



**UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ**  
**CAMPUS LUIZ MENEGHEL**

**GUILHERME ZUCATO**

**ANÁLISE DE DESEMPENHO DE BANCO DE DADOS  
EM SERVIDORES VIRTUALIZADOS**

Bandeirantes  
2014

**GUILHERME ZUCATO**

**ANÁLISE DE DESEMPENHO DE BANCO DE DADOS  
EM SERVIDORES VIRTUALIZADOS**

Trabalho de Conclusão de Curso submetido na Universidade Estadual do Norte do Paraná, como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Orientador: Prof. Me. Luiz Fernando Legore do Nascimento

Bandeirantes

2014

**GUILHERME ZUCATO**

**ANÁLISE DE DESEMPENHO DE BANCO DE DADOS  
EM SERVIDORES VITUALIZADOS**

Trabalho de Conclusão de Curso  
submetido a Universidade Estadual do  
Norte do Paraná, como requisito parcial  
para a obtenção do grau de Bacharel em  
Sistemas de Informação.

**COMISSÃO EXAMINADORA**

---

Prof. Me. Luiz Fernando Legore do  
Nascimento  
UENP – *Campus* Luiz Meneghel

---

Prof. Me. Ricardo Gonçalves Coelho  
UENP – *Campus* Luiz Meneghel

---

Prof. Me. Rafaella Aline Lopes da Silva  
UENP – *Campus* Luiz Meneghel

Bandeirantes, 26 de junho de 2014.

## RESUMO

Com os avanços da virtualização, muitas práticas que eram apenas utilizadas em servidores físicos, estão sendo utilizadas em servidores virtualizados, e Banco de dados é uma delas. Esse trabalho avaliou o Banco de Dados relacional Postgres 9.2 com o benchmarking Yahoo Cloud Serving Benchmarking em diferentes virtualizadores com suas técnicas de virtualização. Os virtualizadores testados foram o Xen Server, com utilização da técnica de paravirtualização e o VMware, com a virtualização total. Nesse, foram avaliados sobrecargas e o tempo de execução para as principais operações de Banco de Dados. Os erros de paginação os quais são costumeiramente encontrados em Banco de Dados Virtualizados sendo portanto repudiada essa técnica, podem segundo vários autores serem corrigidos com alguns ajustes no buffer de memória. Esses ajustes podem em alguns casos serem implementados sem muito esforço. Em outros casos, é necessária uma reengenharia completa da aplicação. Assim, não há uma forma única em resolver o problema, porém há solução. Esse trabalho concluiu que o problema em virtualizar um Banco de Dados não está na técnica da virtualização em si, mas na necessidade de ajustes os quais devem ser analisados caso a caso. Esse trabalho reforça, portanto o uso de Banco de Dados em servidores virtualizados.

**Palavras Chave:** Virtualização, Sistema Operacional e Banco de Dados

## ABSTRACT

With advances in virtualization, many practices that were only used on physical servers, are being used in virtualized servers, and database is one of them. This study evaluated the relational Database Postgres 9.2 with Yahoo benchmarking Benchmarking Cloud Serving on different servers with their virtualization techniques. The virtualizadores were tested Xen Server, using the technique of paravirtualization, and VMware, with full virtualization. In this, overloading and runtime for main operations database were evaluated. Errors of paging which are customarily found in Virtualized Data Bank is therefore repudiated this technique, according to several authors can be fixed with some adjustments in the memory buffer. These adjustments may in some cases be implemented without much effort. In other instances, a complete re-engineering of the application is required. Thus, there is no single way to solve the problem, but no solution. This study concluded that the problem to virtualize a bunch of data is not in the art of virtualization itself, but the need for adjustments which should be analyzed case by case. This work thus reinforces the use of Database in virtualized servers.

**Keywords:** Virtualization, Operating System and Database

## LISTA DE ABREVIATURAS E SIGLAS

<b>ACID</b>	Atomicity, Consistency, Isolation, Durability; Atomicidade, Consistência, Isolamento, Durabilidade
<b>DB</b>	Database, Banco de Dados
<b>SO</b>	Sistema Operacional
<b>SGBD</b>	Sistema Gerenciador de Banco de Dados
<b>VM</b>	Virtual Machine, Máquina virtual
<b>VMM</b>	Virtual Machine Monitor, Monitor de Máquina virtual
<b>YCSB</b>	Yahoo Cloud Serving Benchmarking

## LISTA DE FIGURAS

Figura 1. Sistemas não Virtualizados(a) e Sistema Virtualizados(b) (VERAS, 2011 pag 100) .....	15
Figura 2. Arquitetura da Virtualização Completa (VERAS , 2011 pag 106).....	18
Figura 3. Arquitetura de Paravirtualização (Fonte: SANTOS, 2011 pag 14) .....	19
Figura 4. VSphere Interface de gerenciamento Fonte: virtualizationsoftwares.com..	21
Figura 5. Arquitetura de virtualização do Xen (Xen Archives) .....	22
Figura 6 Funcionamento do YCSB.....	26
Figura 7 Servidor sendo testado por outra máquina (DDEBIAN). .....	32
Figura 8 Servidor se testando (DEBIAN).....	33
Figura 9 Tempo de execução em máquina não virtualizadas .....	34
Figura 10 Número de Operações por segundo em máquina não virtualizadas.....	35
Figura 11 Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Tempo de Execução para READ. ....	36
Figura 12 Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Operações por Segundo para READ.....	37
Figura 13 Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Tempo de Execução para UPDATE. ....	37
Figura 14 Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Operações por Segundo para UPDATE. ....	38
Figura 15 Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Tempo de Execução para INSERT.....	39
Figura 16 Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Operações por segundo para INSERT. ....	39
Figura 17. Média para Tempo de Execução.....	41

Figura 18 Média para Operações por Segundo .....42



## LISTA DE TABELAS

Tabela 1. Diferença entre os Tipos de Visualização (VERAS , 2011 pag 109).....	19
Tabela 2 Configuração padrão do benchmark YCSB.....	31
Tabela 3 Configuração específica do benchmark YCSB.....	32

## SUMÁRIO

<b>1 INTRODUÇÃO .....</b>	<b>10</b>
1.1 JUSTIFICATIVA .....	12
1.3 OBJETIVOS.....	12
1.3.1 <i>Objetivos Específicos</i> .....	12
1.4 ORGANIZAÇÃO DO TRABALHO .....	13
<b>2 FUNDAMENTAÇÃO TEÓRICA.....</b>	<b>14</b>
2.1 SISTEMA OPERACIONAL.....	14
2.1.1 <i>Máquina Virtual</i> .....	15
2.1.1 <i>Monitor de Máquina Virtual (VMM)</i> .....	16
2.2 TIPOS DE VIRTUALIZAÇÃO.....	16
2.2.1 <i>Emulação de hardware</i> .....	16
2.2.2 <i>Virtualização Completa</i> .....	17
2.2.3 <i>Paravirtualização</i> .....	18
2.3 SOFTWARES VIRTUALIZADORES .....	20
2.3.1 VMWARE.....	20
2.3.2 XEN.....	22
2.6 BANCO DE DADOS.....	22
2.6.1 <i>Postgres</i> .....	23
2.6.2 <i>Banco de dados virtualizados</i> .....	24
2.7 BENCHMARK.....	25
2.7.1 <i>O Benchmark TPC-H</i> .....	25
2.7.2 <i>O Benchmark Yahoo Cloud Serving Benchmarking</i> .....	26
<b>3 TRABALHOS RELACIONADOS.....</b>	<b>27</b>
3.1 SISTEMAS DE BANCO DE DADOS EM MÁQUINAS VIRTUAIS.....	27
3.2 VIRTUALIZAÇÃO DE BANCO DE DADOS: UMA NOVA FRONTEIRA PARA MELHORIAS DE BANCO DE DADOS E O PROJETO FÍSICO .....	29
<b>4 DESENVOLVIMENTO .....</b>	<b>30</b>

