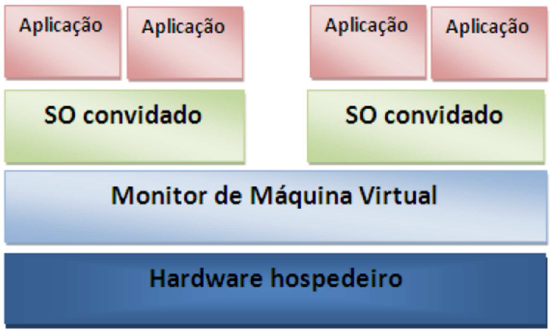


Figura 2.1 - Arquitetura de *hypervisor* que executa diretamente no *hardware* (tipo 1).
Fonte: Andrade (2006)

b) *Tipo 2*: é aquele que é executado sobre um sistema operacional existente em uma segunda camada, como mostrado na Figura 2.2 (ANDRADE, 2006). Já os sistemas operacionais virtualizados rodam em um terceiro nível acima do *hardware*, logo acima do *hypervisor*. Alguns exemplos são: *VMware Workstation*, *VMware Player*, *Microsoft Virtual Server*, Xen, dentre outros.

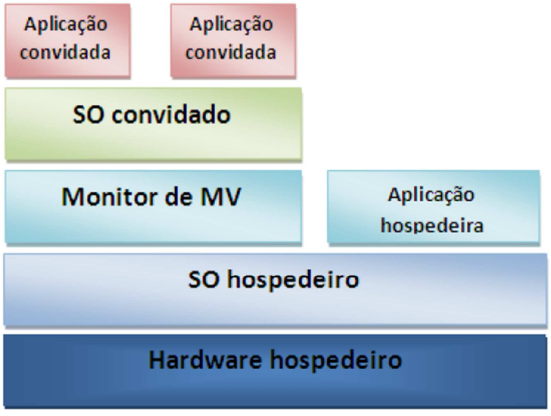


Figura 2.2 - Arquitetura de *hypervisor* que executa sobre um sistema operacional (tipo 2).
Fonte: Andrade (2006)

Segundo Junior (2008), objetivando o melhor desempenho das aplicações executadas nos sistemas hóspedes, várias otimizações são aplicadas aos VMMs de tipo 1 e

do tipo 2. Estas otimizações dizem respeito, principalmente, às operações de entrada e saída, devido à sua importância em um ambiente virtualizado. As otimizações mais utilizadas são divididas em dois grupos: otimizações em VMMs do tipo 1 e otimizações em VMMs do tipo 2.

Na abordagem híbrida do tipo 1, o sistema hóspede acessa diretamente o *hardware* hospedeiro, sendo que para que isso seja possível, o VMM é modificado para ter acesso às *Application Programming Interfaces* (API) do sistema hospedeiro. Na abordagem híbrida do tipo 2, o sistema hóspede pode ter acesso direto ao sistema hospedeiro com o VMM oferecendo ao sistema hóspede partes da API do sistema hospedeiro; ou pode ter o *hardware* acessado diretamente pelo sistema hóspede*;* ou, finalmente, o VMM tem acesso direto ao *hardware*.

De um modo geral vale destacar que, para criar partições virtuais em um servidor, uma fina camada de *software* chamada de *Virtual Machine Monitor* (VMM), é executado diretamente sobre a plataforma do *hardware* físico (WILLIAMS & GARCIA, 2007). Dessa forma, um ou mais sistemas operacionais hospedeiros podem executar as aplicações em cima do VMM, como mostrado na Figura 2.3 (ANDRADE, 2006).

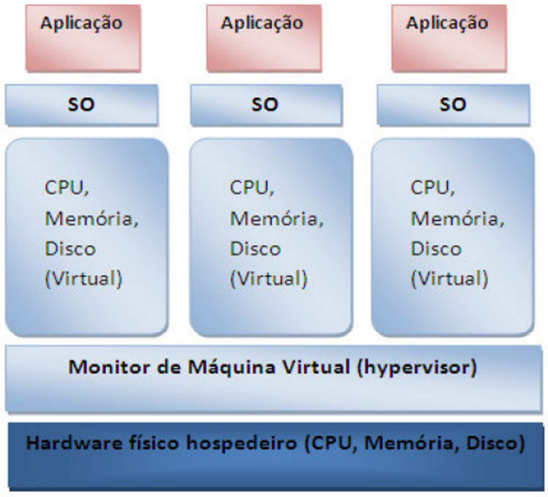


Figura 2.3 - O Sistema operacional e aplicações empilhadas sobre a camada de *software* VMM
Fonte: Andrade (2006)

2.2 Tipos de Virtualização

Existem várias técnicas usadas na virtualização. As principais são: virtualização completa, paravirtualização e recompilação dinâmica. Essas técnicas são descritas nas subseções seguintes.

2.2.1 Virtualização Completa

A virtualização completa (*full virtualization*) é uma técnica utilizada para permitir que qualquer *software* possa ser executado sem alterações. Para isso, essa técnica realiza uma simulação completa do *hardware* da máquina de modo que qualquer sistema operacional possa ser executado. Na virtualização completa, toda uma infraestrutura do *hardware* subjacente é virtualizada, de forma que não é necessário modificar o sistema operacional convidado para que o mesmo execute sobre o VMM (ANDRADE, 2006). A Figura 2.4 (CARISSIMI, 2009) ilustra essa situação. Entretanto, podem ocorrer penalidades em relação ao desempenho da máquina virtual, uma vez que, já que o *hardware* é virtualizado, as instruções devem ser interpretadas pelo VMM. Uma desvantagem dessa técnica na arquitetura x86 é que a mesma não foi projetada tendo em vista a virtualização, mas sim teve uma evolução a partir de versões anteriores (FAVACHO *et al.*, 2008).

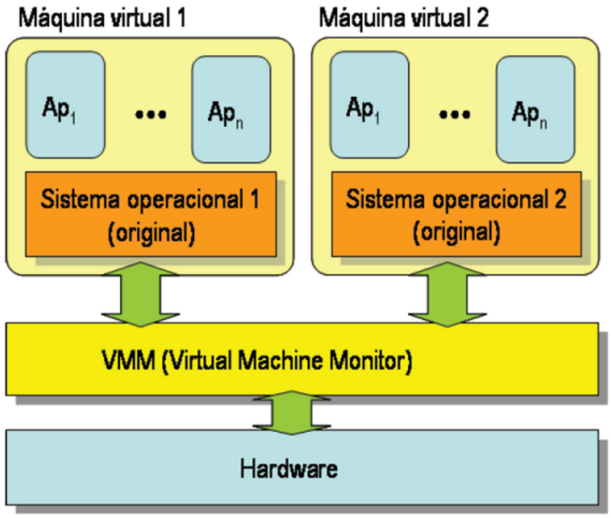


Figura 2.4 - Virtualização completa.
Fonte: Carissimi (2009)

2.2.2 Paravirtualização

Na paravirtualização o sistema que será virtualizado (sistema convidado/hóspede) sofre alterações no *kernel* para que a interação com o monitor de máquinas virtuais seja mais eficiente. Devido a essa modificação, o sistema convidado virtualizado perde em portabilidade, porém é permitido ao mesmo acessar diretamente os recursos do *hardware*, possibilitando ganho em desempenho. O acesso é monitorado pelo monitor de máquinas

virtuais, como é visto na Figura 2.5 (CARISSIMI, 2009), que fornece ao sistema convidado todos os “limites” do sistema, tais como endereços de memória que podem ser utilizados e endereçamento em disco (LAUREANO, 2006).

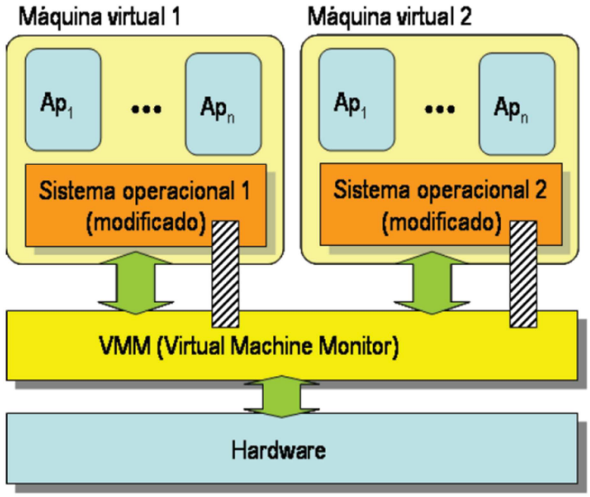


Figura 2.5 - Paravirtualização.
Fonte: Carissimi (2009)

A paravirtualização aparece como uma abordagem alternativa para contornar os problemas de desempenho da virtualização total. Assim, o sistema hóspede deve ser modificado para chamar a máquina virtual sempre que for executar uma instrução ou ação considerada sensível. Na prática isso demonstra a principal desvantagem da paravirtualização, que é a necessidade de alterar todas as instruções de sistema do hóspede por chamadas a máquina virtual para que ela interprete e emule essas ações de forma adequada. Por outro lado, a principal vantagem é que as instruções de usuário não precisam ser alteradas e podem ser executadas diretamente sobre o processador nativo (CARISSIMI, 2009).

Antigamente, a paravirtualização reduzia a complexidade do desenvolvimento das máquinas virtuais, já que historicamente os processadores não suportavam a virtualização nativa (LAUREANO, 2006). Atualmente, os processadores já possuem suporte a virtualização. Assim, a principal razão para utilizar a paravirtualização é a melhora na performance obtida, que compensa as modificações que deverão ser implementadas nos sistemas convidados.

2.2.3 Recompilação Dinâmica

A recompilação dinâmica (*dynamic recompilation*) ou tradução dinâmica (*dynamics translation*) de partes do código de um sistema é bastante utilizada. Com a compilação durante a execução do sistema virtualizado, o sistema pode adequar o código gerado de forma a refletir o ambiente original do programa, onde informações que normalmente não estão disponíveis para um compilador estático tradicional são exploradas, para que o código gerado seja mais eficiente (LAUREANO, 2006). A recompilação dinâmica pode ser utilizada por sistemas como parte de uma estratégia de otimização adaptável, para executar uma representação portável do programa.

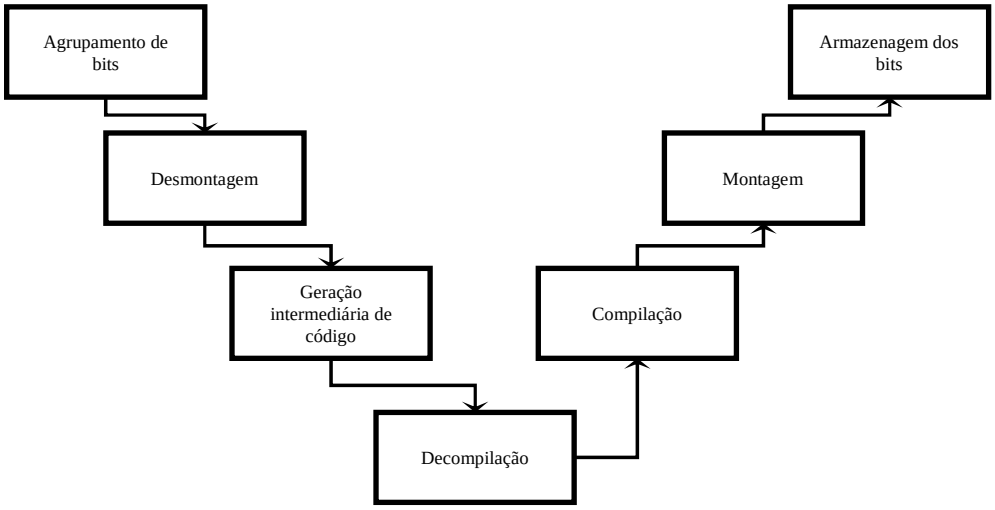


Figura 2.6 - Sete passos da Recompilação Dinâmica.

Dessa forma, como descrito na Figura 2.6, a recompilação dinâmica é composta de sete passos (FAVACHO *et al.*, 2008):

- **Agrupamento de *bits***: quando um programa é compilado e transformado em um arquivo executável, ele armazena uma determinada quantidade de características comuns que identificam a memória, os registradores e as funções do sistema operacional que são manipulados. O emulador ou a máquina virtual tendo o conhecimento sobre o formato executável, e utilizando-se de técnicas heurísticas, recupera os agrupamentos de *bits* do executável e os reordena.
- **Desmontagem (*disassembling*)**: os *bits* são desmontados e transformados em um conjunto de instruções e operadores ordenados em pares.

- **Geração intermediária do código**: as instruções são transformadas para uma representação de máquina independente.
- **Decompilação**: a representação gerada é transformada em uma linguagem de alto nível (como o código na linguagem C).
- **Compilação**: o código gerado é novamente compilado para a nova plataforma.
- **Montagem**: os códigos-objeto (gerados pela compilação) são novamente montados, preparando a criação de um “novo” executável.
- **Armazenagem dos *bits***: os *bits* são agrupados de forma a gerar o novo executável.

Dessa forma, para se conseguir implementar a virtualização de forma adequada, faz-se necessário a escolha e o uso correto das ferramentas de virtualização. Essas ferramentas são abordadas no próximo capítulo.

3 Ferramentas de Virtualização

Atualmente estão disponíveis no mercado inúmeras ferramentas destinadas à construção de ambientes virtualizados. Muitas delas são de uso livre e *open source*, baseadas no licenciamento *GNU General Public License* (FREE SOFTWARE FOUNDATION, 2007). Outras são ferramentas comerciais, que precisam ser licenciadas, mas também possuem versões sem custo de licenciamento, porém com funções limitadas.

Pollon (2008) realizou um levantamento de diversas ferramentas de virtualização, que foram limitadas aquelas que funcionassem tanto com sistemas operacionais Linux quanto Windows, relacionadas no Quadro 3.1.

Nome	Criador	SO Hospedeiro	SO Hóspede
Citrix XenServer	Citrix Systems Inc.	Nenhum	Windows Linux Solaris Outros
HyperV	Microsoft	Windows 2008 w/Hyper V Role	Windows Linux
KVM	Qumranet	Linux	Windows Linux
LynxSecure	LinuxWorks	Nenhum	LynxOS Linux Windows
Oracle VM	Oracle Corporation	Nenhum	Windows Linux
Parallels Workstation	Parallels, Inc.	Windows Linux	Windows Linux FreeBSD Solaris
Proxmox Virtual Enviroment	ProMox	Debian	Linux Windows Unix
RTS Hypervisor	RealTime Systems	Nenhum	Windows Linux
SimNow	AMD	Linux Windows	Linux Windows
Simics	Virtutech	Windows Linux Solaris	Linux FreeBSD Windows TinyOS Solaris outros
Sun xVM	Sun Microsystems	Nenhum	Windows Linux
VirtualBox	Sun Microsystems	Windows Linux MacOS Solaris	DOS Windows Linux OS/2 FreeBSD Solaris
Virtual Iron	Virtual Iron Software Inc.	Nenhum	Windows RedHat SuSE
VirtualPC	Microsoft	Windows	Windows Linux OpenSolaris
Virtual Server	Microsoft	Windows	Windows Linux
Virtuozzo	SWsoft	Linux Windows	Linux Windows
VMware ESX	VMware	Nenhum	Linux Solaris Windows Outros
VMware ESXi	VMware	Nenhum	Linux Solaris Windows Outros

VMware Server	VMware	Windows Linux	Linux Solaris Windows Outros
Xen	Universidade Cambridge, Intel, AMD	Nenhum FreeBSD Linux Solaris	Linux Solaris Windows

Quadro 3.1 – Ferramentas de virtualização que dão suporte a Windows e Linux
Fonte: Pollon (2008)

Neste trabalho, as ferramentas selecionadas como foco de estudo devem possuir a funcionalidade de virtualização completa do tipo 1, ou seja, funcionar diretamente sobre o *hardware,* independente do sistema operacional. Dessa forma, eliminando as ferramentas do Quadro 3.1 que não atendem esse requisito, restaram as ferramentas abaixo relacionadas:

a. Xen
b. RTS Hypervisor
c. LynxSecure
d. Sun xVM
e. Oracle VM
f. Citrix XenServer
g. Virtual Iron
h. VMware ESX
i. VMware ESXi

A seguir, essas ferramentas serão mais bem detalhadas nas próximas seções.

3.1 Xen

O Xen é um VMM *open source* para a arquitetura x86, que utiliza a técnica de paravirtualização como padrão (CLARK, 2002), mas também utiliza a virtualização completa. Desenvolvido originalmente pelo *Systems Research Group at the University of Cambridge Computer Laboratory* como parte do projeto *XenoServers*, cujo objetivo era criar uma infraestrutura global para computação distribuída, o Xen permite que uma única máquina física seja eficientemente dividida. Permitindo assim, a execução simultânea de múltiplos sistemas hóspedes, com proteção e isolamento dos recursos e um bom desempenho.

Para que um sistema hóspede possa ser executado sob o Xen, ele precisa ser alterado, devido à técnica de paravirtualização que é utilizada na construção da máquina virtual. Em ambientes que utilizem sistemas operacionais *open source*, como o Linux, por exemplo, que possibilita a recompilação do núcleo, isto é perfeitamente possível. Já em sistemas não *open source*, como no caso do Windows, que não permite alterações, torna-se impossível executá-lo sobre um ambiente Xen paravirtualizado (JUNIOR, 2008).

Para o Xen utilizar a técnica de virtualização completa, é necessário que o equipamento físico possua um processador com a extensão *Intel Virtualization Technology* (IVT) ou *AMD Virtualization* (AMD-V), que implementa a virtualização com a ajuda do *hardware*. A utilização desta técnica no Xen simplifica a sua implementação e possibilita a instalação de sistemas como o Windows que não é suportado pelo Xen na técnica de paravirtualização .

No Xen, as máquinas virtuais são chamadas de `Dom` ou domínios. O sistema hospedeiro ou o *hypervisor*, no caso de virtualização direta no *hardware*, recebe o nome `Dom0` ou domínio zero e tem a responsabilidade de controlar os demais sistemas hospedados, chamados de `DomU` e que não possuem os mesmos privilégios do `Dom0` (CLARK, 2002).

Os `DomU`s se comunicam diretamente com o *hypervisor* que tem o controle da memória e do processador do sistema físico. Os demais periféricos de *hardware* são acessados pelos `DomUs` por meio dos *drivers* do `Dom0`, o que possibilita que todos os periféricos compatíveis com o `Dom0` estejam disponíveis para os `DomU`s, mesmo que estes sejam sistemas operacionais diferentes (JUNIOR, 2008).

O Xen pode ser aplicado nos mais diversos ambientes, tendo como principais cenários de uso a consolidação de servidores, o suporte a aplicações legadas e a computação em *cluster*. Diversas distribuições Linux já suportam o Xen de forma nativa, o que facilita bastante o processo de instalação (POLLON, 2008).