4.1 MÉTODOS .....	30
4.2 DESENVOLVIMENTO E RESULTADOS OBTIDOS .....	31
<b>5 CONCLUSÃO .....</b>	<b>41</b>

# 1 Introdução

Com o avanço tecnológico contínuo devido a grande evolução computacional nos últimos anos, a virtualização que engloba várias técnicas, vem crescendo constantemente. A primeira técnica utilizada na aparição da virtualização, foi a “*multiprogramming*” em 1959 onde em um único *mainframe* tinha pouco recurso de armazenagem, processamento e memória, que permitia rodar dois programas ao mesmo tempo por processos intercalados, e com ela é possível escrever o código fonte em um programa A e compilar em outro programa B (POLLON, 2008).

Essa técnica foi melhorada e hoje é possível observar seu emprego em diversos locais. Exemplo disso, é a existência de *hardware* que está sendo virtualizado e hospeda diversos Sistemas Operacionais (SO), os quais trabalham de forma independente uns dos outros, compartilhando um mesmo *hardware* físico, isto é, uma máquina física que pode ser virtualizada, hospedando diversas outras máquinas virtuais, denominadas de VM (*Virtual Machine*).

Segundo Lima e Santos, (2010) a virtualização abrange várias tecnologias, que compartilham os recursos de *hardware*, ou seja, os recursos de uma única máquina são compartilhados de forma a possibilitar que várias aplicações sejam executadas. Essa técnica permite que espaços físicos possam ser economizados, de forma que os custos sobre a aquisição de novos *hardware* e manutenção sejam reduzidos.

Ao se virtualizar um *hardware*, torna-se possível obter diversos benefícios (POLLON, 2008). Dentre eles destacam-se os seguintes recursos:

- Melhor tolerância a falhas;
- Maior mobilidade dos sistemas;
- Rápida recuperação de problemas;
- Maior disponibilidade de acesso;
- Reduz custos com manutenção e *hardware*;
- Melhor utilização dos recursos já existentes;
- Evita ociosidade da infra-estrutura;
- Diversos sistemas operacionais em uma mesma plataforma física.

Segundo Pollon (2008) também existem alguns malefícios, sendo o principal, a perda de desempenho acarretada pela inclusão da camada de *software* entre o SO e o *hardware*.

A virtualização depende da inserção de algumas ferramentas na camada de *software* entre o *hardware* físico e o sistema operacional, essa camada é chamada de *Hypervisor* ou Monitor de Máquina Virtual, também denominada pela sigla, VMM (Virtual Machine Monitor). Esta recebe todas as instruções do Sistema Operacional (SO) e as executa no *hardware* realizando o gerenciamento dos processos intercalados (LIMA e SANTOS, 2010).

Segundo Aboulhaga (2007), alguns *software* são prejudicados com o uso desse tipo de tecnologia. Os Sistemas Gerenciadores de Banco de Dados apresentam segundo Minhas et. al (2008), problemas referente à perda de desempenho. Tais perdas decorrem de erros de paginação e por existir uma camada extra para ser gerenciada pelo sistema virtualizador.

Segundo Minhas et. al, (2008), considerando que as vantagens em utilizar a técnica de virtualização se sobrepõem ao uso comum, desenvolvedores de *hardware* e *software* buscam constantemente melhorar o desempenho de sistemas virtualizados.

Atualmente, existem no mercado diversos *software* que fazem o gerenciamento de Máquinas Virtuais (VM), como é o caso do *Xen* (Xen, 2010); *VirtualBox* (ORACLE, 2011); *VMware* (VMware, 2011) entre outros.

A metodologia de Pesquisa utilizada para esse trabalho é de natureza Exploratória, onde foram realizados testes de desempenho do Banco de Dados em dois servidores de sistemas virtualizados. A pesquisa também utilizou o meio experimental, onde buscou-se as melhores técnicas de virtualização para um determinado *hardware*, além da configuração mais adequada para que o benchmark pudesse atender os objetivos propostos.

Esse trabalho analisou dois virtualizadores anteriormente citados, e comparou seus resultados de desempenho, os quais foram obtidos com testes baseados em uma ferramenta de análise: Yahoo Cloud Serving Benchmarking (YCSB) a fim de colaborar com a melhor escolha do virtualizador quando se pretende disponibilizar um serviço de Banco de Dados.

## 1.1 Justificativa

A tecnologia utilizada para virtualizar um determinado *hardware*, permite a redução do número de servidores físicos requeridos por uma organização bem como o espaço físico, a energia e requisitos de arrefecimento. Assim, considerando que a consolidação de servidores já é ampla e os recursos de virtualização já são utilizados por muitas empresas e prestadores de serviços, então questiona-se o motivo de existir algumas restrições quanto ao fato de se virtualizar sistemas de banco de dados (VERAS, 2011). E esse trabalho teve como principal justificativa a necessidade de responder tal questionamento. Para tal, buscou-se comparar o desempenho do serviço de Banco de Dados em um determinado Sistema Operacional instalado sobre um *hardware* físico e virtualizado, mudando sua ferramenta de virtualização.

Outro fator que justifica tal estudo, está no pequeno quantitativo de trabalhos sobre o desempenho de banco de dados nos diferentes *software* de virtualização. O que remete ao final deste trabalho é auxiliar à desenvolvedores de *software*, quanto a escolha do mais viável virtualizador (Xen ou VmWare) para Banco de Dados, contribuindo assim, para o processo de desenvolvimento de aplicações computacionais mais eficientes.

## 1.3 Objetivos

Quantificar a sobrecarga encontrada para sistema de Banco de Dados virtualizados, comparando com alguns dos principais virtualizadores disponíveis atualmente.

### 1.3.1 Objetivos Específicos

- Avaliar o desempenho de banco de dados PostgreSQL em um ambiente não virtualizado;

- Avaliar o desempenho de banco de dados PostgreSQL nos ambientes virtualizados: Xen e VmWare;
- Apresentar um relatório comparativo e quantitativo referente a todos os virtualizadores testados.

## **1.4 Organização do Trabalho**

Esse trabalho está organizado da seguinte forma: No Capítulo 2 é apresentado a Fundamentação Teórica sobre Sistemas Operacionais, os diferentes tipos de técnicas de virtualização, e o Banco de Dados Postgres. No Capítulo 3 são apresentados alguns trabalhos relacionados ao tema. No Capítulo 4 encontra-se a base na metodologia de pesquisa e o desenvolvimento com os resultados obtidos no trabalho. O Capítulo 5 apresenta as conclusões obtidas a partir dos resultados.

## 2 Fundamentação Teórica

O intuito deste capítulo é familiarizar o leitor sobre algumas tecnologias de virtualização, permitindo que seja criado embasamento teórico sobre os assuntos os quais serão mais aprofundadamente tratados no desenvolvimento deste trabalho.

### 2.1 Sistema Operacional

O Sistema Operacional (SO) é composto por um conjunto de programas que gerenciam recursos e tarefas, os quais ficam de intermediários entre o usuário e o computador. Todos os componentes de um computador são gerenciados pelo seu SO, assim como seus dispositivos de entrada e saída.

*“Sistemas Operacionais funcionam baseados em processo. Um processo é uma abstração que representa um programa em execução. Cada processo é um ambiente de execução isolado dos demais processos que executa sobre um processador lógico, isto é, um processador virtual, vinculado a si no momento da criação do processo. Cabe ao núcleo do sistema operacional, através de seu escalonador, alternar os diferentes processadores lógicos (virtuais) sobre um processador físico. A ilusão de paralelismo é criada pelo chaveamento rápido entre os processos.” (MANOEL VERAS , 2011, p.98).*

O paralelismo é quando diversas tarefas são executadas de forma tão rápida que passa uma ilusão de que foram executadas ao mesmo tempo (MANOEL VERAS ,2011).

Existem diversos tipo de Sistemas Operacionais proprietários e de uso livre (Open Source) como Windows e distribuições Linux. Alguns para servidores que contém apenas ferramentas essenciais para seu funcionamento e para aquilo que será designado, como servidor de página ou Banco de Dados. Segundo Aboulmaga, 2007 escolha de um SO é fundamental para um bom resultado.



Em sistemas virtualizados, o SO é instalado e configurado em máquinas virtuais, essas gerenciadas pelo sistema virtualizador.

### 2.1.1 Máquina Virtual

A Máquina Virtual (VM) é uma simulação da instância de um *hardware* virtualizado, que funciona de forma isolada em diferentes ambientes. Todos os ambientes virtuais com seus próprios sistemas operacionais executam de forma isolada qualquer tipo de *software* sobre um mesmo recurso físico bem similar a uma máquina real. (SANTOS, 2011).

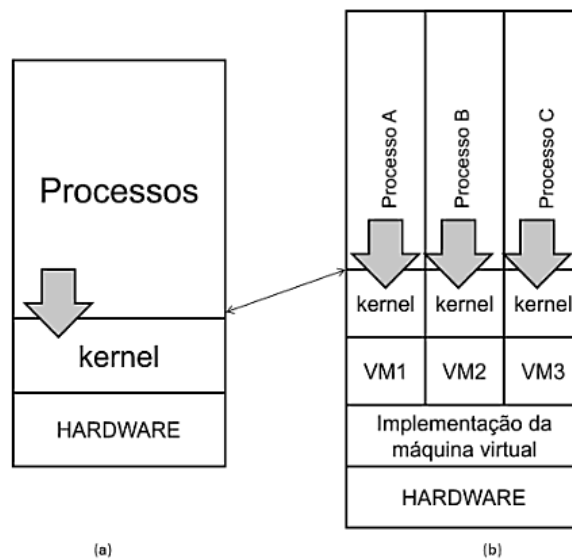


Figura 1. Sistemas não Virtualizados(a) e Sistema Virtualizados(b) (VERAS, 2011 pag 100)

A Figura 1, mostra uma comparação entre os recursos alocados de forma virtual e não virtual. A imagem “a” representa o sistema operacional convencional onde se encontra o *hardware*, o *kernel* (Sistemas Operacionais) e os processos executados.

A imagem “b” mostra o mesmo *hardware* com a implementação da máquina virtual, o qual possibilita vários (*kernel*) Sistemas Operacionais cada um em uma máquina virtual.

Atualmente, existem inúmeras pesquisas com o intuito de melhorar o desempenho de VM com foco na abordagem de operações de entrada e saída. Pelo fato de ter uma camada de *software* adicional. Isto influencia diretamente no desempenho das VM's (BARUCHI e MIDORIKAWA, 2008).

A virtualização é utilizada amplamente na área de servidores por oferecer benefícios significativos relacionados a gerenciamento, disponibilidade, flexibilidade, portabilidade, assim como a recuperação de falhas (CAMPOS e VASCONCELOS, 2010).

### **2.1.1 Monitor de Máquina Virtual (VMM)**

Os monitores de máquinas virtuais VMM (*Virtual Monitor Machine*) ou HYPERVISORS, estão na camada de *software* e fazem o gerenciamento dos recursos do *hardware* para criar um ambiente virtual, onde um visitante (*quest system*) executará seu SO isolado de outro convidado (*quest*). Nesse, todos compartilham o mesmo *hardware*. (SANTOS, 2011).

Os monitores de máquinas Virtuais surgiram para melhorar uma das maiores dificuldades de virtualização, que é administrar uma camada a mais de *software*, isso, segundo Manoel Veras, 2011, acarreta uma perda de desempenho.

Tanto nos virtualizadores VMware quanto no Xen Server há um *hypervisor* ou VMM, entretanto, com abordagens diferentes: o Xen utiliza a técnica de paravirtualização, enquanto o VMware utiliza virtualização total.

## **2.2 Tipos de Virtualização**

Os virtualizadores citados no tópico anterior utilizam de técnicas de virtualização, como: a paravirtualização e a virtualização total. Cada uma tem sua abordagem de como interagir com o *hardware* e administrar a Máquina Virtual. Esse tópico irá abordar as principais técnicas de virtualização.

### **2.2.1 Emulação de *hardware***

Na técnica de emulação de *hardware* é possível simular um *hardware* completamente diferente do físico para suportar sistemas de arquiteturas novas em máquinas antigas como em máquinas novas suportando arquiteturas antigas, porém como todas as instruções são mapeadas e gerenciadas de uma arquitetura para outra, ocorre uma considerável perda de desempenho. (SANTOS e LIMA, 2010).

Os recursos e funções apresentados na máquina virtual podem não estar disponíveis nos recursos físicos do servidor. Nesta técnica de virtualização, o *hardware* da VM é emulado por *software* nos sistemas hospedeiros (VERAS, 2011).

### 2.2.2 Virtualização Completa

A virtualização completa é uma técnica preferencialmente utilizada em Sistemas Operacionais com arquitetura x64, entretanto, em casos de arquiteturas x86 utiliza-se a técnica de “*binary trasnlation*”, para se evitar erros em algumas chamadas de instruções pelo processador.

A virtualização "total" ou completa, funciona através de um sistema hospedeiro que faz uma simulação completa do *hardware* físico para que qualquer outro sistema operacional *guest* (convidado) possa ser executado. A VMM é a responsável para que não haja necessidade de alteração prévia nas máquinas virtuais (SANTOS, 2011).

Uma das vantagens da virtualização completa quando comparada com a técnica de virtualização paravirtualizada é a portabilidade. Em tal prática não é necessário que a máquina virtual seja previamente configurada para executar, facilitando na locomoção e recuperação de determinada máquina virtual. Outra vantagem é a maior compatibilidade de *hardware* entre as VM's (COSTA, 2010).