3.2 RTS Hypervisor

Criado pela empresa alemã *Real-Time Systems GmbH* (2010), é uma solução completa de virtualização em *hardware* Intel x86 que contenha extensões de virtualização. Suporta diversos sistemas operacionais, tais como Windows e Linux, podendo funcionar

como paravirtualização, no caso de uso de sistemas operacionais personalizados ou com intuito de adquirir maior desempenho.

O monitor de máquina virtual é do tipo 1, isto é, permite que qualquer sistema operacional seja usado de forma independente um do outro. Dessa forma, um sistema operacional pode ser reinicializado sem afetar os outros sistemas instalados (*REAL-TIME SYSTEMS*, 2010).

O RTS *Hypervisor* não foi desenvolvido baseado em nenhuma solução existente, como o KVM ou o Xen, sendo desenvolvido para aplicações do tipo *real-time* ou embarcadas. Ao contrário das outras soluções de virtualização, a sua política de particionamento de *hardware* possibilita o ganho em desempenho e o uso em sistemas específicos. O RTS *Hypervisor* pode atribuir para cada máquina virtual um conjunto de *hardware* independente para cada máquina virtual criada (*REAL-TIME SYSTEMS*, 2010).

3.3 Sun xVM

O Sun xVM foi desenvolvido pela *Sun Microsystems* em 2007, baseado na arquitetura do Xen, para funcionar com máquinas que utilizem processadores x86 que contenham extensões de virtualização (processadores Intel VT ou AMD-V). Cada instância de máquina virtualizada é chamada de domínio. Assim, como no caso do Xen, as máquinas virtuais são chamadas de `Dom` ou domínios. O *hypervisor* recebe o nome `Dom0` ou domínio zero e tem a responsabilidade de controlar os demais sistemas hospedados, chamados de `DomU` e que não possuem os mesmos privilégios do `Dom0`. Dessa forma, os `DomU`s utilizam os *drivers* do `Dom0` para comunicação com os periféricos (COTTEN, 2008).

O *hypervisor* virtualiza o *hardware* de forma transparente, compartilhando e particionando os recursos de *hardware* entre os `DomUs`. O *hypervisor* se baseia no controle dos domínios para prover os domínios convidados criados e determinar que recursos os domínios possam acessar.

Além da virtualização completa, o Sun xVM também suporta paravirtulização, funcionando apenas com sistemas operacionais específicos, tais como o Solaris, Linux e o FreeBSD, por exigir mudanças para suportar os dispositivos virtuais.

3.4 LynxSecure

O *LynxSecure* distribuído pela *LynuxWorks Inc*. é um *hypervisor open source*, normalmente utilizado por usuários de aplicações Linux e POSIX (*Portable Operating System*). O LynxSecure é muito usado pelas instituições militares americanas, devido aos seus recursos de segurança (LYNUXWORKS, 2010).

O LynxSecure desenvolvido para a arquitetura x86 que contenha extensões de virtualização (processadores *Intel VT* ou AMD-V), permite a execução de diferentes tipos de sistemas operacionais em modo de virtualização completa, podendo funcionar com paravirtualização, no caso de sistemas operacionais que permitam modificação de seus núcleos (LYNUXWORKS, 2010).

3.5 Oracle VM

O Oracle VM, desenvolvido pela *Oracle Corp*., é um conjunto de soluções para gerenciamento, manutenção, migração e controle e balanceamento de carga de máquinas virtuais. Dessa forma, sua comercialização é divida em pacotes que depende do conjunto de aplicações inclusas nos pacotes.

O *hypervisor* do Oracle VM, assim como no Sun xVM, foi baseado no Xen (ORACLE, 2010). O Oracle VM possui os mesmos pré-requisitos do Xen e suas características de funcionamento.

3.6 Citrix XenServer

O *Citrix XenServer*, desenvolvido pela *Citrix Systems Inc.*, assim como o nome já diz, possui seu *hypervisor* baseado no Xen. Dessa forma, o *Citrix XenServer* possui os mesmos pré-requisitos do Xen e suas características de funcionamento (CITRIX, 2010).

O diferencial deste produto está na forma em que ele é comercializado pela *Citrix*, pois é distribuído em formato de pacotes de soluções, onde dependendo do pacote pode ser gratuito ou não. Os pacotes contêm não só o *hypervisor*, como outros produtos para migração, conversão, gerenciamento, controle e manutenção, além de aplicações para realizar balanceamento de carga, recuperação de desastres, gerenciamento de energia,

criação de pontos de restauração, entre outros recursos, disponíveis somente nas versões pagas.

3.7 Virtual Iron

O V*irtual Iron* criado pelo *Virtual Iron Software Inc*. é um monitor de máquina virtual que habilitava recursos e capacidade de gerenciamento de ambientes virtualizados em *data centers*, muito utilizado como ferramenta para consolidação de servidores. Seu desenvolvimento, assim como no Sun xVM, foi baseado no Xen (ORACLE, 2010).

Em meados de 2009, a *Oracle Corp*., devido o nível de maturidade atingido pelo *Virtual Iron*, comprou a *Virtual Iron Software Inc*, com o intuito de adquirir o conhecimento do funcionamento dessa solução para, mais tarde, aplicar em sua solução própria de virtualização (Oracle VM e Sun xVM) (ORACLE, 2010).

Poucos meses após a compra da *Virtual Iron*, a Oracle descontinuou o produto, prestando somente o serviço de suporte para os clientes já existentes.

3.8 VMware ESX

Criado no ano de 1999, o *VMware* é um dos aplicativos de virtualização para plataforma x86 mais populares atualmente. Ele foi a primeira solução de virtualização para a arquitetura x86 e fornece uma implementação completa para o sistema convidado. A *VMware Inc*., empresa que desenvolve o *software* de mesmo nome, possui uma vasta linha de produtos voltados para a virtualização. A *VMware* também possui algumas versões de seus produtos sem custo de licenciamento. Estas versões, normalmente, possuem algum tipo de limitação (VMWARE, 2010).

No *VMware*, o monitor emula algumas instruções, denominadas instruções sensíveis, objetivando representar corretamente o processador virtual para cada máquina hospedada. Para executar essas instruções sensíveis, a máquina virtual utiliza o mecanismo de interrupção de *software* (*trap*) do processador (VMWARE, 2010), com intuito de executar as instruções em multitarefa.

No entanto, nem sempre os processadores da arquitetura x86 conseguem capturar tais instruções. Para controlar as instruções sensíveis que não foram capturadas corretamente, o *VMware* utiliza a técnica de reescrita binária, que possibilita uma análise

prévia de todas as instruções antes da execução (LAUREANO, 2006). Neste caso, o monitor substitui as instruções sensíveis por pontos de parada que serão capturados e executados pelo processador.

No *VMware*, o próprio sistema hóspede gerencia a memória e, para garantir que não ocorrerá a tentativa de utilização do mesmo endereço de memória pelo sistema hóspede e pelo sistema hospedeiro, o *VMware* reserva uma parte da memória para seu uso exclusivo. É essa memória reservada previamente que é disponibilizada ao sistema hóspede (POLLON, 2008).

Dentre os produtos desenvolvidos pela *VMware Inc*., destacam-se: o *VMware ESX Server* que é voltado para servidores de grande porte; o *VMware Server*, disponibilizado gratuitamente a partir de 2006, que é indicado para aplicação em ambientes de pequeno porte (VMWARE, 2010).

O *VMware* ESX Server utiliza a abordagem clássica na sua construção. Trata-se de um produto indicado para utilização em ambientes de produção, em servidores de grande porte. O *VMware* ESX conta com um sistema operacional Linux, chamado de console de serviço, para desempenhar algumas funções de gerenciamento, inclusive a execução de lista de comandos e a instalação de agentes de terceiros para monitoramento de *hardware*, cópia de segurança (*backup*) ou gerenciamento de sistemas (VMWARE, 2010). Cada máquina virtual criada no *VMware* ESX Server possui todos os componentes de um sistema real, como processador, memória, disco rígido, placa de rede e BIOS. Para o sistema hóspede, é como se ele estivesse sendo executado em um computador real e, por este motivo, não há a necessidade de alterações no sistema hóspede (POLLON, 2008).

O *VMware ESX Server* possui um sistema de arquivos próprio, o *Virtual Machine File System* (VMFS). Trata-se de um sistema de arquivos distribuído que permite o acesso concorrente ao mesmo volume VMFS por diversos sistemas hóspedes. O VMFS é otimizado para operações de entrada e saída (E/S) com arquivos grandes, realidade das imagens das máquinas virtuais e este é um dos principais diferenciais deste sistema de arquivos. Ele também possibilita a criação de áreas de armazenamento que podem ser compartilhadas entre diferentes hóspedes, com diferentes sistemas operacionais (JUNIOR, 2008).

O *VMware Server*, por sua vez, utiliza a abordagem de virtualização hospedada, ou do tipo 2. Trata-se de uma ferramenta voltada para ambientes de pequeno e médio porte.

Por ser uma máquina virtual do tipo 2, a execução do *VMware Server* acontece sobre um sistema operacional hóspede que pode ser tanto *Windows* quanto *Linux* (POLLON, 2008).

O *VMware Server* é disponibilizado gratuitamente e tem suporte aos seguintes sistemas operacionais hóspedes: *Windows*, *Linux*, *Novell Netware* e *Sun Solaris*. No entanto, outros sistemas operacionais podem ser instalados utilizando-se a opção “outros”, disponível no momento da criação da máquina virtual.

Atualmente, a *VMware* tem disponibilizado uma solução composta por uma suíte de ferramentas para virtualização, gerenciamento, manutenção e migração de máquinas virtuais e redes virtuais, otimização de recursos, aplicação de disponibilidade e capacidade operacional de automatização, chamada *VMware vSphere*.

3.9 VMware ESXi

O *VMware* ESXi possui todas as funcionalidades do *VMware* ESX, sendo uma evolução desse *hypervisor* anteriormente citado. A principal diferença entre o *VMware* ESX e o ESXi está no console de serviço que existe na versão ESX, porém na versão ESXi foi removido, reduzindo significativamente a ocupação de espaço (VMWARE, 2010).

Ao remover o console de serviço, o *VMware* ESXi segue a tendência de migrar o recurso de gerenciamento da interface local de linha de comando para ferramentas de gerenciamento remoto. A função do console de serviço é substituída pelas interfaces de linha de comando remotas e se adere aos padrões de gerenciamento do sistema (VMWARE, 2010).

No próximo capítulo é descrito a metodologia de aplicada nesse trabalho e os procedimentos e as ferramentas selecionadas para avaliação.

4 METODOLOGIA

Este trabalho é caracterizado como sendo uma pesquisa de natureza aplicada, pois tem por objetivo gerar conhecimentos para aplicação prática à solução de problemas específicos (SILVA, 2001).

Pela forma de sua abordagem, este trabalho se caracteriza por uma pesquisa quantitativa, cujo método é um levantamento de dados por meio de coleta de dados baseadas em métricas. Dessa forma, podendo formar opiniões e informações e assim classificá-las e analisá-las (SILVA, 2001).

Quanto aos seus objetivos, este trabalho pode ser classificado como uma pesquisa exploratória, porque visa tornar um problema explícito ou a construir hipóteses, envolvendo levantamento bibliográfico, análise de exemplos, através de estudos de caso (GIL, 1991).

Assim, quanto ao procedimento, este trabalho pode ser classificado como do tipo estudo de caso, pois envolve o estudo de um ou poucos objetos de maneira que se permita o seu amplo e detalhado conhecimento (GIL, 1991).

4.1 Procedimentos Metodológicos

O ambiente de desenvolvimento é uma servidor HP Proliant ML115 G5 Quad-Core AMD *Opteron* 1352 +2100MHz, possuidor da extensão de virtualização IVT e AMD-V, com 3GB de memória e 150GB de HD e as ferramentas de virtualização elencadas foram o *VMware* ESXi 4.1.0 e o XenServer 5.6.2 da Citrix, devido a capacidade de poderem funcionar em modo de virtualização completa do tipo 1 nesse ambiente e por possuírem compatibilidade com o *hardware* usado como base de testes.

Os sistemas operacionais hóspedes escolhidos como base de testes foram o Windows 2003 Server e o Fedora 14. O Windows 2003 por ser uma versão largamente difundida e utilizada nas organizações. No caso do Fedora, por ser um sistema operacional gratuito disponibilizado pela RedHat, utilizado como laboratório para o desenvolvimento de seu sistema principal.

As ferramentas elencadas para avaliação dos ambientes virtualizados foram o *PassMark Performance Test* 7.0 e o *Phoronix Test Suite* 3.0.1, para os sistemas Windows e Linux, respectivamente.

Os testes foram divididos em duas fases:

a) Análise de desempenho com o sistema Windows.
b) Análise de desempenho com o sistema Linux.

Cada um desses testes foi realizado com cada um dos VMM escolhidos, montando ambientes de *hardware* idênticos para cada uma das VMs (2 processadores, 1GB de memória e 20GB de disco).

Para efeito de análise, foram considerados relevantes os dados obtidos baseados em modalidades de desempenho em operações realizadas por processador, memória e disco.

O *PassMark Performance Test* é uma ferramenta desenvolvida pela *PassMark Software*, utilizada para realizar *benchmark* usando uma variedade de diferentes testes de velocidade e desempenho para comparação de resultados entre máquinas com a plataforma Windows. Essa ferramenta possui 28 testes padrão em sua suíte, distribuídos em testes de CPU, operações gráficas e vídeo, operações de leitura, escrita e busca em disco, acesso em memória, além de testes para unidades de CD/DVD.

Para o *PassMark* foram realizados 35 testes para cada uma das modalidades (processador, memória e disco), com um intervalo de confiança de 95%. Os demais resultados dos testes relativos as outras categorias foram ignorados.

O *Phoronix Test Suite* é uma ferramenta *open source* desenvolvida pela Phoronix Media desde 2004. Originalmente desenvolvido para testes extensos e automatizados para a plataforma Linux, o Phoronix também pode ser usado para outras plataformas. Por possuir um conjunto muito grande de ferramentas de testes, foram elencadas somente as seguintes ferramentas para as modalidades escolhida, por critério de semelhança aos testes do *PassMark*:

1) Ferramentas de teste de processador:
    a) Compress-gzip-1.1.0: verifica o tempo gasto pela compressão de arquivos utilizando o algoritmo de compressão GZip.
    b) Bork-1.0.0: utilitário de encriptação escrito em Java, que verifica o tempo gasto para a encriptação de arquivos.
    c) Sudokut-1.0.0: teste baseado na resolução de um quebra-cabeça chamado *sudokut*, verificando o tempo gasto pela resolução de 100 *sudokuts*.
2) Ferramenta de teste de memória:

a) Ramspeed-1.4.0: teste que verifica a performance da memória RAM, realizando operações de cópia, escalonamento, adição e operações com inteiros e ponto flutuante.

3) Ferramentas de teste de disco:
   a) Fio-1.1.0: conhecido como *Flexible IO Tester*, é uma ferramenta de teste de disco dependente da biblioteca *AIO Access*, realizando um teste para o *Intel IOMeter File Server Access Pattern* e outro teste chamado *Example Network Job*.
   b) Tiobench-1.1.0: conhecido *Threaded I/O Tester*, é uma ferramenta para medir a performance de disco por sistema de arquivo, realizando testes de escrita, escrita randômica, leitura e leitura randômica com *threads* de 32MB de tamanho, com 4, 8, 16 e 32 *threads*.
   c) Aio-stress-1.1.0: é um teste de disco assíncrono que utiliza um arquivo de 2GB de tamanho e blocos de 64KB.

Para estudo de desempenho e concorrência dos recursos foram realizados testes com VMs executando simultaneamente as ferramentas de análise, assim concorrendo por recursos, partindo da execução de apenas uma máquina até a execução simultânea de cinco máquinas idênticas, com cada sistema operacional hóspede.

Com dados obtidos e tabulados foi calculado e uma média dos valores obtidos e, desta forma, realizado as análises e conclusões do presente trabalho.

5 RESULTADOS E DISCUSSÃO

Nesse capítulo são apresentados os dados obtidos nos testes com os sistemas hóspedes Windows e Linux selecionados nos *hypervisors* escolhidos.

5.1 Sistema Operacional Hóspede Windows 2003 Server

Nessa seção são apresentados os dados obtidos nos testes com o *hypervisor VMware* ESXi 4.1.0 e o Citrix XenServer 5.6.2, após os resultados gerados pela análise das ferramentas de benchmark escolhidas, com o sistema operacional hóspede Windows 2003 Server.

5.1.1 Resultados do *PassMark*

O *PassMark* retornou os seguintes resultados para os testes de processador após a execução simultânea das VMs:

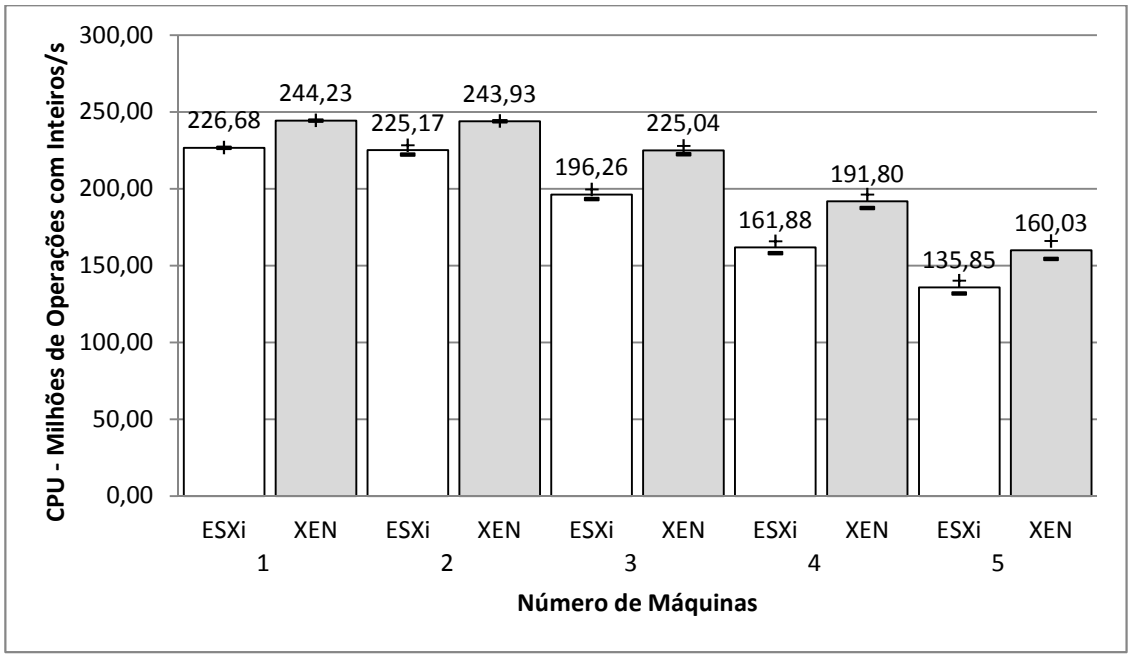


Figura 5.1 - Teste de processador: operações com inteiros por segundo no Windows 2003 utilizando *PassMark Performance Test.*

A Figura 5.1 representa os valores médios obtidos no teste de desempenho em milhões de cálculos utilizando operações com valores inteiros por segundo (MOPS). Os valores médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para

uma, duas, três, quatro e cinco máquinas executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.1.

Tabela 5.1 – Teste de processador: resultados médios do *PassMark* para milhões de operações com inteiros/s (MOPS)

# Máquinas	Hypervisor	Média (MOPS)	Erro (±)
1	ESXi	226,68	0,12
	XEN	244,23	0,09
2	ESXi	225,17	3,01
	XEN	243,93	0,11
3	ESXi	196,26	3,14
	XEN	225,04	2,75
4	ESXi	161,88	3,82
	XEN	191,80	4,36
5	ESXi	135,85	4,16
	XEN	160,03	5,82

Esses resultados denotam a degradação gerada pela concorrência pelos mesmos recursos, sendo mais acentuada logo na concorrência gerada por três máquinas. Também pode ser observado o melhor desempenho obtido pelo XenServer.

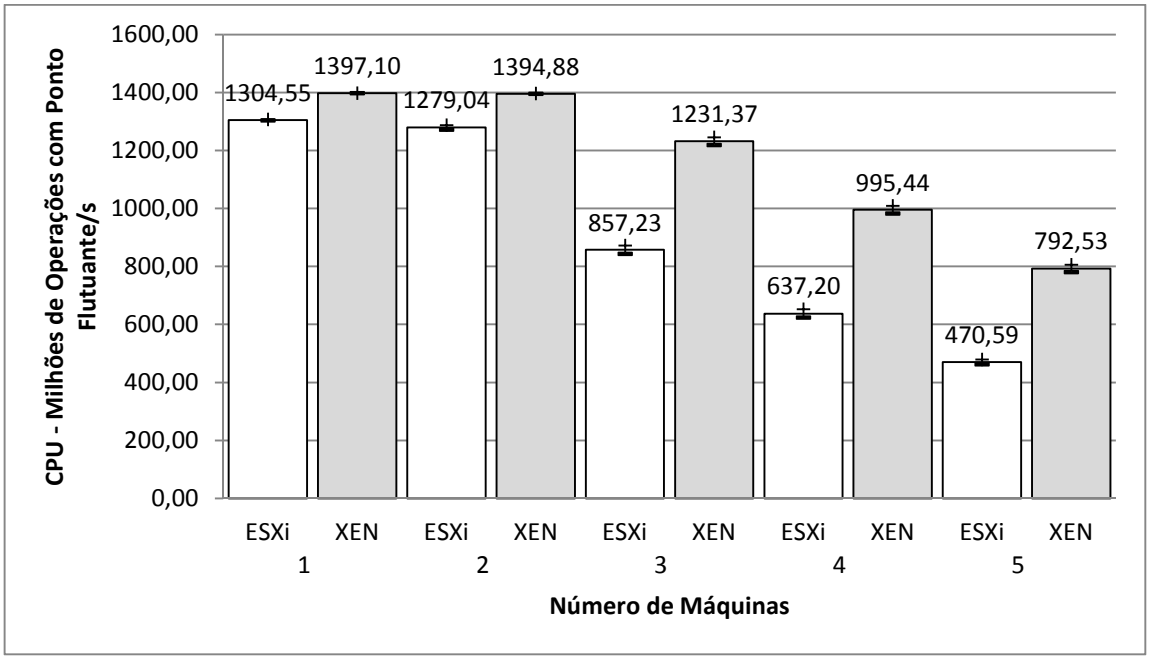


Figura 5.2 - Teste de processador: operações com ponto flutuante por segundo no Windows 2003 utilizando *PassMark Performance Test.*

A Figura 5.2 representa os valores médios obtidos no teste de desempenho em milhões de cálculos utilizando valores com ponto flutuante por segundo (MFLOPS). Os valores médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para

uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.2.

Tabela 5.2 – Teste de processador: resultados médios do *PassMark* para milhões de operações com ponto flutuante/s (MFLOPS)

# Máquinas	Hypervisor	Média (MFPLOS)	Erro (±)
1	ESXi	1304,55	1,53
	XEN	1397,10	0,36
2	ESXi	1279,04	7,41
	XEN	1394,88	0,64
3	ESXi	857,23	14,76
	XEN	1231,37	13,69
4	ESXi	637,20	14,39
	XEN	995,44	13,40
5	ESXi	470,59	8,15
	XEN	792,53	13,07

Esses resultados denotam, assim como no caso anterior, a degradação gerada pela concorrência pelos mesmos recursos, sendo mais acentuada logo na concorrência gerada por três máquinas. Além disso, novamente, denota-se o ganho em desempenho do XenServer quando comparado ao *VMware* ESXi.

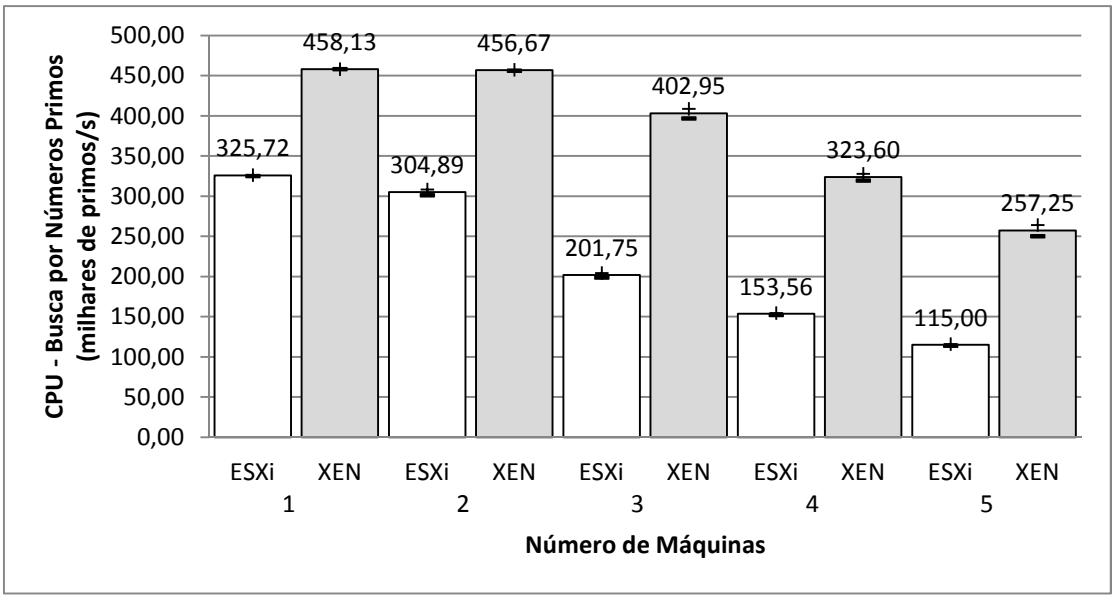


Figura 5.3 - Teste de processador: busca por números primos (milhares por segundo) no Windows 2003 utilizando *PassMark Performance Test*.

A Figura 5.3 representa os valores médios obtidos no teste de desempenho em busca por números primos, numa taxa de milhares de números por segundo. Os valores

médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.3.

Tabela 5.3 – Teste de processador: resultados médios do *PassMark* para operações de busca por números primos (milhares/s)

# Máquinas	Hypervisor	Média (milhares/s)	Erro (±)
1	ESXi	325,72	0,48
	XEN	458,13	0,06
2	ESXi	304,89	3,59
	XEN	456,67	0,57
3	ESXi	201,75	2,77
	XEN	402,95	5,95
4	ESXi	153,56	1,16
	XEN	323,60	4,18
5	ESXi	115,00	1,04
	XEN	257,25	7,11

Esses resultados denotam as mesmas características do teste anterior, com destaque para o desempenho do XenServer que chega a ser 40% superior que o ESXi nessa categoria.

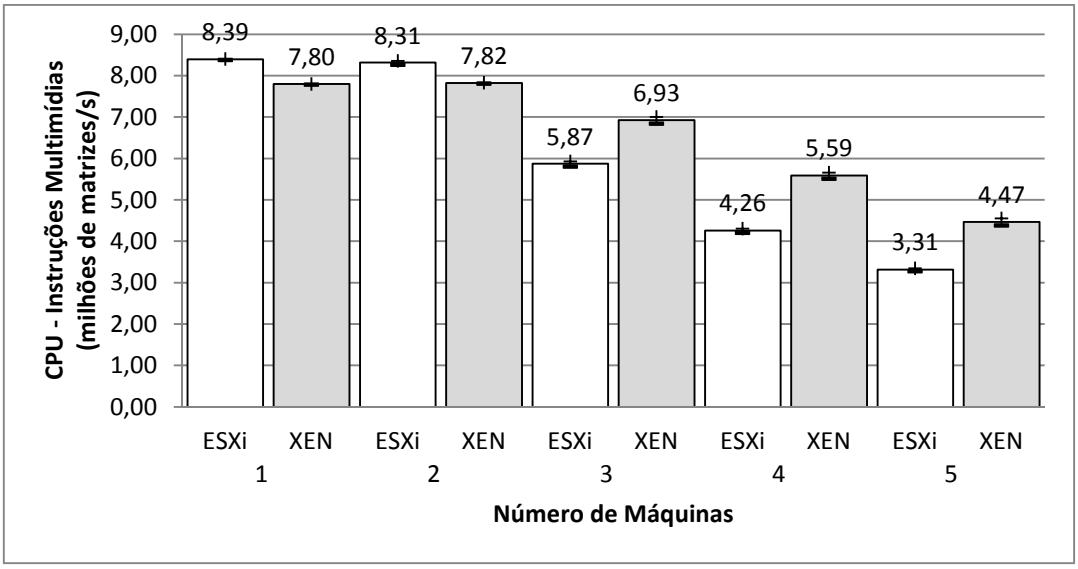


Figura 5.4 - Teste de processador: instruções multimídias (milhões de matrizes por segundo) no Windows 2003 utilizando *PassMark Performance Test*.

A Figura 5.4 representa os valores médios obtidos no teste de desempenho em operações com instruções multimídias, numa taxa de milhões de matrizes por segundo. Os valores médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para

uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.4.

Tabela 5.4 – Teste de processador: resultados médios do *PassMark* para operações com instruções multimídia (milhares/s)

# Máquinas	Hypervisor	Média (milhares/s)	Erro (±)
1	ESXi	8,39	0,01
	XEN	7,80	0,01
2	ESXi	8,31	0,05
	XEN	7,82	0,01
3	ESXi	5,87	0,06
	XEN	6,93	0,08
4	ESXi	4,26	0,05
	XEN	5,59	0,07
5	ESXi	3,31	0,04
	XEN	4,47	0,08

Esses resultados denotam uma variação na degradação, com um fator interessante, inicialmente o *VMware* possui um desempenho discretamente superior, mas significativo, e após o acréscimo de máquinas simultâneas, o XenServer passa a se destacar nessa categoria.

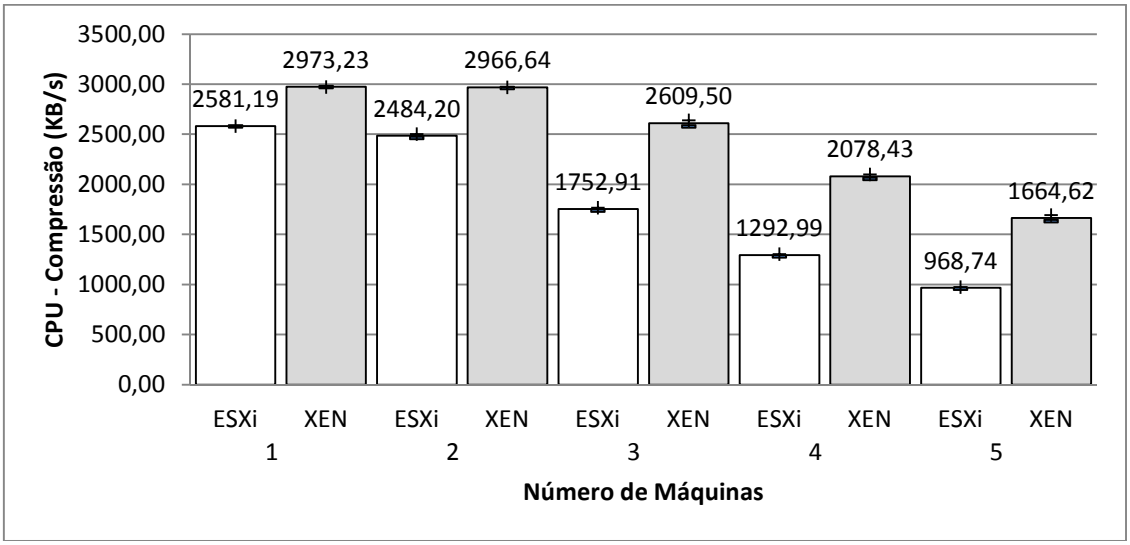


Figura 5.5 - Teste de processador: operações de compressão de arquivo (Kbytes por segundo) no Windows 2003 utilizando *PassMark Performance Test*.

A Figura 5.5 representa os valores médios obtidos no teste de desempenho em operações de compressão de arquivos, numa taxa de Kbytes por segundo. Os valores obtidos médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para

uma, duas, três, quatro e cinco máquinas, com intervalo de confiança de 95%, podem ser observados na Tabela 5.5.

Tabela 5.5 – Teste de processador: resultados médios do *PassMark* para operações compressão de arquivo (KB/s)

# Máquinas	Hypervisor	Média (KB/s)	Erro (±)
1	ESXi	2581,19	2,59
	XEN	2973,23	0,47
2	ESXi	2484,20	18,40
	XEN	2966,64	1,89
3	ESXi	1752,91	13,77
	XEN	2609,50	29,60
4	ESXi	1292,99	10,48
	XEN	2078,43	21,54
5	ESXi	968,74	7,77
	XEN	1664,62	29,21

Esses resultados denotam uma variação na degradação já observado e o destaque para o XenServer nessa categoria.

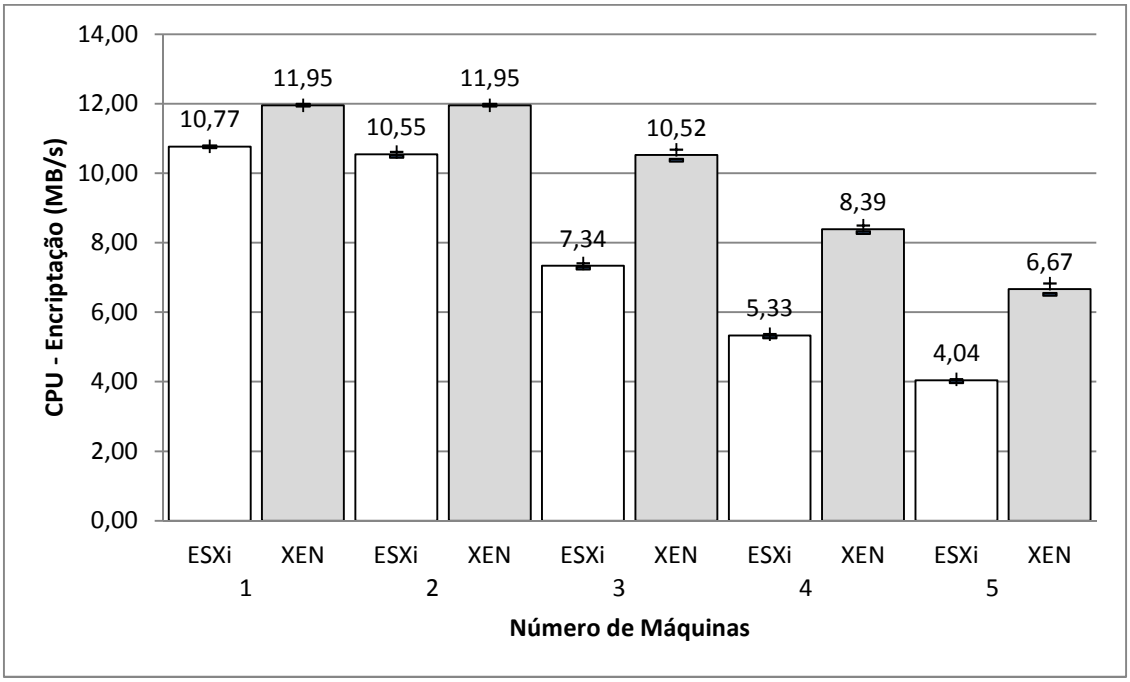


Figura 5.6 - Teste de processador: encriptação de dados (Mbytes por segundo) no Windows 2003 utilizando *PassMark Performance Test*.

A Figura 5.6 representa os valores médios obtidos no teste de desempenho em operações de encriptação de dados, numa taxa de Mbytes por segundo. Os valores obtidos médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma,

duas, três, quatro e cinco máquinas, com intervalo de confiança de 95%, podem ser observados na Tabela 5.6.

Tabela 5.6 – Teste de processador: resultados médios do *PassMark* para operações encriptação de dados (MB/s)

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	10,77	0,01
	XEN	11,95	0,00
2	ESXi	10,55	0,07
	XEN	11,95	0,00
3	ESXi	7,34	0,07
	XEN	10,52	0,15
4	ESXi	5,33	0,04
	XEN	8,39	0,10
5	ESXi	4,04	0,03
	XEN	6,67	0,15

Esses resultados denotam uma variação na degradação e o destaque para o XenServer nessa categoria.

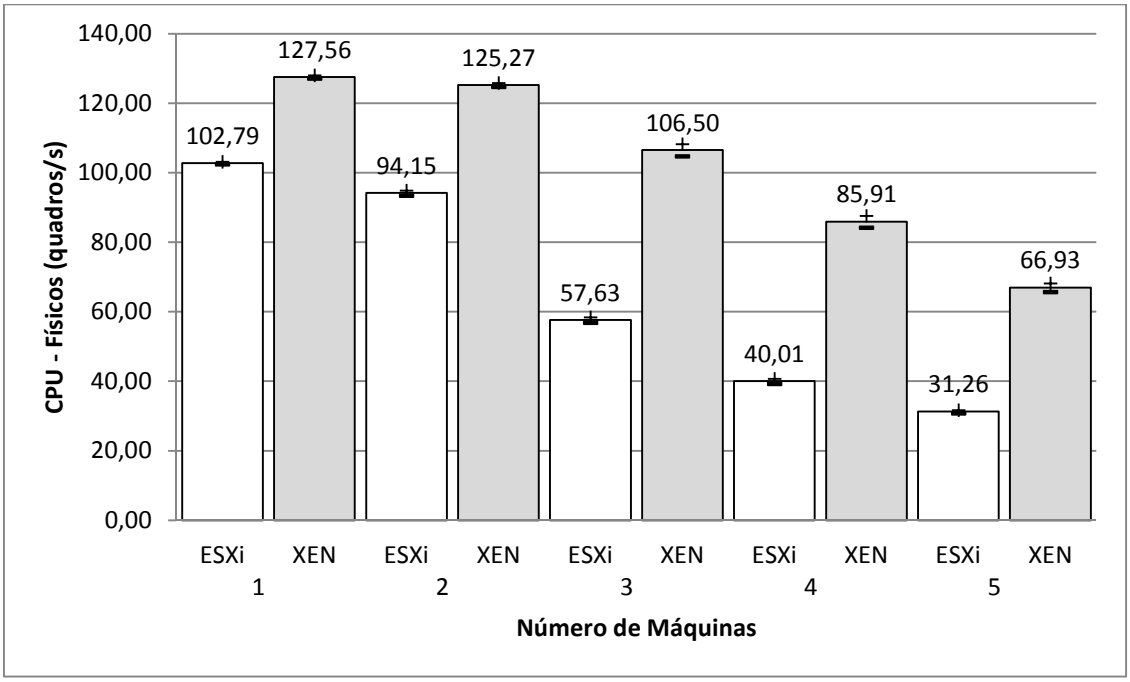


Figura 5.7 - Teste de processador: Físicos (quadros por segundo) no Windows 2003 utilizando PassMark Performance Test.