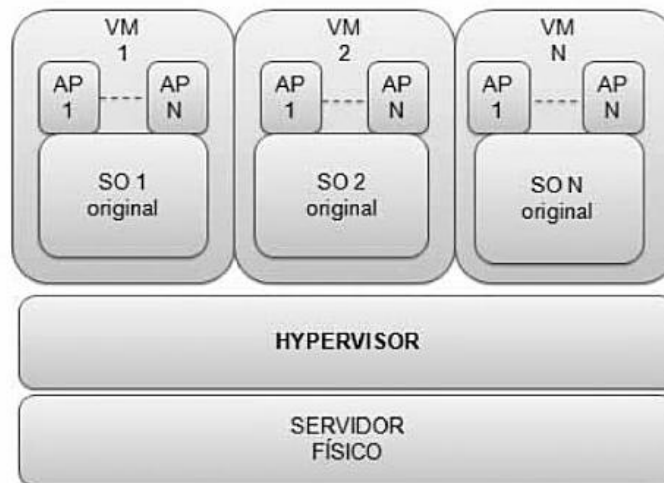


Figura 2. Arquitetura da Virtualização Completa (VERAS , 2011 pag 106)

A Figura 2 ilustra o funcionamento de uma VMM. Nessa, é garantido o isolamento dos recursos utilizados pelos SO *guests* em tempo integral. O *Hypervisor* ou VMM faz todo o gerenciamento de recursos e instruções das VM's.

### 2.2.3 Paravirtualização

A paravirtualização é uma técnica de virtualização semelhante à virtualização total, mas possui algumas diferenças (representadas na Tabela 1).

A paravirtualização necessita que o sistema hospedeiro de sua máquina virtual seja modificado para ter uma maior interação direta com o *hardware*, agilizando alguns processos específicos, otimizando o desempenho, sem a necessidade do intermédio da VMM para todas as execuções. Com isso, é permitido que o sistema consiga obter uma melhor eficiência, já que não depende da interpretação do VMM para todas as execuções. A Figura 3 ilustra a comunicação extra entre a VM e a VMM (CAMPOS e VASCONCELOS, 2010).

Dentre as várias tecnologias de virtualização, a paravirtualização segundo Santos, (2011) apresenta a melhor abordagem para aqueles que buscam a redução de custos através de servidores virtualizados. Ela apresenta um ganho de desempenho significativo em processos específicos comparado as demais.

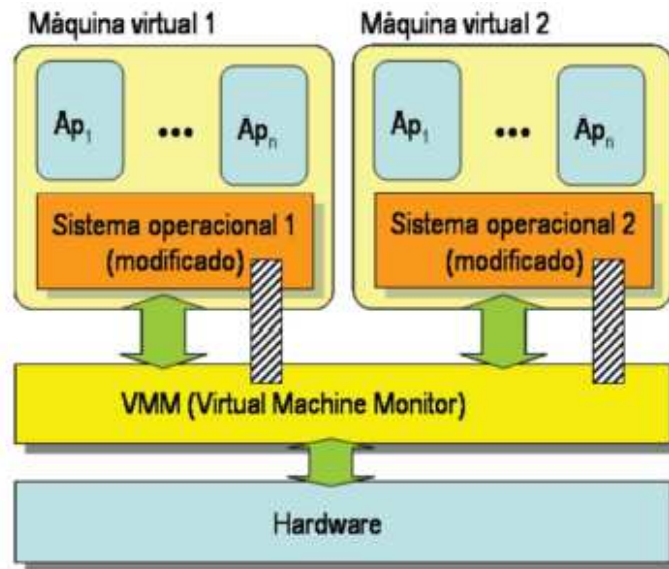


Figura 3. Arquitetura de Paravirtualização (Fonte: SANTOS, 2011 pag 14)

A necessidade de recursos físicos se faz necessária para garantir o desempenho de Banco de Dados, a escolha de utilização da paravirtualização se torna mais viável por ser uma técnica que faz boa alocação dos recursos físico para a VM's, mantendo um desempenho elevado dos Sistemas Operacionais virtualizados, quando comparado com outras técnicas de virtualização (SANTOS e LIMA, 2010).

	VIRTUALIZAÇÃO TOTAL	VIRTUALIZAÇÃO ASSISTIDA POR HARDWARE	PARA VIRTUALIZAÇÃO
<b>Técnica</b>	Translação Binária e Execução Direta	Saída para modo raiz nas instruções privilegiadas	<i>Hypercalls</i>
<b>Modificação do SO Convidado/ Compatibilidade</b>	SO convidado não modificado/excelente	SO convidado não modificado/excelente	SO convidado modificado/baixa compatibilidade
<b>Desempenho</b>	Bom	Considerável	Melhor em certos casos
<b>Usado por</b>	VMware, Microsoft, Parallels	VMware, Microsoft, Parallels, Xen	VMware, Xen
<b>Independência entre SO convidado e VMM</b>	Sim	Sim	Não

Tabela 1. Diferença entre os Tipos de Virtualização (VERAS , 2011 pag 109)

A Tabela 1 mostra um comparativo entre os tipos de virtualização. Nessa, a paravirtualização apresenta-se como sendo o melhor resultado (VERAS, 2011).

## 2.3 Softwares Virtualizadores

Nessa seção são apresentados, alguns dos principais virtualizadores disponíveis e suas técnicas de virtualização.

### 2.3.1 VMware

O VMware é uma ferramenta de virtualização que utiliza ambas as técnicas de virtualização e tem várias versões, a utilizado foi a vSphere por não necessitar de licença.

A arquitetura do VMware vSphere se concentra em capacitar as organizações de TI a entregar os serviços de forma flexível, escalável e confiável. Todos os tipos de componentes como processador, armazenamento e dispositivos de rede são virtualizados para fornecer um grande ambiente de operação dinâmico.

Nessa arquitetura, inicialmente o *hardware* servidor precisa de uma camada de abstração em forma de um *hypervisor* do sistema operacional. Para isso, são necessários os *softwares*: VMware ESX e VMware ESXi, os quais fazem o gerenciamento da alocação dos recursos do *hardware* para a VM's e acompanham seus desempenhos mostrado na interface gráfica. Figura 4.

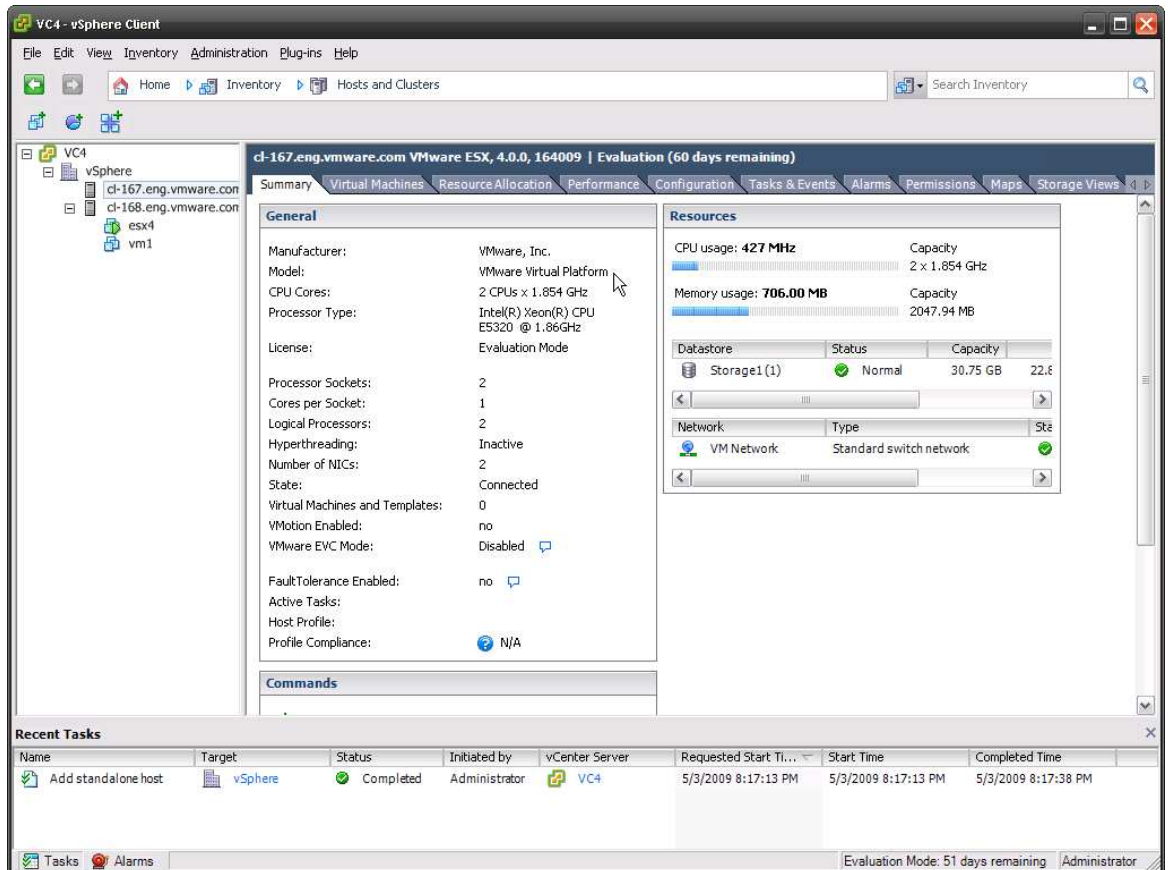


Figura 4. VSphere Interface de gerenciamento Fonte: virtualizationsoftwares.com

Uma vez que as ferramentas de virtualização estejam instaladas no servidor físico, esse passa a ter recursos de *hardware* abstraídos, como processadores, memória e armazenamento. Isto proporciona a criação de uma base de máquinas virtuais em execução utilizando este mesmo *hardware* (VMWare, 2011).

Para gerenciar *hypervisors* VMware utiliza-se o VMware vSphere vCenter Server. O ponto central para configuração, garantindo o gerenciamento da infraestrutura virtual. Todos os centros de dados, incluindo sistemas de acolhimento, armazenamento de dispositivos, interfaces de rede e máquinas virtuais são gerenciados por esse servidor (JIN, 2009).

O VMware vSphere vCenter Server pode segundo seus desenvolvedores ser instalado em *hardware* físico, mas também é possível executar o *software* de uma máquina virtual. Além disso, há ainda o *software* cliente, o qual permite que todos os tipos de componentes como: processadores, dispositivos de rede e armazenamento, sejam virtualizados de forma a fornecer um grande ambiente de operação dinâmico (PAPE e TROMMER, 20012).

### 2.3.2 Xen

O Xen é mais um virtualizador, que por sua vez permite rodar vários Sistemas Operacionais como: Linux, FreeBSD ou Windows em um servidor físico. Ele utiliza como tecnologia a paravirtualização, o qual é encarregado de organizar as requisições feitas pelas máquinas virtuais e repassá-las ao sistema principal. O Xen limita-se a repassar as instruções, sem interpretá-las como faria um emulador, causando uma diminuição de desempenho muito pequena em relação à emulação (POLLON, 2008).

Se comparado a outros virtualizadores, o Xen tem por objetivo ser mais voltado para uso em servidores, que permite rodar vários servidores virtuais numa única máquina.

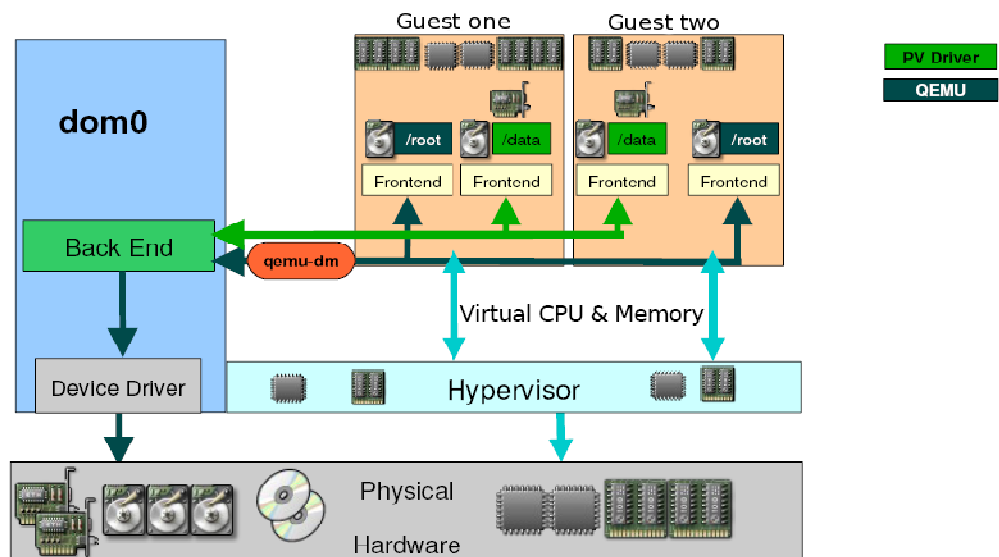


Figura 5. Arquitetura de virtualização do Xen (Xen Archives)

A Figura 5 mostra a maneira de como o Hypervisor faz o gerenciamento dos recursos do *hardware* para as VM. O Xen cria um domínio onde sua comunicação com as VM's corresponde aos dispositivos de armazenamento e o VMM (Hypervisor) fica responsável pela memória e processamento.

## 2.6 Banco de Dados

O Banco de Dados (Data Base ou DB), propriamente dito, é onde as informações estão localizadas em formas de dados sobre registros e em vários



formatos e complexidades. Todo o Banco de Dados possui um conjunto de programas denominados Sistema Gerenciador de Banco de Dados (SGBD). Tais programas são executados para disponibilizar uma interface de criação e manipulação dos dados, de forma a garantir a consistência e integridade da informação.

Existem diversos tipos de Banco de Dados, e nesse trabalho foi abordado o “Banco de Dados Relacional”, que é formado por uma coleção de tabelas únicas interligadas por relacionamento de dependência, que garante a integridade referencial dos dados por chaves primárias e estrangeiras, a fim de auxiliar na recuperação, comparação e tratamento desses dados para que produza resultado com informações relevantes (BOSCARIOLI, et al 2006).