A Figura 5.7 representa os valores médios obtidos no teste de desempenho em operações de físicas, numa taxa de quadros por segundo. Os valores médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco

máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.7.

Tabela 5.7 – Teste de processador: resultados médios do *PassMark* para desempenho operações físicas (quadros/s)

# Máquinas	Hypervisor	Média (quadros/s)	Erro (±)
1	ESXi	102,79	0,37
	XEN	127,56	0,44
2	ESXi	94,15	0,77
	XEN	125,27	0,58
3	ESXi	57,63	0,82
	XEN	106,50	1,78
4	ESXi	40,01	0,73
	XEN	85,91	1,70
5	ESXi	31,26	0,44
	XEN	66,93	1,28

Esses resultados denotam a evolução da taxa de variação na degradação do desempenho e o destaque para o XenServer nessa categoria.

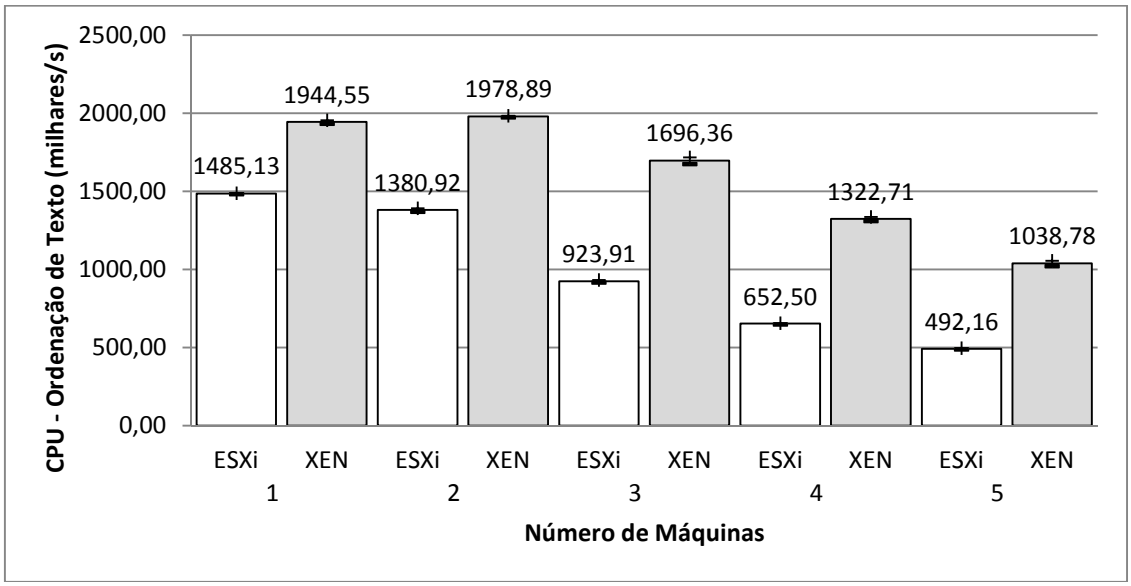


Figura 5.8 - Teste de processador: ordenação de texto (milhares por segundo) no Windows 2003 utilizando PassMark Performance Test.

A Figura 5.8 representa os valores médios obtidos no teste de desempenho em operações de ordenação de texto, numa taxa de milhares por segundo. Os valores obtidos médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma,

duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observadas na Tabela 5.8.

Tabela 5.8 – Teste de processador: resultados médios do *PassMark* para operações de ordenação de texto (milhares/s)

# Máquinas	Hypervisor	Média (milhares/s)	Erro (±)
1	ESXi	1485,13	3,54
	XEN	1944,55	11,49
2	ESXi	1380,92	11,47
	XEN	1978,89	4,86
3	ESXi	923,91	8,00
	XEN	1696,36	21,04
4	ESXi	652,50	4,11
	XEN	1322,71	13,92
5	ESXi	492,16	3,73
	XEN	1038,78	17,56

Esses resultados denotam a evolução da taxa de variação na degradação do desempenho e o destaque para o XenServer nessa categoria.

Em resumo, na categoria testes de processador em sistema operacional hóspede *Windows 2003 Server*, o XenServer obteve um desempenho superior ao *VMware* ESXi em todas as categorias analisadas, sendo uma opção a ser considerada, quando o enfoque for desempenho em processamento para ambientes de máquinas virtualizadas.

Na categoria memória, a forma de gerenciamento de memória das máquinas virtuais desses dois *hypervisors* são diferentes, o XenServer não virtualiza a memória disponível, ele praticamente compartilha o recurso de memória entre suas máquinas virtuais, ou seja a quantidade de memória total utilizada pelas suas VMs não pode ultrapassar a quantidade de memória física disponível sendo dividida entre suas VMs que se encontram ativas, porém é possível realizar otimizações dinâmicas de seu uso, melhorando o desempenho. No caso do *VMware*, é possível utilizar máquinas virtuais simultaneamente, ultrapassando a quantidade de memória física, porém gerando perda em desempenho, além de ser possível gerar otimizações para seu melhor uso.

Desta forma, o PassMark retornou os seguintes resultados para os testes de memória:

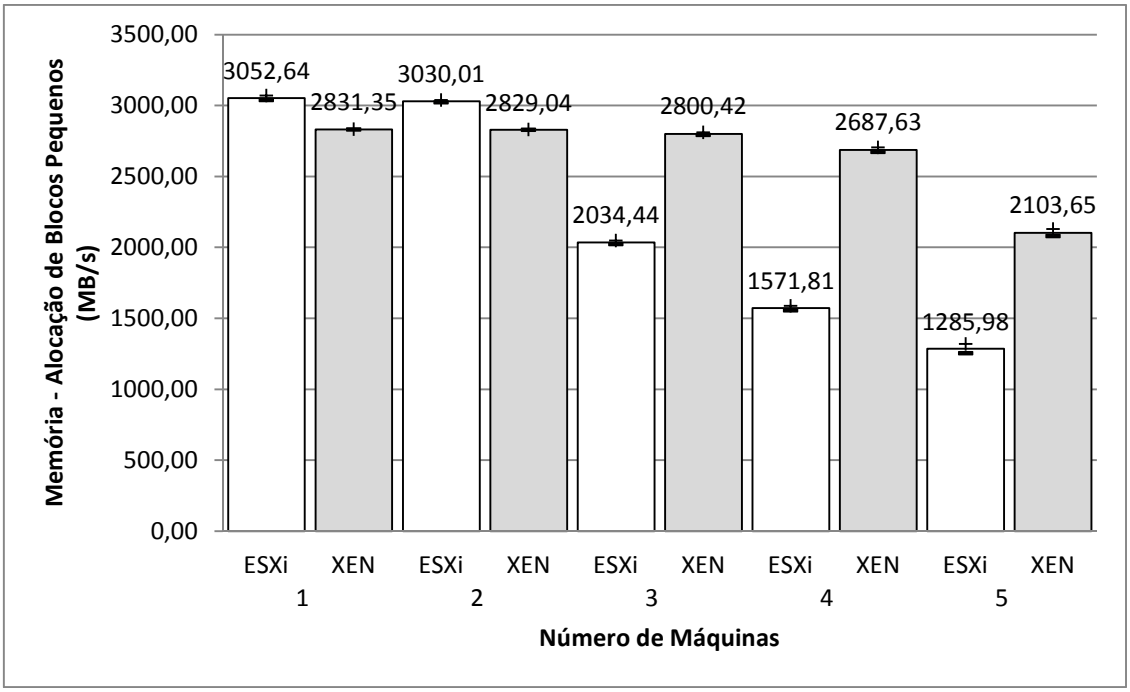


Figura 5.9 - Teste de memória: alocação de blocos pequenos (Mbytes /s) no Windows 2003 utilizando *PassMark Performance Test*.

A Figura 5.9 representa os valores médios obtidos no teste de desempenho em operações de alocação e liberação de blocos pequenos (100KB de tamanho), numa taxa de Mbytes por segundo. Os valores médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%. Pode ser observado na Tabela 5.9.

Tabela 5.9 – Teste de memória: resultados médios do *PassMark* para operações de alocação de blocos de 100KB numa taxa de MB/s

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	3052,64	16,04
	XEN	2831,35	1,62
2	ESXi	3030,01	7,60
	XEN	2829,04	2,09
3	ESXi	2034,44	12,40
	XEN	2800,42	9,07
4	ESXi	1571,81	15,59
	XEN	2687,63	16,47
5	ESXi	1285,98	32,54
	XEN	2103,65	25,54

Esses resultados denotam diferença de comportamento no desempenho relativo a concorrência de recursos, onde a degradação gerada pela concorrência praticamente não

existe entre as VMs hóspedes do XenServer, com destaque para o momento em que os recursos de memória (3GB) se esgotam, gerando concorrência de recursos de máquinas executando simultaneamente, gerando uma nova taxa de degradação na passagem de duas para três máquinas. O *VMware* inicialmente possui um melhor desempenho, porém com a degradação de desempenho gerada pela concorrência de recursos, o XenServer passa a ter melhor desempenho mantendo-se praticamente constante, porém fazendo com que o recurso de memória se torne um limitador crítico da VM.

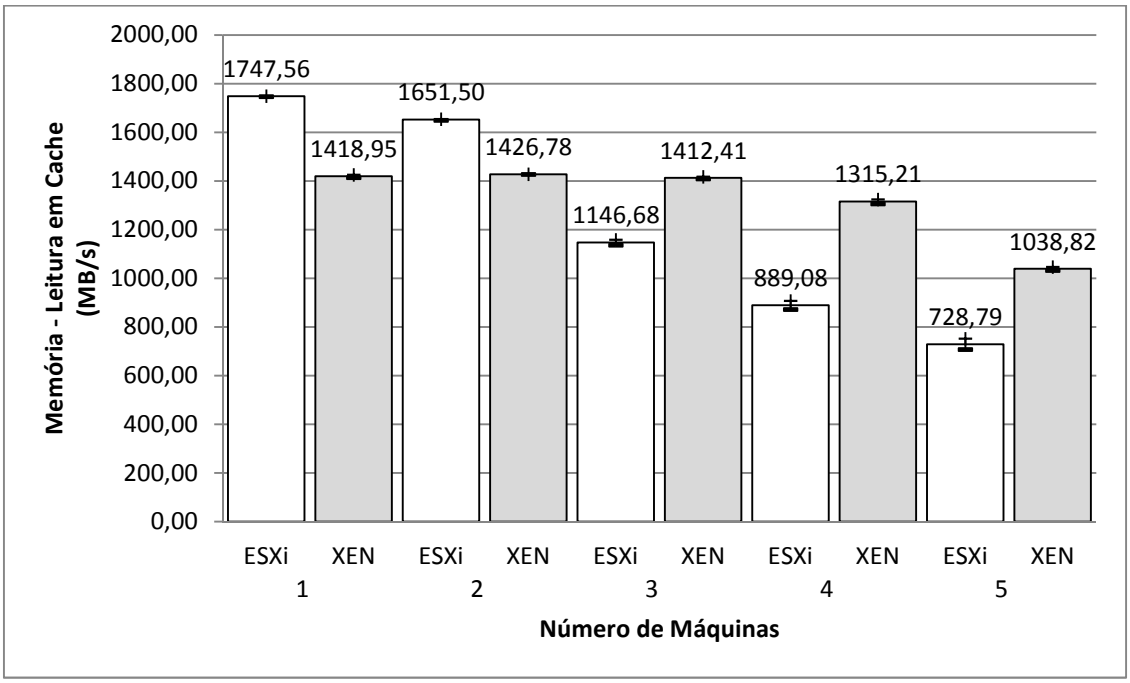


Figura 5.10 - Teste de memória: leitura em cache (Mbytes/s) no Windows 2003 utilizando *PassMark Performance Test*.

A Figura 5.10 representa os valores médios obtidos no teste de desempenho em operações de leitura de blocos de memória em cache, numa taxa de Mbytes por segundo. Os valores obtidos médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, pode ser observado na Tabela 5.10.

Tabela 5.10 – Teste de memória: resultados médios do *PassMark* para operações de alocação de blocos de 100KB numa taxa de MB/s

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	1747,56	1,25
	XEN	1418,95	5,99
2	ESXi	1651,50	2,20
	XEN	1426,78	0,68
3	ESXi	1146,68	11,58
	XEN	1412,41	4,51
4	ESXi	889,08	18,25
	XEN	1315,21	9,29
5	ESXi	728,79	22,20
	XEN	1038,82	7,27

Assim como no teste anterior, observa-se a mudança no desempenho ao se esgotar os recursos físicos reais e o escalonamento dos recursos virtualizados. O *VMware* se destaca com melhor desempenho incialmente, e depois o XenServer o supera com o acréscimo de máquinas.

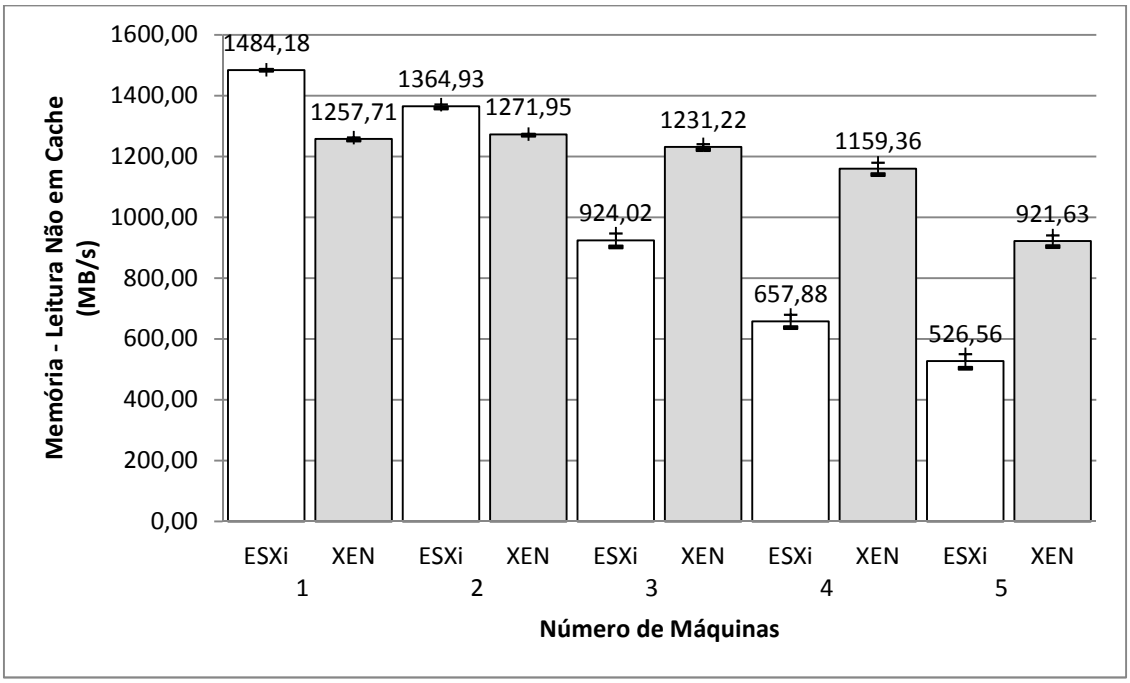


Figura 5.11 - Teste de memória: leitura não em cache (Mbytes por segundo) no Windows 2003 utilizando *PassMark Performance Test.*

A Figura 5.11 representa os valores médios obtidos no teste de desempenho em operações de leitura de memória não em *cache*, sendo o bloco grande o bastante para não estar armazenado em *cache*, numa taxa de Mbytes por segundo. Os valores médios obtidos

com o *PassMark* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, pode ser observado na Tabela 5.11.

Tabela 5.11 – Teste de memória: resultados médios do *PassMark* para operações de leitura de memória não em cache (MB/s)

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	1484,18	1,30
	XEN	1257,71	4,33
2	ESXi	1364,93	5,74
	XEN	1271,95	2,20
3	ESXi	924,02	22,22
	XEN	1231,22	8,65
4	ESXi	657,88	21,14
	XEN	1159,36	19,75
5	ESXi	526,56	23,05
	XEN	921,63	18,58

Assim como no teste anterior, observa-se a mudança no desempenho ao se esgotar os recursos físicos reais e o escalonamento dos recursos virtualizados, além da inversão na liderança do desempenho entre o *VMware* e o XenServer.

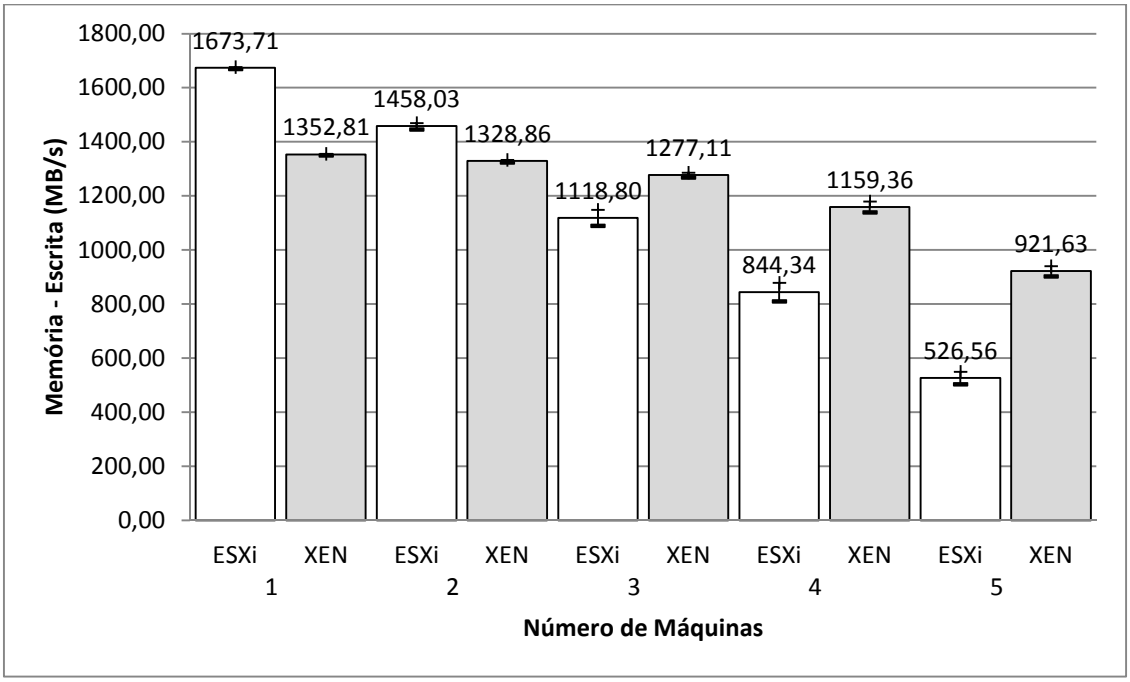


Figura 5.12 - Teste de memória: operação de escrita (Mbytes por segundo) no Windows 2003 utilizando *PassMark Performance Test.*

A Figura 5.12 representa os valores médios obtidos no teste de desempenho em operações de escrita em memória, numa taxa de Mbytes por segundo. Os valores médios

obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%., podendo ser observado na Tabela 5.12.

Tabela 5.12 – Teste de memória: resultados médios do *PassMark* para operações de escrita em memória (MB/s)

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	1673,71	3,53
	XEN	1352,81	3,28
2	ESXi	1458,03	11,82
	XEN	1328,86	4,51
3	ESXi	1118,80	29,60
	XEN	1277,11	8,83
4	ESXi	844,34	34,36
	XEN	1159,36	19,75
5	ESXi	526,56	23,05
	XEN	921,63	18,58

Nesse teste, assim como nos anteriores o *VMware* se destacou como o melhor inicialmente em desempenho, porem se mostrou que com o acréscimo de novas máquinas, tal quadro se inverteu, tornando o XenServer uma melhor opção no momento em que os recursos físicos de esgotam.

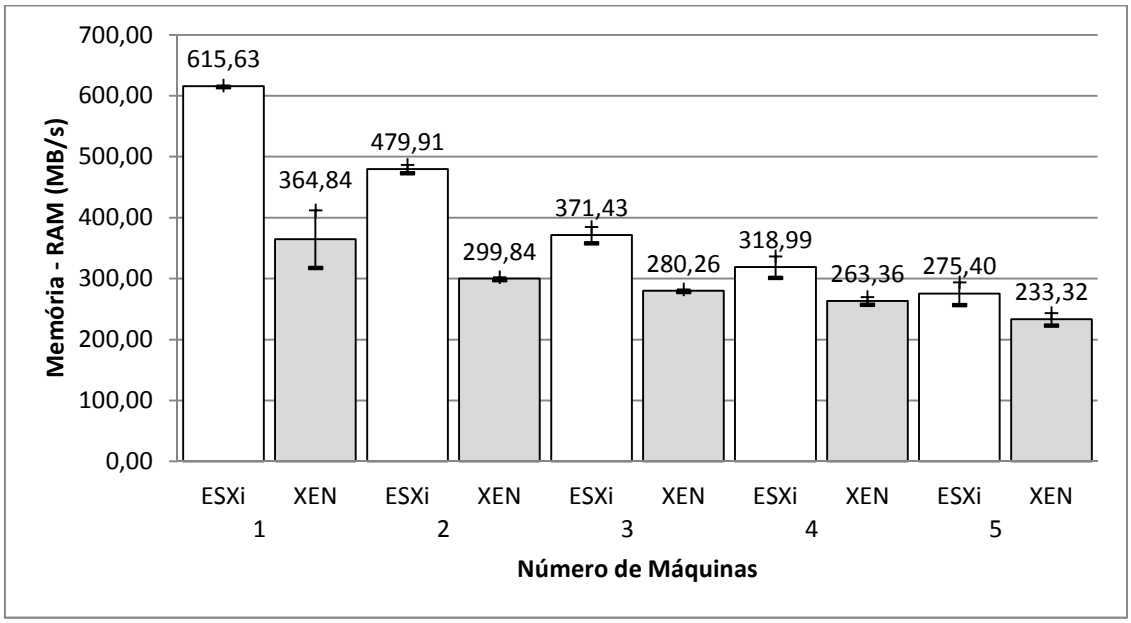


Figura 5.13 - Teste de memória: número de operações em memória RAM numa taxa de MB/s no Windows 2003 utilizando *PassMark Performance Test*.

A Figura 5.13 representa os valores médios obtidos no teste de desempenho alocação de grandes blocos de memória e seu tempo de leitura para aplicações que

consomem muita memória RAM, numa taxa de operações por segundo. Os valores médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, pode ser observado na Tabela 5.13.

Tabela 5.13 – Teste de memória: resultados médios do *PassMark* para operações em memória RAM (MB/s)

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	615,63	1,39
	XEN	364,84	47,17
2	ESXi	479,91	6,78
	XEN	299,84	2,34
3	ESXi	371,43	13,41
	XEN	280,26	2,19
4	ESXi	318,99	17,47
	XEN	263,36	6,29
5	ESXi	275,40	18,76
	XEN	233,32	10,33

Nesse teste diferente dos anteriores o *VMware* se manteve com melhor desempenho durante todas as fases do teste.

Nessa categoria foi confirmado que o gerenciamento de memória do VMM do *VMware* e do XenServer têm comportamentos diferentes, e que dependendo da quantidade de memória física, da quantidade de memória mínima alocada para cada VM e o número de VMs em execução, uma opção pode ser melhor que a outra. Assim, para a configuração em questão, o XenServer se mostrou como melhor opção na maioria dos testes realizados.

Os recursos de disco são trabalhados com a política de reserva de recurso nos dois VMMs. Existem opções de configurações de otimizações de seu uso, porém esse recurso funciona mais como um limitador, pois não é possível alocar mais espaço em disco do que disponível fisicamente. Dessa forma, com a configuração escolhida para esse trabalho, o PassMark retornou os seguintes resultados para os testes de disco:

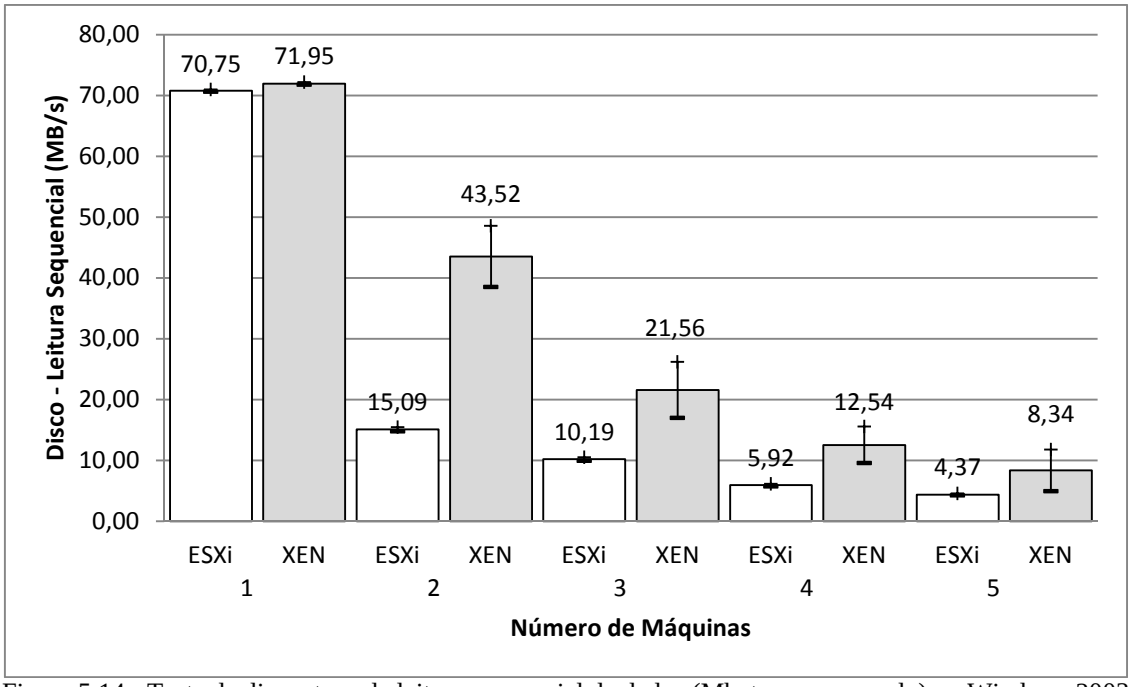


Figura 5.14 - Teste de disco: taxa de leitura sequencial de dados (Mbytes por segundo) no Windows 2003 utilizando *PassMark Performance Test*.

A Figura 5.14 representa os valores médios obtidos no teste de desempenho em leitura sequencial em disco, numa taxa de Mbytes por segundo. Os valores médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, pode ser observado na Tabela 5.14.

Tabela 5.14 – Teste de disco: resultados médios do *PassMark* para operações de leitura sequencial em disco (MB/s)

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	70,75	0,17
	XEN	71,95	0,22
2	ESXi	15,09	0,34
	XEN	43,52	5,03
3	ESXi	10,19	0,27
	XEN	21,56	4,59
4	ESXi	5,92	0,19
	XEN	12,54	3,00
5	ESXi	4,37	0,16
	XEN	8,34	3,41

Nesse teste denota-se uma elevada queda no desempenho gerada pela concorrência do recurso de acesso ao disco, sendo muito mais acentuada no caso do *VMware* e consequentemente, melhor desempenho do XenServer.

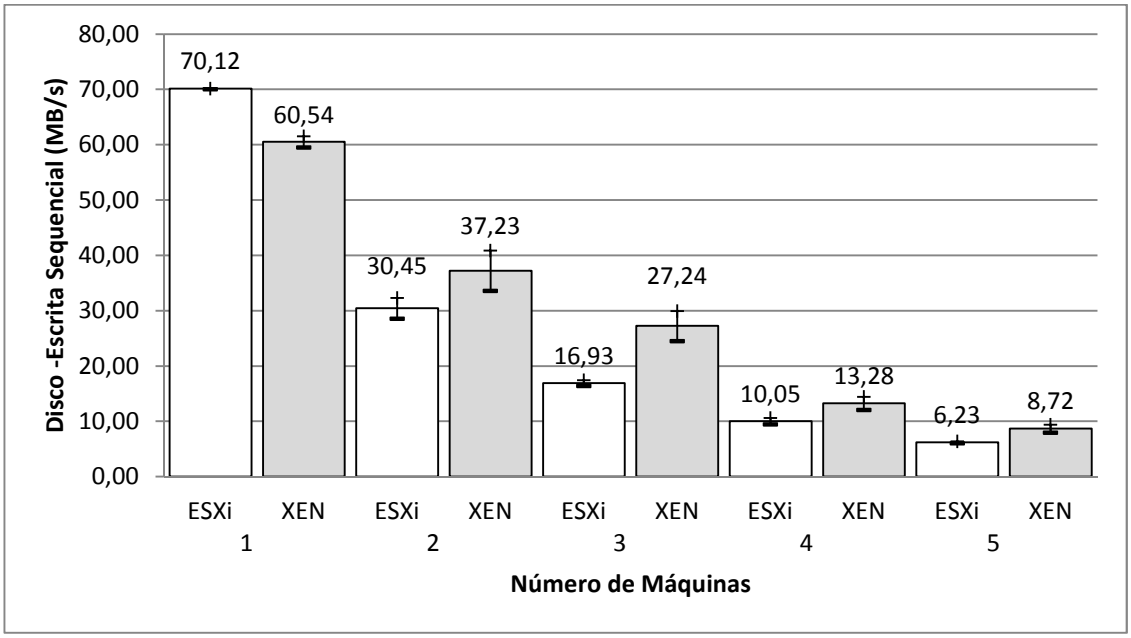


Figura 5.15 - Teste de disco: taxa de escrita sequencial de dados (Mbytes por segundo) no Windows 2003 utilizando *PassMark Performance Test*.

A Figura 5.15 representa os valores médios obtidos no teste de desempenho em escrita sequencial em disco, numa taxa de Mbytes por segundo. Os valores médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%. Pode ser observado na Tabela 5.15.

Tabela 5.15 – Teste de disco: resultados médios do *PassMark* para operações escrita sequencial em disco (MB/s)

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	70,12	0,07
	XEN	60,54	1,02
2	ESXi	30,45	1,87
	XEN	37,23	3,65
3	ESXi	16,93	0,52
	XEN	27,24	2,71
4	ESXi	10,05	0,59
	XEN	13,28	1,19
5	ESXi	6,23	0,18
	XEN	8,72	0,73

Assim como no teste anterior, denota-se uma elevada queda no desempenho gerada pela concorrência do recurso de acesso ao disco, sendo o *VMware* com melhor desempenho inicialmente e depois o XenServer torna-se a melhor opção.

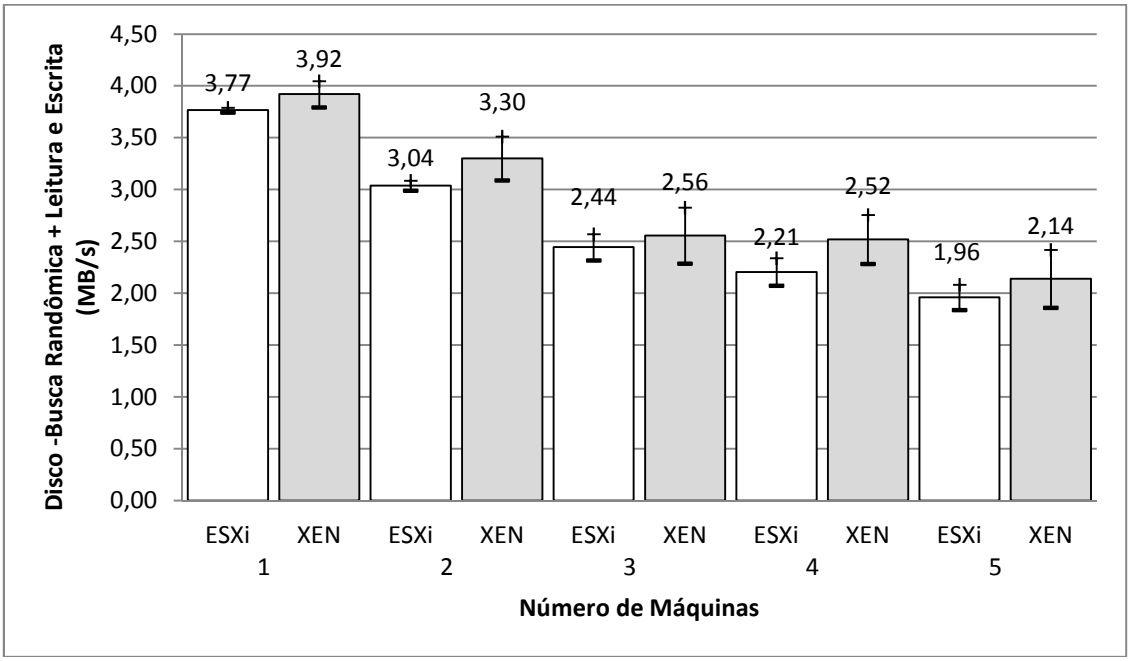


Figura 5.16 - Teste de disco: desempenho em operações de busca randômica, leitura e escrita (Mbytes por segundo) no Windows 2003 utilizando *PassMark Performance Test*.

A Figura 5.16 representa os valores médios obtidos no teste de desempenho busca randômica com leitura e escrita em disco, numa taxa de Mbytes por segundo. Os valores médios obtidos com o *PassMark* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, pode ser observado na Tabela 5.16.

Tabela 5.16 – Teste de disco: resultados médios do *PassMark* para operações busca randômica, leitura e escrita em disco (MB/s)

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	3,77	0,02
	XEN	3,92	0,13
2	ESXi	3,04	0,05
	XEN	3,30	0,21
3	ESXi	2,44	0,13
	XEN	2,56	0,27
4	ESXi	2,21	0,13
	XEN	2,52	0,24
5	ESXi	1,96	0,12
	XEN	2,14	0,28

Esses resultados denotam a evolução padrão da degradação em relação ao número de máquinas, apesar do XenServer possuir valores médios absolutos superiores, nada se pode afirmar, devido a interseção entre os intervalos de confiança apresentados em comparação com o *VMware*.

Nessa categoria o XenServer se mostrou como a melhor opção de VMMs, por possuir um melhor desempenho para múltiplas máquinas virtualizadas simultâneas, no caso de leitura sequencial em disco. O *VMware* obteve um melhor desempenho, somente quando a VM se encontrava rodando sozinha sem a sobrecarga gerada pela competição de recursos no caso escrita .

5.2 Sistema Operacional Hóspede Fedora 14

Nessa seção são apresentados os dados obtidos nos testes com o *hypervisor VMware* ESXi 4.1.0 e o Citrix XenServer 5.6.2, após os resultados gerados pela análise das ferramentas de *benchmark* escolhidas, com o sistema operacional hóspede Fedora 14.

5.2.1 Resultados do *Phoronix*

O Phoronix retornou os seguintes resultados para os testes de processador após a execução simultânea das VMs com teste “compress-gzip”:

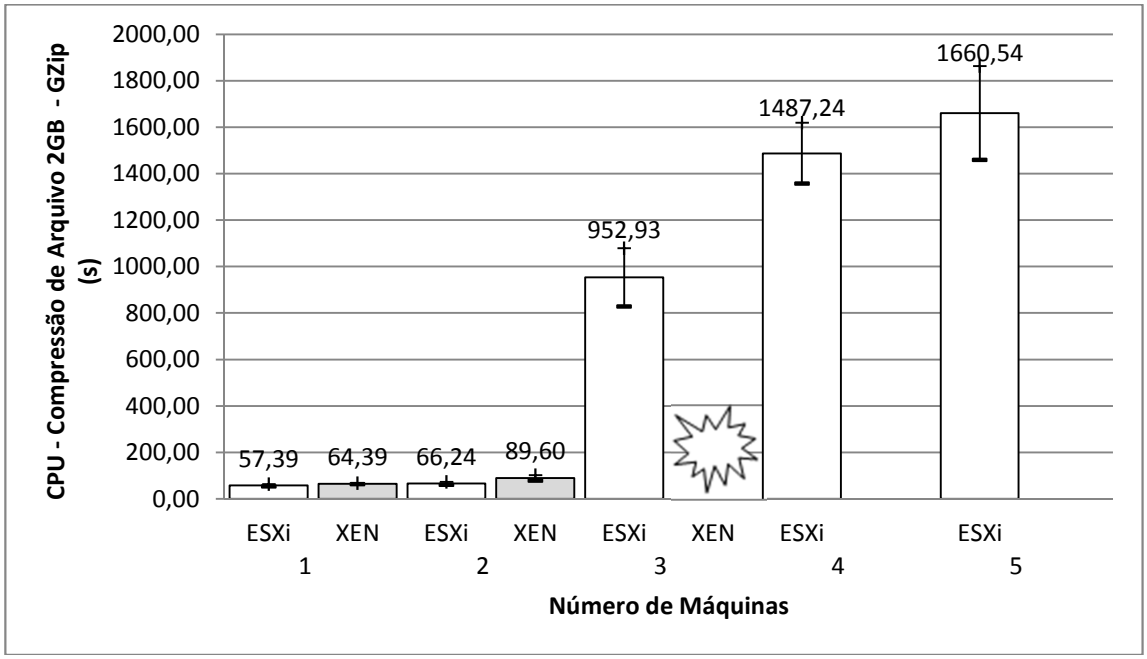


Figura 5.17 - Teste de processador: tempo (segundos) gasto para compactar um arquivo de 2GB, utilizando o algoritmo GZip no Fedora 14 com *Phoronix Test Suite*.