Os Banco de Dados relacionais têm como as principais operações *Insert* que armazena algum valor ou valores em um campo ou tabela de uma base de dados, o *delete* que apaga tanto dados como tabelas e bancos inteiros, *read* que busca algum valor ou tabelas fazendo uma consulta na base de informação e o *update* que atualiza uma tabela com alguma alteração. Todas essas operações são garantidas pela Atomicidade, Consistência, Isolamento e Durabilidade (ACID) onde a atomicidade garante que uma transação seja totalmente executada ou ela não acontece para garantir os dados totalmente consistentes. A consistência garante que em toda transação em que o estado estava consistente, o mesmo deve permanecer consistente no final da transação. O isolamento garante que os dados de uma transação fiquem indisponíveis durante essa transação, para não ocorrer mudanças diferentes de um mesmo dado. A durabilidade garante que as transações são permanentes e seja possível recriar alguma transação em caso de falha (PERKUSICH et al, 1999).

Existem diversos Sistemas de Gerenciamento de Banco de Dados com suporte a ACID e o Postgres 9.2 é um deles.

### **2.6.1 Postgres**

O PostgreSQL é um SGBD de código aberto, que trabalha com objeto-relacional totalmente compatível com a ACID (Atomicidade, Consistência, Isolamento e Durabilidade) e suporta grande parte do padrão SQL, assim como, oferece muitas funcionalidades.

A arquitetura do sistema básico do PostgreSQL utiliza o modelo cliente-servidor, no qual o servidor é responsável pela conexão com as aplicações clientes e execução das ações no Banco de Dados em nome dos clientes. O programa servidor do Banco de Dados é denominado: *postmaster*. O aplicativo cliente pode ter diversas naturezas como ser uma aplicação Web ou uma ferramenta especializada em manutenção de DB, essas fazem a conexão com banco via TCP/IP e aguardam a resposta do *postmaster* (POSTGRESSQL , 2005).

### **2.6.2 Banco de dados virtualizados**

Os Banco de Dados virtualizados têm vantagens por englobar todos os benefícios que a virtualização traz , como a possibilidade de copiar e migrar uma VM no mesmo estado em que ela se encontrava. Isso no âmbito de Banco de Dados é uma relevante qualidade, porque os SGBDs - por trabalhar com enormes volumes de dados - têm um alto consumo de recursos. A instalação e configuração demandam de tempo, pois não é simples otimizar seu desempenho. Dado esses fatores, é bastante viável virtualizar um DB, uma vez que a recuperação deste elimina o tempo de preparo do ambiente (LIMA e SANTOS, 2010).

Outra vantagem é a velocidade de entrega de um serviço, a qual ocorre de maneira rápida por as imagens se encontrarem previamente configuradas, além de todos os Bancos de Dados poderem ser virtualizados.

Por ter um alto consumo de recursos, a perda de desempenho é a maior desvantagem de virtualizar um serviço de Banco de Dados, pois o custo de outras VM adicionais sobre o mesmo *hardware* influencia diretamente no desempenho deste serviço (MINHAS, 2008).

Essa perda de desempenho de Banco de Dados virtualizados, segundo (MINHAS, 2008), ocorre devido a falhas de paginação, as quais são perdidas na memória principal, ocasionando uma perda aproximada de 10% no desempenho do banco.

## 2.7 Benchmark

Com os avanços nas arquiteturas computacionais e os diversos fornecedores de sistemas computacionais, a única comparação era feita apenas pelas especificações dos fornecedores. Para melhor avaliação do desempenho dos sistemas computacionais, ferramentas foram criadas, denominadas de Benchmark.

O Benchmark é um processo que testa um sistema ou serviço computacional, através de ferramentas configuradas e conjuntos de instruções adequadas para sobrecarregar o sistema avaliado, gerando relatórios comparativos (CAMPOS e VASCONCELOS, 2010).

Dois conceitos fundamentais para se avaliar um sistema virtualizado é o de *Workload* e *Throughput*. O *Workload* é o escopo das instruções que serão executadas (sobrecargas), e suas especificações são modificadas para cada serviço, normalmente aplicados em SGBD's.

*“O Throughput define a capacidade do hardware/software para processar dados. É o composto de velocidade de I/O(entrada e saída), velocidade de processamento capacidade de paralelismo e eficiência do Sistema Operacional” (MANOEL VERAS , 2011, p.86).*

O *Throughput* é uma unidade básica quando se pretende mensurar desempenho de sistemas virtualizados.

### 2.7.1 O Benchmark TPC-H

O *Benchmark* citado é o utilizado no método de Minhas 2008, que é uma referência de apoio à decisão. Trata-se de uma ferramenta popular utilizada para comparar a eficiência de diversos fornecedores de banco de dados. Sua métrica de desempenho é chamada TPC-H Composite Query-per-Hour Performance Metric, a qual reflete aos aspectos de desempenho do sistema em processar consultas, executando-as em um único fluxo, além da taxa de transferência quando as

consultas são feitas por múltiplos usuários, auxiliando na tomada de decisão de qual é o melhor DB para determinada situação (TPC, 2013).

O Benchmark TPC-H auxilia determinando qual a melhor situação de ambiente em que o SGBD Postgres se mostra mais eficiente quanto ao desempenho.

### 2.7.2 O Benchmark Yahoo Cloud Serving Benchmarking

O Benchmark YCSB é uma ferramenta de código livre (Open Source), que atende todas as necessidades para se mensurar um servidor virtualizado, pois funciona tanto internamente como externamente do servidor possibilitando uma melhor comparação.

O processo de teste de desempenho funciona a partir da escolha do Banco de Dados, assim como o arquivo de configuração do YCSB. A maneira de como será a sobrecarga é definida no arquivo de *Workload*, que contém as especificações da sobrecarga, a quantidade de operações e a instrução dessas operações, além de quantos processos serão abertos para executar essas sobrecargas.

O *Workload* é executado e cria os processos que irão testar o Banco de Dados, por final um relatório é gerado com as informações do tempo de execução e de *Throughput*.

As sobrecargas têm o tamanho de 1 Kb para todas as operações. O funcionamento de uma sobrecarga do YCSB é mostrado na Figura 6.

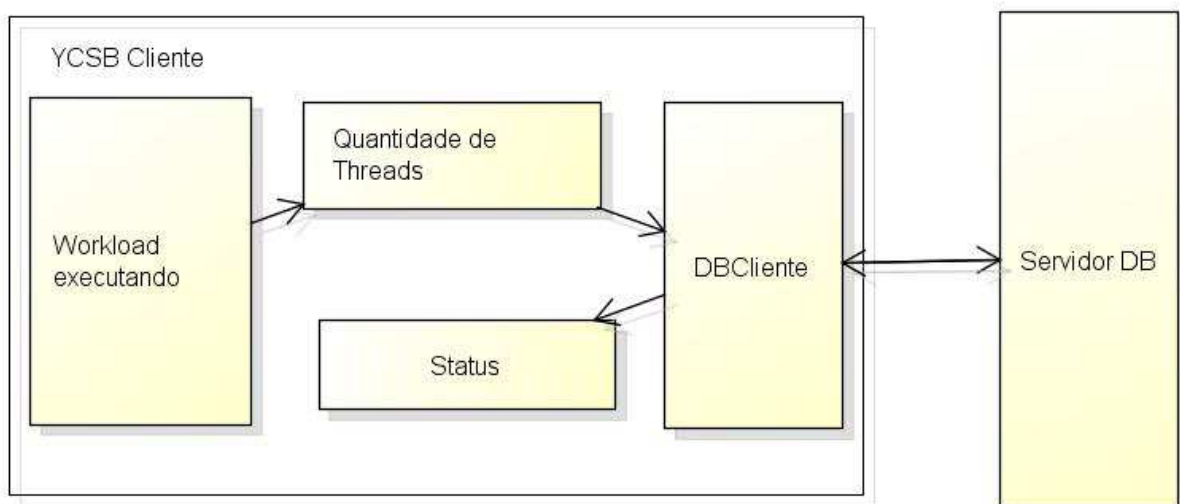


Figura 6 Funcionamento do YCSB

## 3 Trabalhos Relacionados

Nesse capítulo, são apresentadas algumas referências bibliográficas as quais devido a sua metodologia de testes, foram utilizadas para embasamento desse trabalho.