A Figura 5.17 representa os valores médios em segundos gastos pelo processador, obtidos com o teste de compressão de um arquivo de 2Gbytes com o algoritmo GZip. Os valores médios obtidos com o *Phoronix* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.17.

Tabela 5.17 – Teste de processador: resultados médios do *Phoronix* para o tempo gasto por operações de compressão para arquivo de 2GB (s)

# Máquinas	Hypervisor	Média (s)	Erro (±)
1	ESXi	57,39	4,91
	XEN	64,39	1,99
2	ESXi	66,24	5,82
	XEN	89,60	11,31
3	ESXi	952,93	125,34
	XEN	instabilidade	---
4	ESXi	1487,24	131,10
	XEN	---	---
5	ESXi	1660,54	201,83
	XEN	---	---

Esses resultados denotam uma semelhança muito grande nas taxas iniciais de desempenho para os dois VMMs, sendo o ESXi ligeiramente melhor. Porém, quando surgiu a necessidade de competição por recurso de processador, no XenServer, durante o teste com três máquinas simultâneas, a terceira VM ficou instável e passou a reiniciar o sistema operacional hóspede durante o teste, não sendo possível concluí-lo. Isso não aconteceu com o VMM da *VMware*, seguindo os testes até o fim e se mostrando mais estável que seu concorrente. Outro ponto negativo para o XenServer foi devido ao seu modo de gerenciamento de memória, não foi possível realizar testes com quatro ou cinco máquinas simultâneas, pois o sistema hóspede não conseguia completar a inicialização por falta de recurso disponível de memória. Desta forma, impossibilitando aplicar os testes para esse sistema hóspede com quatro ou cinco máquinas simultâneas no XenServer. O mesmo não aconteceu com o VMM da *VMware*. Deixando esse ponto de lado, pode-se notar uma grande perda de desempenho ao se iniciar a concorrência de recurso, na passagem de duas para três máquinas simultâneas, no caso do *VMware*.

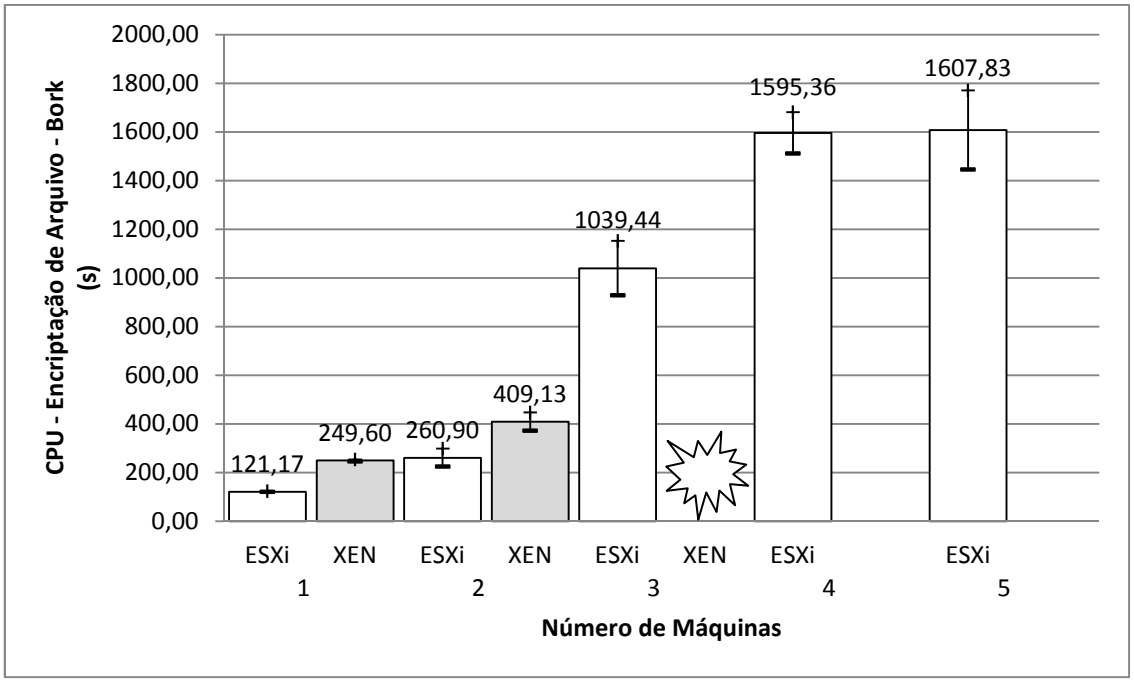


Figura 5.18 - Teste de processador: tempo (em segundos) gasto para encriptar um arquivo no Fedora 14 com *Phoronix Test Suite*.

A Figura 5.18 representa os valores médios em segundos gastos pelo processador, obtidos com o teste "bork" de encriptação de um arquivo. Os valores médios obtidos com o *Phoronix* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, pode ser observado na Tabela 5.18.

Tabela 5.18 – Teste de processador: resultados médios do *Phoronix* para o tempo gasto por operações encriptação de arquivos (s)

# Máquinas	Hypervisor	Média (s)	Erro (±)
1	ESXi	121,17	1,69
	XEN	249,60	3,85
2	ESXi	260,90	36,82
	XEN	409,13	37,10
3	ESXi	1039,44	111,66
	XEN	instabilidade	---
4	ESXi	1595,36	84,99
	XEN	---	---
5	ESXi	1607,83	162,57
	XEN	---	---

Assim como no teste anterior, esses resultados denotam o melhor desempenho do VMM da *VMware*, além de uma maior estabilidade e flexibilidade, apesar da perda de desempenho.

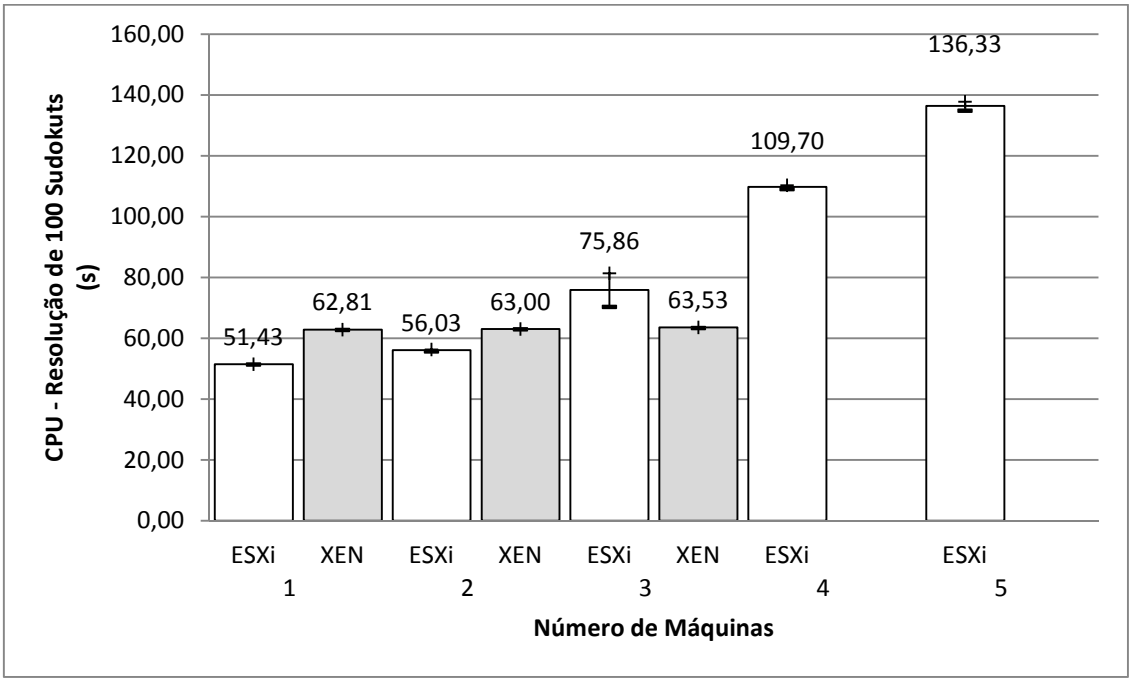


Figura 5.19 - Teste de processador: tempo (em segundos) gasto pela resolução de 100 sudokuts no Fedora 14 com Phoronix Test Suite.

A Figura 5.19 representa os valores médios em segundos gastos pelo processador, obtidos com o teste "sudokut" de resolução de 100 quebra-cabeças de Sudokut. Os valores médios obtidos com o *Phoronix* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.19.

Tabela 5.19 – Teste de processador: resultados médios do *Phoronix* para o tempo gasto para resolução de 100 problemas sudokut (s)

# Máquinas	Hypervisor	Média (s)	Erro (±)
1	ESXi	51,43	0,04
	XEN	62,81	0,06
2	ESXi	56,03	0,36
	XEN	63,00	0,08
3	ESXi	75,86	5,48
	XEN	63,53	0,11
4	ESXi	109,70	0,62
	XEN	---	---
5	ESXi	136,33	1,52
	XEN	---	---

Assim como no teste anterior, o *VMware* obteve melhor desempenho inicial, porém nesse teste, perdeu sua liderança quando houve a concorrência gerada por três máquinas. Porém se pôde concluir se isso se denotaria com mais veemência com um aumento no número de máquinas, por motivos citados anteriormente.

Desta forma, na categoria desempenho de processador em sistema hóspede Fedora 14, o *VMware* se mostrou como melhor opção, por estabilidade e flexibilidade de uso de recursos apesar da perda de desempenho de uma forma geral.

Para os testes de memória foi elencado teste "*ramspeed*", que realiza vários testes diferenciados e um resultado consolidado das análises. Porém para análise e avaliação, foi considerado apenas o resultado consolidado para os dois VMMs. Assim, o *Phoronix* retornou os seguintes resultados para os testes de memória:

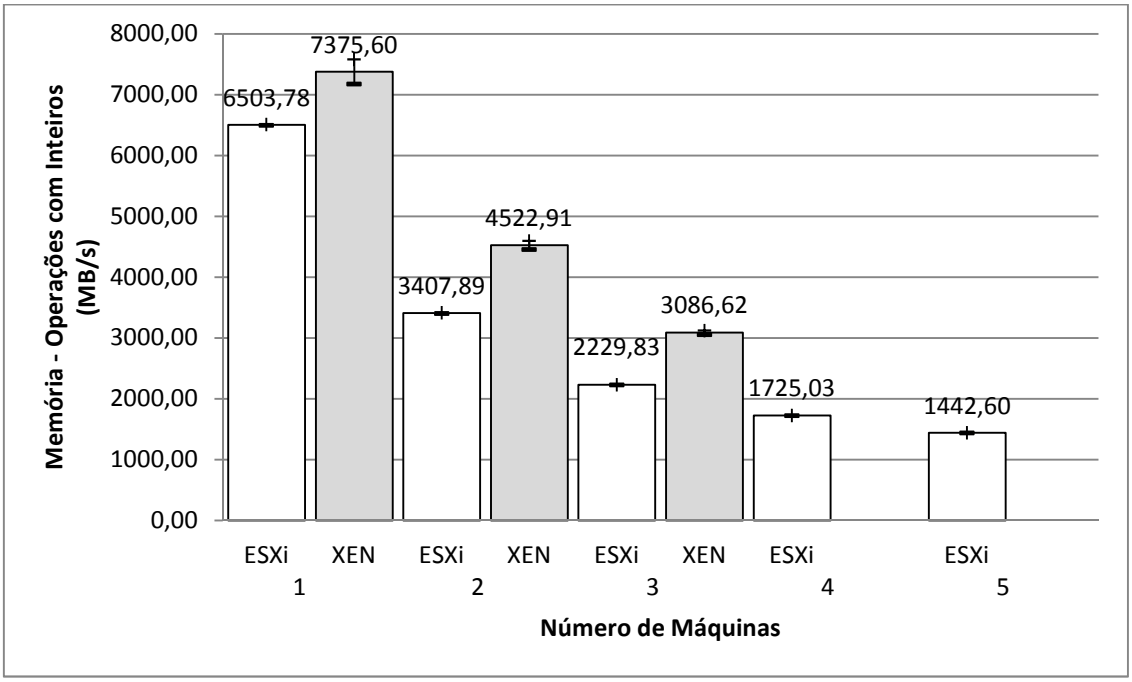


Figura 5.20 - Teste de memória: operações com inteiros (MBytes por segundo) no Fedora 14 com *Phoronix Test Suite*.

A Figura 5.20 representa os valores médios em Mbytes por segundo, gasto em operações de memória com inteiros, fornecidos pelo teste "ramspeed". Os valores médios obtidos com o *Phoronix* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.20.

Tabela 5.20 – Teste de memória: resultados médios do *Phoronix* para taxas de operações com inteiros (MB/s)

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	6503,78	6,70
	XEN	7375,60	203,44
2	ESXi	3407,89	7,31
	XEN	4522,91	70,06
3	ESXi	2229,83	3,99
	XEN	3086,62	32,89
4	ESXi	1725,03	3,64
	XEN	---	---
5	ESXi	1442,60	2,12
	XEN	---	---

Esse teste denota um melhor desempenho do XenServer, em todas as fases do teste onde foi possível sua execução.

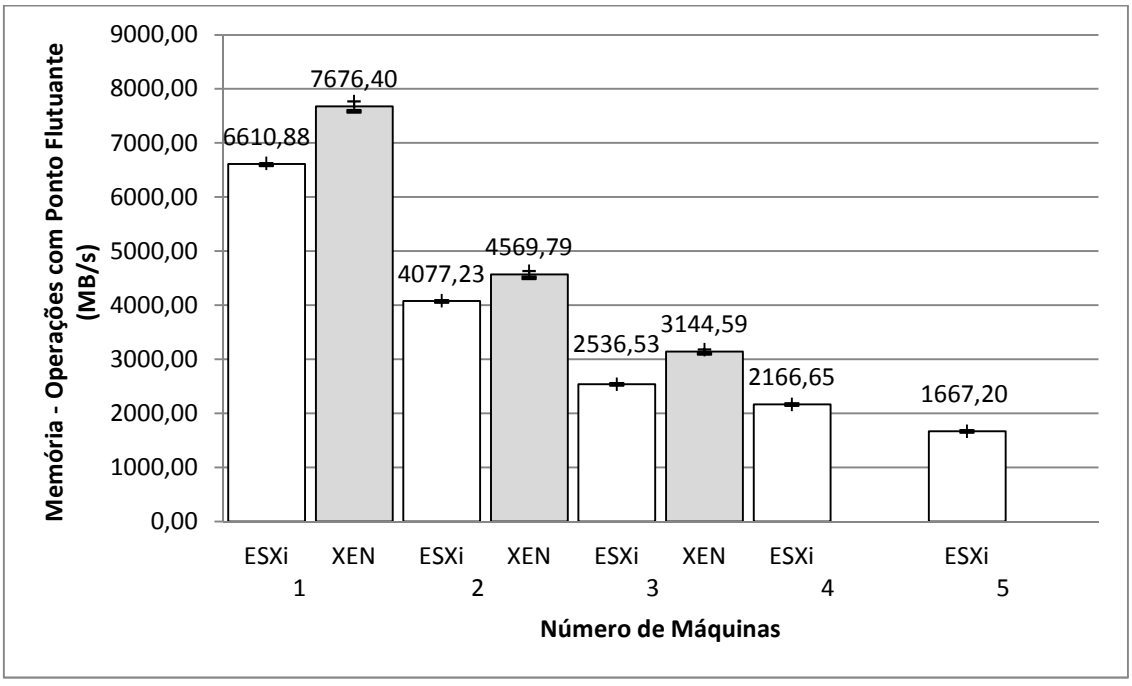


Figura 5.21 - Teste de memória: operações com ponto flutuante (MBytes por segundo) no Fedora 14 com *Phoronix Test Suite*.

A Figura 5.21 representa os valores médios em Mbytes por segundo, gasto em operações de memória com ponto flutuante, fornecidos pelo teste “ramspeed”. Os valores obtidos com o *Phoronix* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.21.

Tabela 5.21 – Teste de memória: resultados médios do *Phoronix* para taxas de operações com ponto flutuante (MB/s)

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	6610,88	16,28
	XEN	7676,40	89,85
2	ESXi	4077,23	8,87
	XEN	4569,79	61,30
3	ESXi	2536,53	3,69
	XEN	3144,59	37,25
4	ESXi	2166,65	3,66
	XEN	---	---
5	ESXi	1667,20	2,11
	XEN	---	---

Esse teste, assim como anterior, denota um melhor desempenho do XenServer, em todas as fases do teste onde foi possível sua execução.

Desta forma, podemos avaliar que o XenServer tem melhor desempenho nas operações de memória com o sistema operacional hóspede Fedora 14.

O Phoronix retornou os seguintes resultados para os testes de disco:

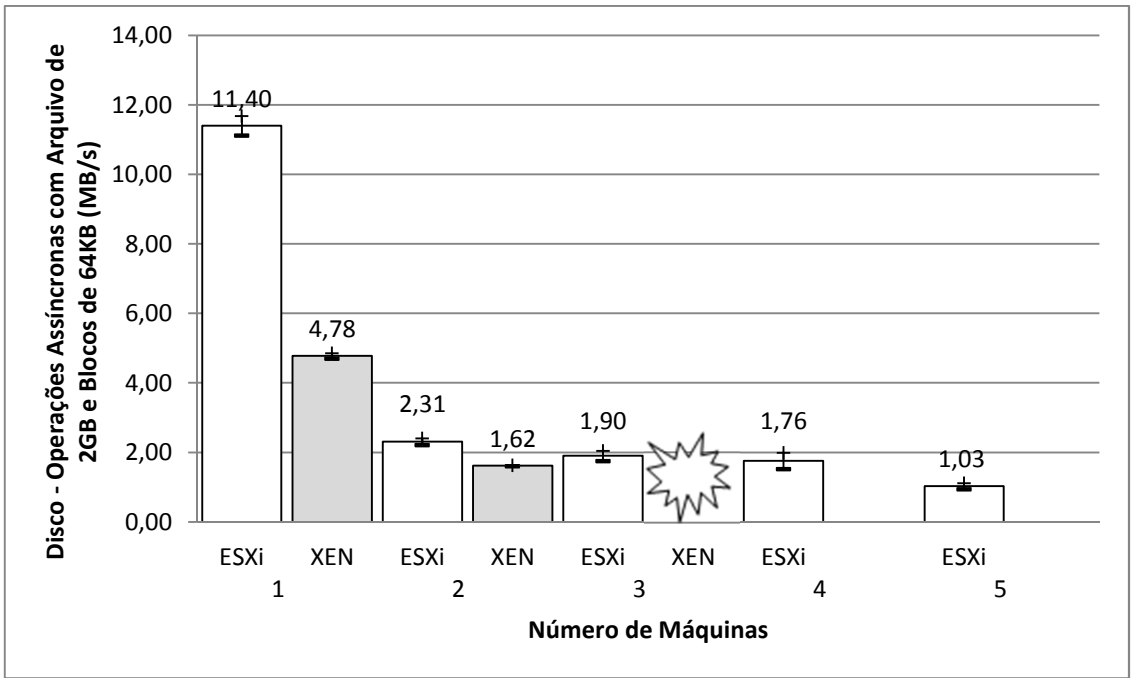


Figura 5.22 - Teste de disco: operações assíncronas (MBytes /s) no Fedora 14 com *Phoronix Test Suite*.

A Figura 5.22 representa os valores médios em Mbytes por segundo, gasto em operações de disco assíncrona, utilizando um arquivo de 2GB e 64KB de blocos, fornecidos pelo teste “aio-stress”. Os valores obtidos com o *Phoronix* nessa categoria e

suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.22.

Tabela 5.22 – Teste de disco: resultados médios do *Phoronix* para taxas de operações de disco assíncrona (MB/s)

# Máquinas	Hypervisor	Média (MB/s)	Erro (±)
1	ESXi	11,40	0,28
	XEN	4,78	0,08
2	ESXi	2,31	0,10
	XEN	1,62	0,01
3	ESXi	1,90	0,15
	XEN	instabilidade	---
4	ESXi	1,76	0,23
	XEN	---	---
5	ESXi	1,03	0,09
	XEN	---	---

Esse teste denota a significante queda no desempenho para operações de disco, logo na primeira situação de concorrência de recursos, além da predominância de desempenho do *VMware*.

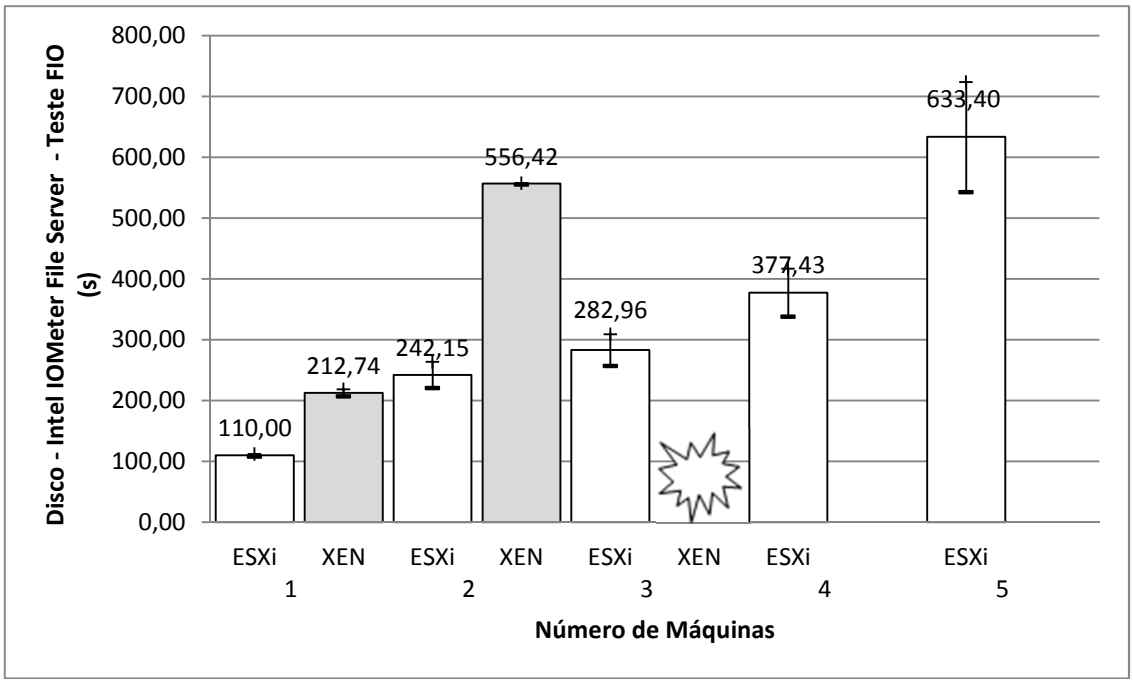


Figura 5.23 - Teste de disco: *Intel IOMeter File Server Access Pattern* (segundos) no Fedora 14 com *Phoronix Test Suite*.

A Figura 5.23 representa os valores médios em segundos, do tempo gasto em operações de disco com o teste “fio”, em específico o teste *Intel IOMeter File Server Access Pattern*. Os valores obtidos com o *Phoronix* nessa categoria e suas respectivas precisões para uma, duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.23.

Tabela 5.23 – Teste de disco: resultados médios do *Phoronix* para o tempo gasto com operações realizadas pelo teste *Intel IOMeter File Server Access Pattern* (s)

# Máquinas	Hypervisor	Média (s)	Erro (±)
1	ESXi	110,00	1,99
	XEN	212,74	5,75
2	ESXi	242,15	21,69
	XEN	556,42	1,26
3	ESXi	282,96	25,94
	XEN	instabilidade	---
4	ESXi	377,43	39,66
	XEN	---	---
5	ESXi	633,40	90,57
	XEN	---	---

Esse teste denota o melhor desempenho do *VMware* em todas as fases do teste.

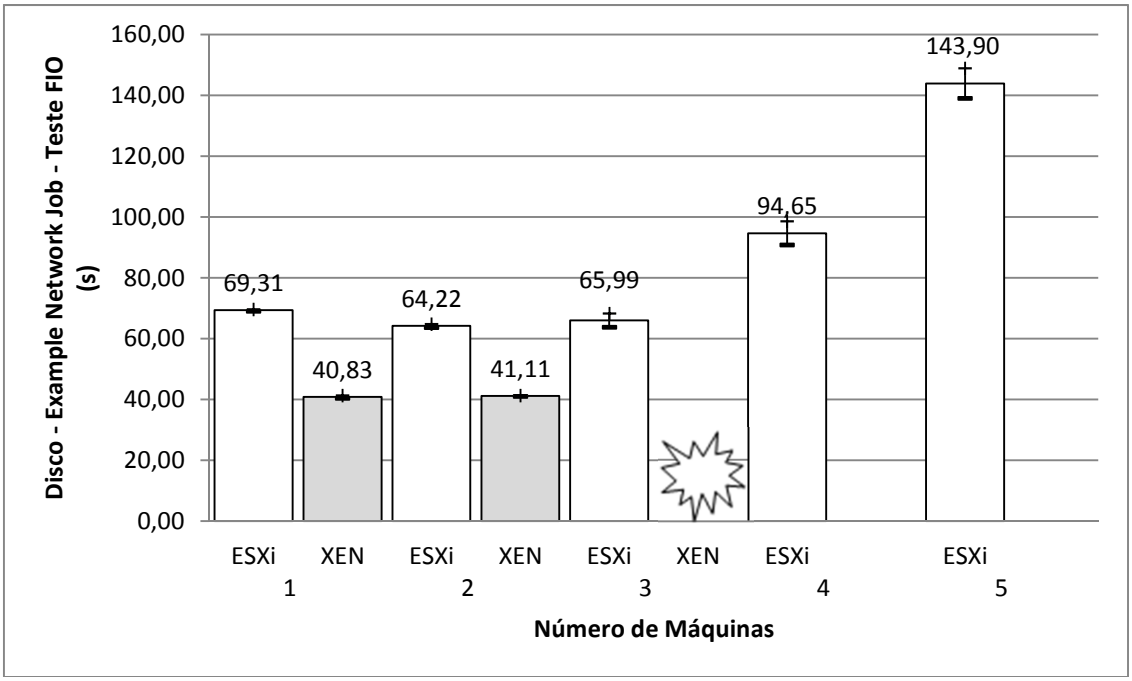


Figura 5.24 - Teste de disco: *Example Network Job* (segundos) no Fedora 14 com *Phoronix Test Suite*.

A Figura 5.24 representa os valores médios em segundos, do tempo gasto em operações de disco com o teste “fio”, em específico o teste *Example Network Job*. Os valores obtidos com o *Phoronix* nessa categoria e suas respectivas precisões para uma,

duas, três, quatro e cinco máquinas, executando simultaneamente, com intervalo de confiança de 95%, podem ser observados na Tabela 5.24.

Tabela 5.24– Teste de disco: resultados médios do *Phoronix* para o tempo gasto com operações realizadas pelo teste *Example Network Job* (s)

# Máquinas	Hypervisor	Média (s)	Erro (±)
1	ESXi	69,31	0,30
	XEN	40,83	0,44
2	ESXi	64,22	0,58
	XEN	41,11	0,17
3	ESXi	65,99	2,27
	XEN	instabilidade	---
4	ESXi	94,65	3,87
	XEN	---	---
5	ESXi	143,90	4,95
	XEN	---	---

Esse teste denota o melhor desempenho do XenServer em todas as fases passíveis de comparação do teste, porém devido a instabilidade ocorrida o *VMware* torna-se a melhor opção.

O próximo teste, chamado "*tiobench*", realiza várias operações de escrita e leitura em disco, através de threads de 32MB de tamanho, sendo calculado o desempenho do disco com o disparo de 4, 8, 16 e 32 *threads* simultâneas. Assim, para as operações de escrita em disco com 4, 8, 16 e 32 *threads*, foram obtidos os seguintes resultados, representados pelas Figuras 5.25, 5.26, 5.27 e 5.28, respectivamente.

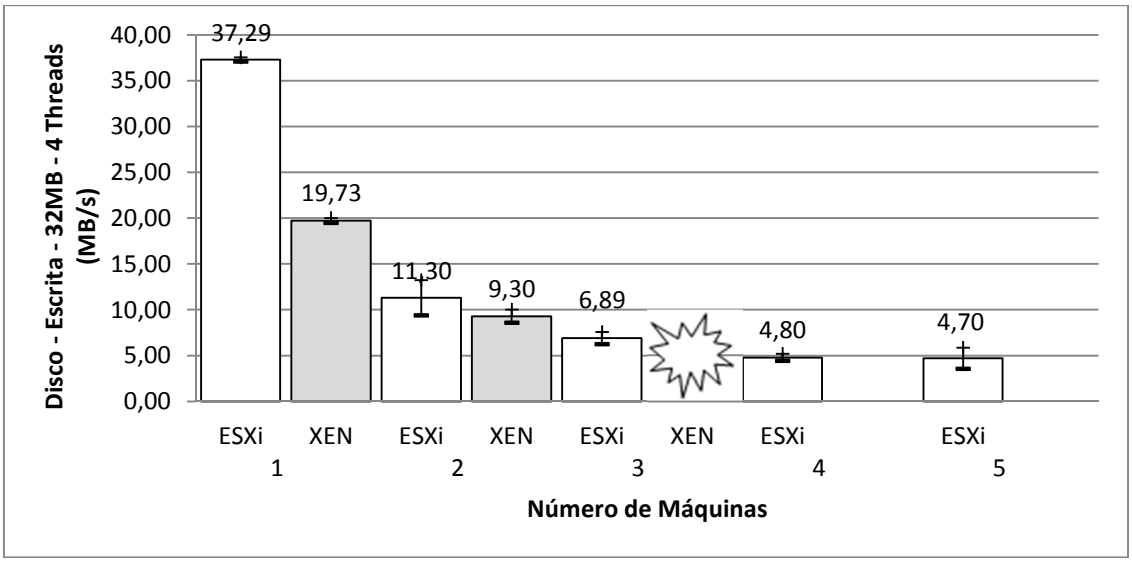


Figura 5.25 - Teste de disco: Escrita em disco com 4 *threads* de 32MB (MB/s).

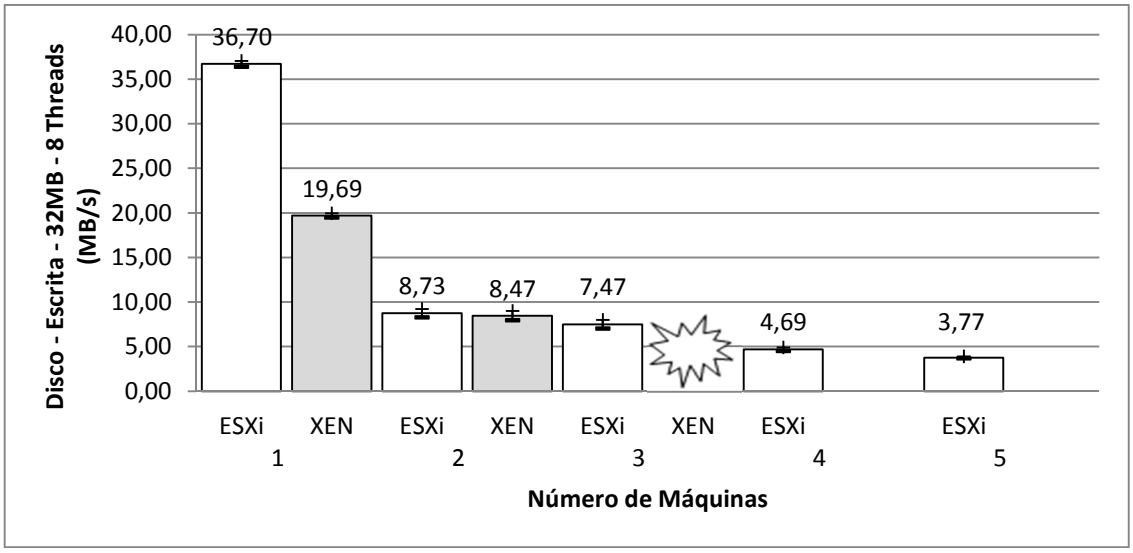


Figura 5.26 - Teste de disco: Escrita em disco com 8 *threads* de 32MB (MB/s).

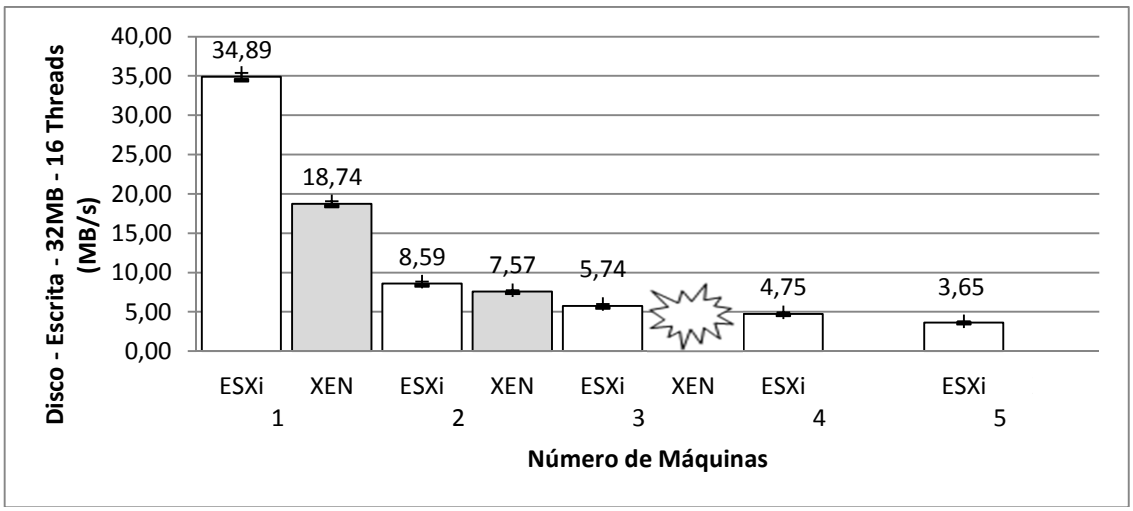


Figura 5.27 - Teste de disco: Escrita em disco com 16 *threads* de 32MB (MB/s).

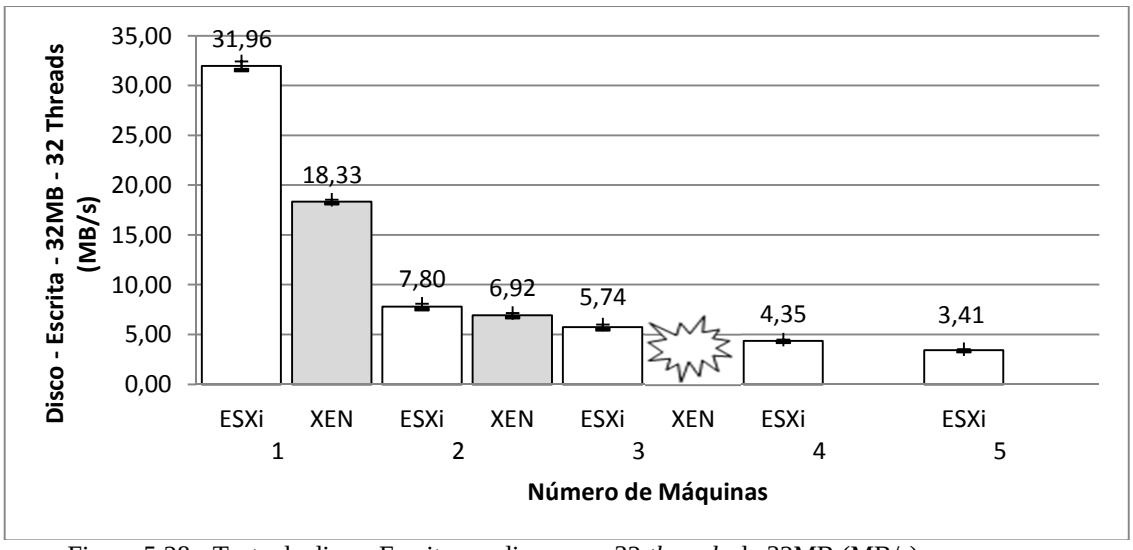


Figura 5.28 - Teste de disco: Escrita em disco com 32 *threads* de 32MB (MB/s).

Assim, pode-se observar que independente do número de *threads*, o desempenho da escrita em disco sempre é semelhante, não havendo variação significativa entre casos, além da predominância do desempenho do VMM do *VMware*.