Durante o levantamento bibliográfico, encontraram-se poucos trabalhos acadêmicos disponíveis sobre testes em máquinas virtuais. Nossa referência foi centrada em dois artigos, pelo fato destes se aproximarem do objeto desse estudo. Esses são: Sistemas de Banco de Dados em Máquinas Virtuais, por Minhas, et. al (2008) e Virtualização de Banco de Dados: Uma Nova Fronteira para Melhorias de Banco de Dados e o Projeto Físico, esse por Abounaga (2007).

### 3.1 Sistemas de banco de dados em máquinas virtuais

Nesse trabalho, Minhas, et. al (2008), apresenta a utilização de máquinas virtuais as quais garantem meios simples e práticos para resolver muitos problemas de sistemas gerenciadores de bancos de dados (SGBD), como alteração de ambiente, implementação, consolidação e recuperação de servidor.

O trabalho de Minhas, et. al (2008) teve por objetivo responder a pergunta relacionado a perda de desempenho de banco de dados em ambiente virtualizado apenas com o Xen. Segundo o autor, ocorrem falhas de página de memória. Essas são uma exceção gerada pelo *hardware* quando ocorre uma instrução a qual não foi carregada na memória física (falha de página maior), ou esta é carregada por outro processo e não é mapeado em seu espaço de endereço (falha de página menor).

Como metodologia utilizada para avaliar o desempenho, o autor utilizou uma máquina servidora x64 Sun Fire X4100 com dois processadores 2.2GHz AMD Opteron Modelo 275 dual core. A ferramenta de avaliação foi benchmark TPC-H sobre o PostgreSQL 8.1.3. Os testes consistiram em uma comparação do desempenho de uma máquina virtual usando hypervisor Xen 3.1 e um sistema nativo (Base), ambos com o sistema operacional SUSE Linux 10.1 com a versão 2.6.18 do *kernel* e o sistema de arquivos ReiserFS.

A carga de trabalho utilizada nos experimentos foi a OSDL (Open Source Development Labs) para ajudar no apoio de decisão com a implementação do

benchmark TPC-H com o fator de escala de 1 GB. Foram feitas 22 consultas, nenhuma relacionada a *update*. A ferramenta para análise do desempenho do sistema virtualizados foi o XenMon. O banco de dados utilizado possuía o tamanho de 2GB, assim como o tamanho do *buffer* do Postgres. Para assegurar que caiba todo o banco de dados na memória em experimentos aquecidos (onde o sistema já fora utilizado antes).

Foram realizados dois experimentos para medir a sobrecarga do Xen e do sistema nativo, que são: aquecido e frio.

No sistema aquecido, o *cache* do sistema de arquivos do Linux e o *buffer* do Postgres são aquecidos antes de realizar a medição. No sistema frio, a memória *cache* e o *buffer* estão limpos antes de se realizar os testes de medição, e para obter maior precisão foram feitos cinco testes em cada experimento e a variação segundo o autor, foi muito baixa.

Nesse trabalho, foi possível concluir que não ocorreu grande diferença no número falhas no Xen e no Base. O problema, portanto, passou a consistir no tempo que cada sistema leva para corrigir essa falhas. Um kit de ferramentas, denominada Imbench fez essa medição e os resultados mostraram que o sistema nativo leva  $1.67\mu\text{s}$  enquanto o Xen demora  $3.5\mu\text{s}$  para lidar com a falha. O Xen levou portanto duas vezes o tempo para corrigir a mesma falha, sendo esse o motivo para a maior desaceleração do banco de dados.

Nesse trabalho, os autores ainda fazem algumas configurações com o objetivo de reduzir ao mínimo o número de falhas. Nessa, a página de memória do *buffer* e do *cache* compartilhado foi previamente mapeada. Segundo os autores, quando uma falha de página é corrigida e mapeada ela não ocorre novamente. Outra modificação feita é que todas as consultas no Postgres foram feitas a partir do mesmo cliente Postgres de forma a ter apenas uma conexão e não um para cada consulta.

Ao fim os autores ressaltam as vantagens em se utilizar uma máquina virtual, pois as mesmas sobrepõem as desvantagens. O estudo mostrou que não há um custo elevado de desempenho e o tempo de execução com correções de falhas fica semelhante ao sistema nativo.

### 3.2 Virtualização de Banco de dados: Uma nova fronteira para melhorias de Banco de Dados e o projeto físico

Nesse outro trabalho relacionado, Abounaga (2007) faz também alguns ajustes, de forma a melhorar o desempenho de sistema de Banco de Dados (SGBD). Nesse, o autor ajusta parâmetros os quais controlam recursos físicos como CPU, memória, e I/O (entrada/saída).

Esses parâmetros tidos em uma máquina virtual interagem também com os ajustes do próprio Banco de Dados com o tamanho do *buffer* o qual deve segundo o autor ser semelhante ao reservado para máquina virtual. É importante definir esses ajustes quando se tem várias máquinas virtuais compartilhando de um mesmo recurso físico. Além disso, o autor afirma que se torna importante que haja modelagem do desempenho de um sistema Banco de Dados dentro de uma máquina virtual, sendo para tanto necessário o uso do otimizador de consultas do próprio Banco de Dados.

Para esse experimento foi utilizado uma máquina com dois processadores 2.8 GHz Intel Xeon e 4 GB de memória rodando SUSE Linux 10.0, *kernel* 2.6.16, o SGBD foi o PostgreSQL 8.1.3 e o virtualizador foi o Xen 3.02 hypervisor. Foram feitos testes de carga com o *benchmark* TPC-H com a implementação OSDB em um BD de 1GB com os índices é de 4GB.

Os testes consistiram de duas cargas de trabalho, uma com 4 consultas e outra com 13. Ambas executadas em máquinas virtuais instaladas na mesma máquina física.

Diversos testes foram realizados alterando o processador e as cargas de trabalho, para concluir que não houvesse melhor alocação de recursos do que independentemente da carga de trabalho dividir os recursos disponíveis igualmente e o modelo com o algoritmo faz a identificação da melhor alocação de recursos, mostrou eficiente para resolver alguns problemas de projetar uma VM para sistemas de Banco de Dados.

## 4 Desenvolvimento

Nesse Capítulo é apresentado o desenvolvimento e os métodos utilizados nesse trabalho.