Porém, para operações de leitura em disco com 4, 8, 16 e 32 *threads* de 32MB cada, foram obtidos os seguintes resultados, representados pelas Figuras 5.29, 5.30, 5.31 e 5.32, respectivamente:

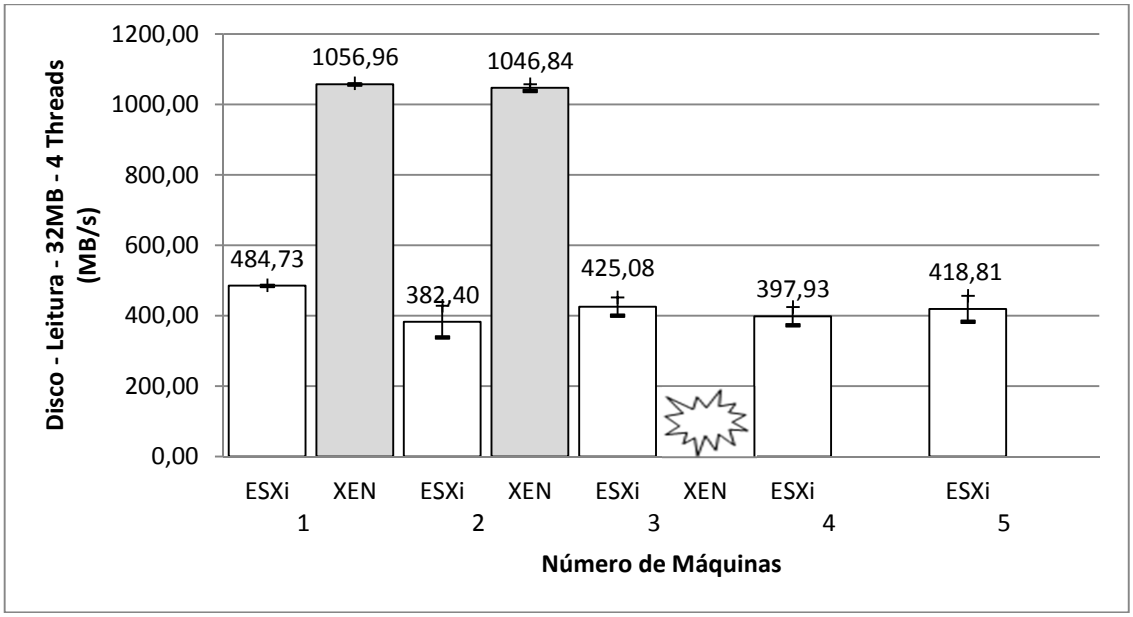


Figura 5.29 - Teste de disco: Leitura em disco com 4 *threads* de 32MB (MB/s).

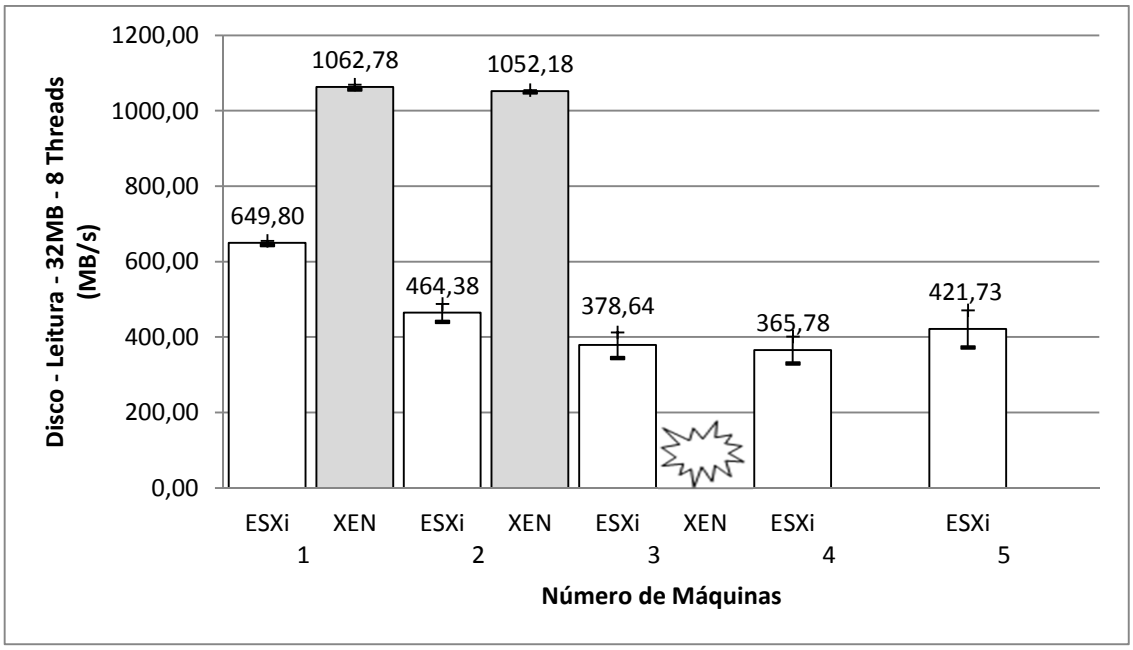


Figura 5.30 - Teste de disco: Leitura em disco com 8 *threads* de 32MB (MB/s).

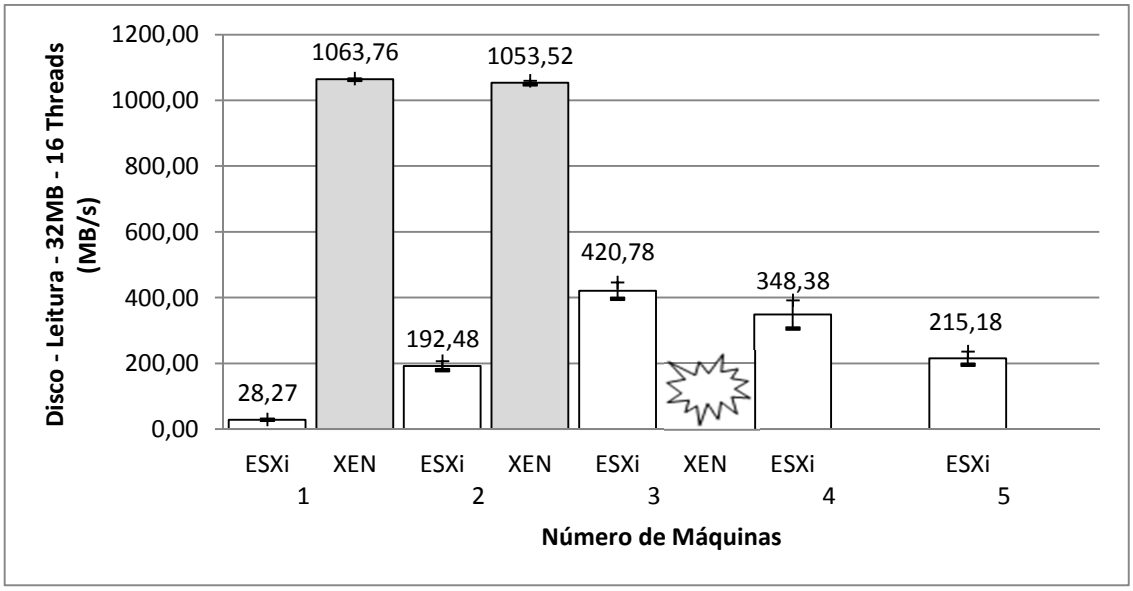


Figura 5.31 - Teste de disco: Leitura em disco com 16 *threads* de 32MB (MB/s).

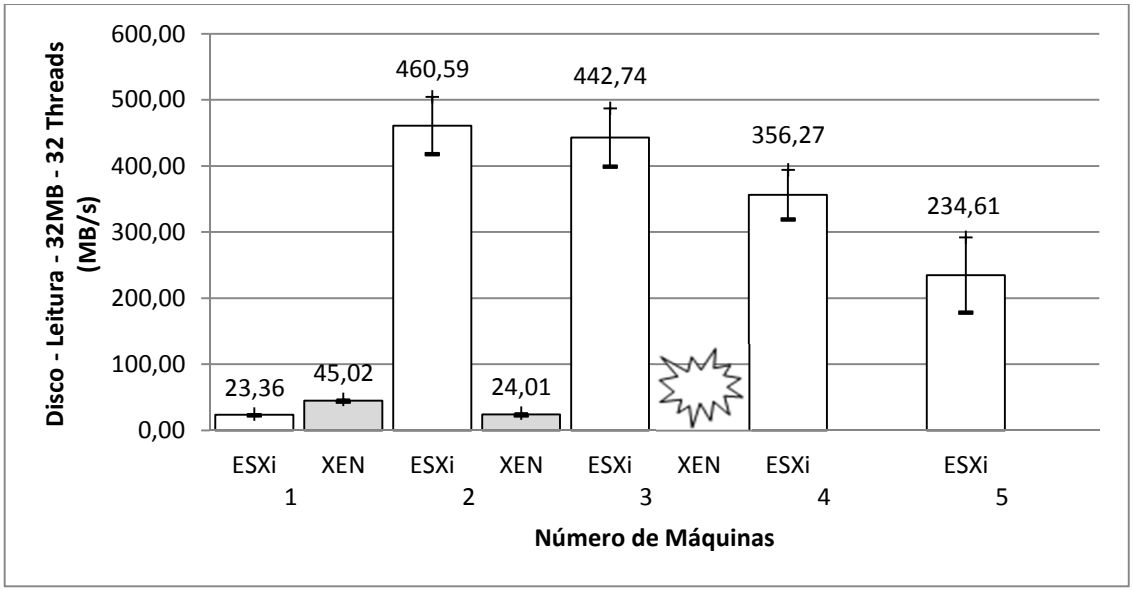


Figura 5.32 - Teste de disco: Leitura em disco com 32 *threads* de 32MB (MB/s).

Assim, pode-se observar que dependendo do número de *threads*, o desempenho da leitura em disco pode variar muito. Para leitura com a até 16 *threads*, a diferença de desempenho entre o *VMware* e o XenServer é muito acentuado, dando destaque para o desempenho do XenServer nesses casos. Tal cenário se inverte na análise feita para o caso de 32 threads, sendo o melhor desempenho do *VMware*.

De forma geral, o *VMware* obteve um melhor desempenho que o XenServer na maioria das categorias testadas, com exceção do quesito memória, além de uma maior robustez, nos testes com o sistema hóspede Fedora.

6 CONCLUSÕES E TRABALHOS FUTUROS

Com esse trabalho percebe-se que virtualização reduz a importância do sistema operacional, permitindo que um *hardware* execute quaisquer aplicações com seu sistema operacional de origem, sem precisar interromper as demais aplicações e serviços já em execução.

Nesse trabalho foram citadas características dos principais *softwares* de virtualização e baseados no cenário proposto foram analisados as performances de dois VMMs mais populares do mercado, o *VMware* ESXi e o XenServer.

De acordo com o cenário proposto, foram elencados também os sistemas operacionais hóspedes que se encontravam em maior número nessa organização, o Windows 2003 Server e o Fedora 14.

Um fator muito importante na escolha do *hypervisor* é o *hardware* da máquina física que deverá ser virtualizada. Para ser possível a virtualização, o processador deve possuir as extensões de virtualização como pré-requisito, porém esse não é um fator suficiente para o funcionamento dos *hypervisors*. O *VMware* possui uma lista de compatibilidade informada no site do fabricante. O XenServer não possui uma lista de compatibilidade, porém dependendo da versão ou do *hardware* da máquina (placa de rede, gerenciador de disco, CD-ROM, entre outros) pode haver problemas de *drivers*, que geralmente são resolvidos através de correções disponibilizadas pelo fabricante ou comunidade de desenvolvedores. No trabalho em questão, esse foi um fator determinante para a escolha das ferramentas de virtualização.

Superado essa primeira etapa, foram realizados estudos de desempenho e sobrecarga de serviços, em cada um dos sistemas operacionais elencados, com o intuito de verificar as deficiências e vantagens de cada VMM em cenários de grande concorrência de recursos e, desta forma, criar uma base para a escolha do VMM ideal para a criação de um ambiente virtualizado de testes.

Os testes aplicados mostraram técnicas de abordagem diferenciadas no gerenciamento de memória, que dependendo do caso, podem ser consideradas uma vantagem ou um limitador, como no caso do XenServer.

Com os testes também ficou claro que o desempenho de uma VM depende do sistema operacional hóspede escolhido e que sua evolução não é linear. O XenServer teve

melhor desempenho com o sistema hóspede Windows 2003 Server do que com o Fedora, gerando até instabilidade e redução da disponibilidade de máquinas virtualizadas. No caso do Fedora, o *hypervisor* mais adequado seria o *VMware*, pois comportou um número maior de máquinas virtualizadas que seu concorrente o XenServer, além de apresentar um melhor desempenho em operações de disco e uso de CPU nesse sistema operacional em particular.

Outro fator aparente nos testes é a robustez e o verdadeiro conceito de "virtualização de *hardware*" apresentado pelo *VMware* ESXi, pois este permitia a criação de máquinas utilizando *hardware* virtualizado, indiferente da disponibilidade do recurso em seu estado físico, porém em sacrifício do desempenho de suas outras máquinas virtuais.

Tendo em vista o objetivo deste trabalho, a criação de um ambiente de virtualização de testes, e melhor uso dos recursos computacionais da organização-alvo, foi indicado a solução de virtualização do *VMware* ESXi, devido priorização da robustez acima do desempenho, de se tratar de uma solução mais adequada para virtualização ambientes onde se pretende virtualizar uma quantidade considerável de máquinas com sistemas hóspedes heterogêneos, sendo esta a escolha mais adequada.

Em contra partida, observou-se que para uma solução de virtualização de pequeno porte, é aconselhável o uso do *hypervisor* XenServer da Citrix, por ser gratuito e por seu notável desempenho para ambientes com poucas máquinas virtualizadas com sistemas operacionais heterogêneos, ou quando se buscar desempenho acima de tudo.

Como trabalhos futuros, ficam a proposta de estudos analisando o desempenho dos VMMs considerando os serviços prestados pelas VMs (servidor de *internet*, servidor de arquivos, servidor de banco de dados, entre outros), e a análise de estudos de desempenho de recursos de redes virtualizados.

REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS - ABNT, **NBR ISO/IEC 27002**, Rio de Janeiro: 2005.

AGOSTINHO, P.; **Virtualização em SAP**, Universidade Lusófona de Humanidades e Tecnologias, 6º SOPCOM, Lisboa, Portugal, 2009.

ANDRADE, M. T.; **Um estudo comparativo sobre principais técnicas de virtualização**. Recife, 2006. TCC, Centro de Informática, UFPE, 2006.

BARUCHI, J. H.; **Comparativo entre ferramentas de virtualização**, Faculdade de Jaguariúna, SP, 2008.

CARISSIMI, A.; **Virtualização: Princípios Básicos e Aplicações**, ERAD 2009, SBC, Caxias do Sul, 2009.

CITRIX SYSTEMS INC.; **Citrix Xenserver: efficient server virtualization software**. Disponível em <http://www.citrix.com/English/ps2/products/product.asp?contentID= 683148>, acesso em: 20/11/2010.

CLARK, C.; **User's Manual Xen v3.3**. 2002. Disponível em <http://bits.xensource.com/ Xen/docs/user.pdf>, acesso em: 20/11/2010.

COTTEN, P.; **Sun xVM Hypervisor Overview**. 2008. Disponível em <http://www.oracle. com/technetwork/systems/articles/xvmhvsrovw-jsp-141603.html>, acesso em: 20/11/2010.

CRUZ, D. I.; FLEXLAB: **Middleware de virtualização de hardware para gerenciamento centralizado de computadores em rede**. UNESP, SP, 2008.

FAVACHO, B. I.; MIRANDA, D. S.; OLIVEIRA, L. H. S.; **Análise comparativa do desempenho da técnica de virtualização de servidor**, TCC, UNAMA, Belém, 2008.

FREEBSD, ORG.; **Jails**. Disponível em <http://www.freebsd.org/doc/handbook/jails. html>, acesso em: 15/11/2010.

FREE SOFTWARE FOUNDATION.; **GNU General Public License v3**. Disponível em <http://www.gnu.org/licenses/gpl.html>, acesso em: 15/11/2010.

GIL, A. C.; **Como elaborar projetos de pesquisa**, São Paulo, Atlas, 1991.

GONÇALVES, D. B.; JUNIOR, J. C. V.; **White Paper – Virtualização**. Disponível em <http://www.sensedia.com/br/anexos/wp_virtualizacao.pdf>, acesso em: 15/11/2010.

JUNIOR, D. P. Q.; **Virtualização: Conceitos, técnicas aplicadas e um comparativo de desempenho entre as principais ferramentas sem custo de licenciamento**. Instituto Superior Tupy, Joinville, 2008.

LAUREANO, M.; **Máquinas virtuais e Emuladoras: conceitos, técnica e aplicações**. 1ª Edição, São Paulo, Novatec, 2006.

LYNUXWORKS, INC.; S**ecure virtualization and secure virtual machines: LynxSecure**. Disponível em <http://www.lynuxworks.com/virtualization/hypervisor.php>, acesso em: 20/11/2010.

MICROSOFT, CORP.; **Microsoft Virtual Server 2005 R2**. Disponível em <http://www.microsoft.com/windowsserversystem/virtualserver/downloads.aspx>, acesso em: 15/11/2010.

ORACLE, CORP.; **Oracle buys Virtual Iron**. 2009. Disponível em <http://www.oracle.com/us/corporate/press/018535>, acesso em: 20/11/2010.

ORACLE, CORP.; **Oracle VM VirtualBox – User's Manual version 3.2.10**. Disponível em < http://download.virtualbox.org/virtualbox/UserManual.pdf>, acesso em: 15/11/2010.

PASSMARK Performance Test. Versão 7.0 [S.I.]: PassMark Software. Disponível em <http://www.passmark.com/products/pt.htm>, acesso em: 15/02/2011.

PHORONIX Test Suite. Versão 3.0.1 [S.I.]: Phoronix Media. Disponível em <http://www.phoronix-test-suite.com>, acesso em: 23/03/2011.

POLLON, V.; **Virtualização de servidores em ambientes heterogêneos e distribuídos – estudo de caso**, UFRGS, RS, 2008.

REAL TIME SYSTEMS,; **Real-Time Hypervisor for Multicore Architecture**. Disponível em <http://www.real-time-systems.com/real-time_hypervisor/index.php>, acesso em: 20/11/2010.

ROSEMBLUM, M.; **The reincarnation of virtual machines**. Queue Focus, ACM Press, 2004.

ROSENBLUM, M.; **Virtual Machine Monitors: Current Technology and Future Trends**. IEEE Computer, Vol. 38, Issue 5. 2005.

SILVA, E. L.; **Metodologia da Pesquisa e Elaboração de Dissertação**. 3ª Edição, Florianópolis, UFSC, 2001.

SMITH, J.; NAIR, J.; **The Architecture of Virtual Machines**. University of Wisconsin-Madison, IEEE Computer Society, 2005.

STRIANESE, A.; **Virtualização: a TI virtual**. Disponível em <http://www.baguete.com.br/artigos/907/anibal-strianese/11/11/2010/virtualizacao-a-ti-virtual>, acesso em 20/11/2010.

VMWARE, INC.; **VMware Documentation**. Disponível em <http://www.vmware.com/support/pubs/>, acesso em 20/11/2010.

WILLIAMS, D.; GARCIA, J.; **Virtualization with Xen: Inclunding XenEterprise, XenServer and XenExpress**. Burlington. Syngress Publishing Inc, 2007.

XEN, ORG.; **Xen Hypervisor**. Disponível em <http://www.xen.org/products/xenhyp.html>, acesso em 20/11/2010.