### 4.1 Métodos

Conforme apresentado no Capítulo 3, buscou-se a partir dos trabalhos relacionados, utilizar parâmetros similares para a composição dessa metodologia de avaliação. Nessa, optou-se pela escolha dos seguintes virtualizadores: XenCitrix, VMware vSphere vCenter. Procurou-se também comparar desempenho com exemplo de paravirtualização (XEN) e virtualização completa (VMware), que é o caso do VMware.

Duas abordagens foram feitas afim de criar um ambiente de teste. A primeira consiste no em uma máquina externa ao banco a qual enviasse testes de sobrecarga ao servidor de Banco de Dados.

Na segunda abordagem, o benchmark estava instalado na própria máquina onde o se encontrava o Banco de Dados. O objetivo destas duas abordagens era confrontar e avaliar possíveis perdas de desempenho no caso da máquina testadora testar ela própria.

O *hardware* físico escolhido para o ambiente de testes possuía as seguintes configurações: (01) processador com 06 núcleos, clock de 2,0 GHz, 08 GB de memória RAM e 02 discos SAS (Serial Attached SCSI), sendo a plataforma 64 bits.

Para obter parâmetros sobre o desempenho dos virtualizadores escolhidos, inicialmente foi instalado o Sistema Operacional sobre o próprio *hardware*, isto é, sem nenhum tipo de virtualização. Desse modo tornou-se possível comparar o desempenho sem que houvesse perdas utilizando as técnicas de virtualização.

O Sistema Operacional escolhido, para o ambiente Linux, foi a distribuição Debian Wheezy - kernel 3.2.54 (versão estável) e o Banco de Dados o PostgreSQL 9.2 (versão estável).

Diferentemente das metodologias utilizadas nos trabalhos de Minhas (2008) e Aboulnaga (2007). As ferramentas utilizadas para avaliação de desempenho foram: YCSB (Yahoo Cloud Serving Benchmarking) de natureza Open Source. A escolha



se deve pelo fato deste permitir mensurar o desempenho de um Banco de Dados tanto na raiz da máquina testada, quanto em outro computador. Além disso, esse benchmark permitiu efetuar testes com as 3 operações de banco de dados, Insert, Read, Update.

As operações nele realizadas possuíam sobrecargas de cem mil, um milhão, dois milhões e cinco milhões para cada uma das operações. O delete não foi testado, pois em casos de apagar uma Base de Dados ou uma tabela, utiliza-se diretamente o SGBD com a função *drop*, e o *benchmark* YCSB não possui uma sobrecarga para essa função.

## 4.2 Desenvolvimento e Resultados Obtidos

Conforme apresentado no Capítulo 4.1, o qual refere-se a metodologia, o benchmark utilizado foi o YCSB. Esse permitiu testar remotamente o Banco de Dados e especificar cada tipo de sobrecarga para uma operação específica como para: *insert*, *update* e *read*.

Para isso foram necessários alguns ajustes nos *workloads* (arquivos de configuração de sobrecargas) onde foi configurado as especificações de sobrecargas.

Os *workloads* utilizados foram o B, C e D. Esses possuem configuração padrão conforme são mostrados na Tabela 2:

Operação \ <i>Workload</i>	B	C	D
Read	0.95	1	0.95
update	0.05	0	0
Scan	0	0	0
Insert	0	0	0.05

**Tabela 2** Configuração padrão do benchmark YCSB.

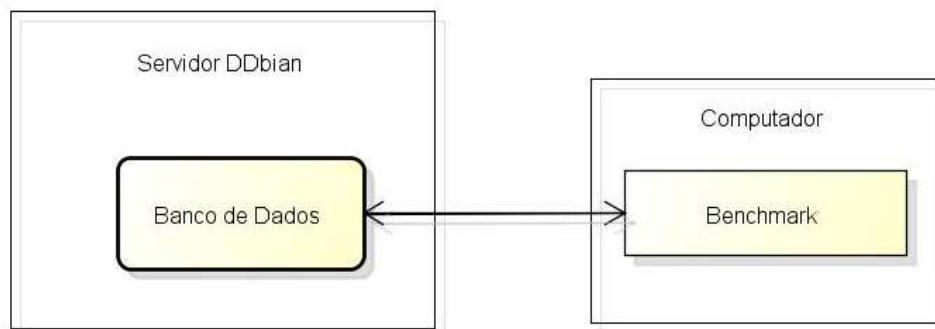
Como o objetivo inicial era testar as quatro operações de Banco de Dados e avaliar cada uma delas de forma separada, foi necessário efetuar algumas modificações nas configurações do benchmark YCSB. Dessa forma os testes ficaram especificados da seguinte forma: Tabela 3.

Operação \ <i>Workload</i>	B	C	D
Read	0	1	0
Update	1	0	0
Scan	0	0	0
Insert	0	0	1

**Tabela 3 Configuração específica do benchmark YCSB**

Uma vez definido e configurado o benchmark YCSB iniciou-se a fase de avaliação de desempenho.

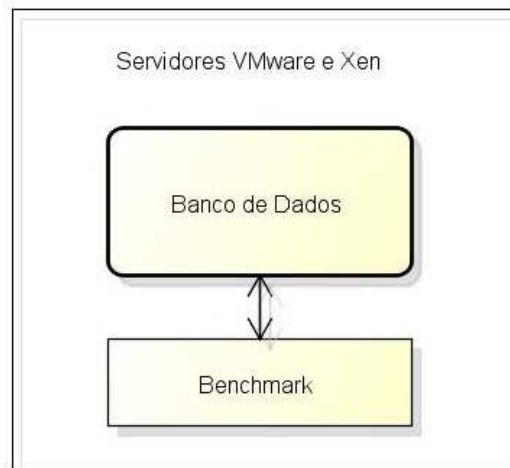
Na primeira abordagem uma máquina externa ao servidor de Banco de Dados foi conectada via rede 100 Mbits. Entre as duas máquinas não houve nenhum processo de roteamento o qual pudesse ser considerado como perda de desempenho ou possíveis falhas, ilustrado na Figura 7.



**Figura 7 Servidor sendo testado por outra máquina.**

A máquina utilizada para o envio da sobrecarga do benchmark possuía uma configuração de *hardware* inferior a máquina que hospedava o Banco de Dados, contudo não se observou sobrecarga de processamento e memória para o cumprimento das tarefas, ambas com o mesmo Sistema Operacional e sem nenhum tipo de virtualização.

Para a segunda abordagem foi igualmente utilizado o sistema operacional Debian, testando ele mesmo para se ter uma comparação entre os servidores. Nesse, o mesmo *hardware* testa e recebe os testes, isto é, um único *hardware* foi utilizado para testar a si mesmo ilustrado na Figura 9.



**Figura 8 Servidor se testando.**

No conjunto de gráficos mostrados a seguir, Figura 9 pode-se observar que todos os testes iniciais utilizando a primeira abordagem, apontam uma perda de desempenho quando o testador é uma máquina externa a máquina que hospeda o Banco de Dados.

Nesse conjunto de figuras, são apresentados o resultado dos testes nas três operações de banco de dados relacionados ao tempo de execução e operações por segundo para “DEBIAN” onde o servidor si testa e “Máquina Testadora” utilizando a primeira abordagem.

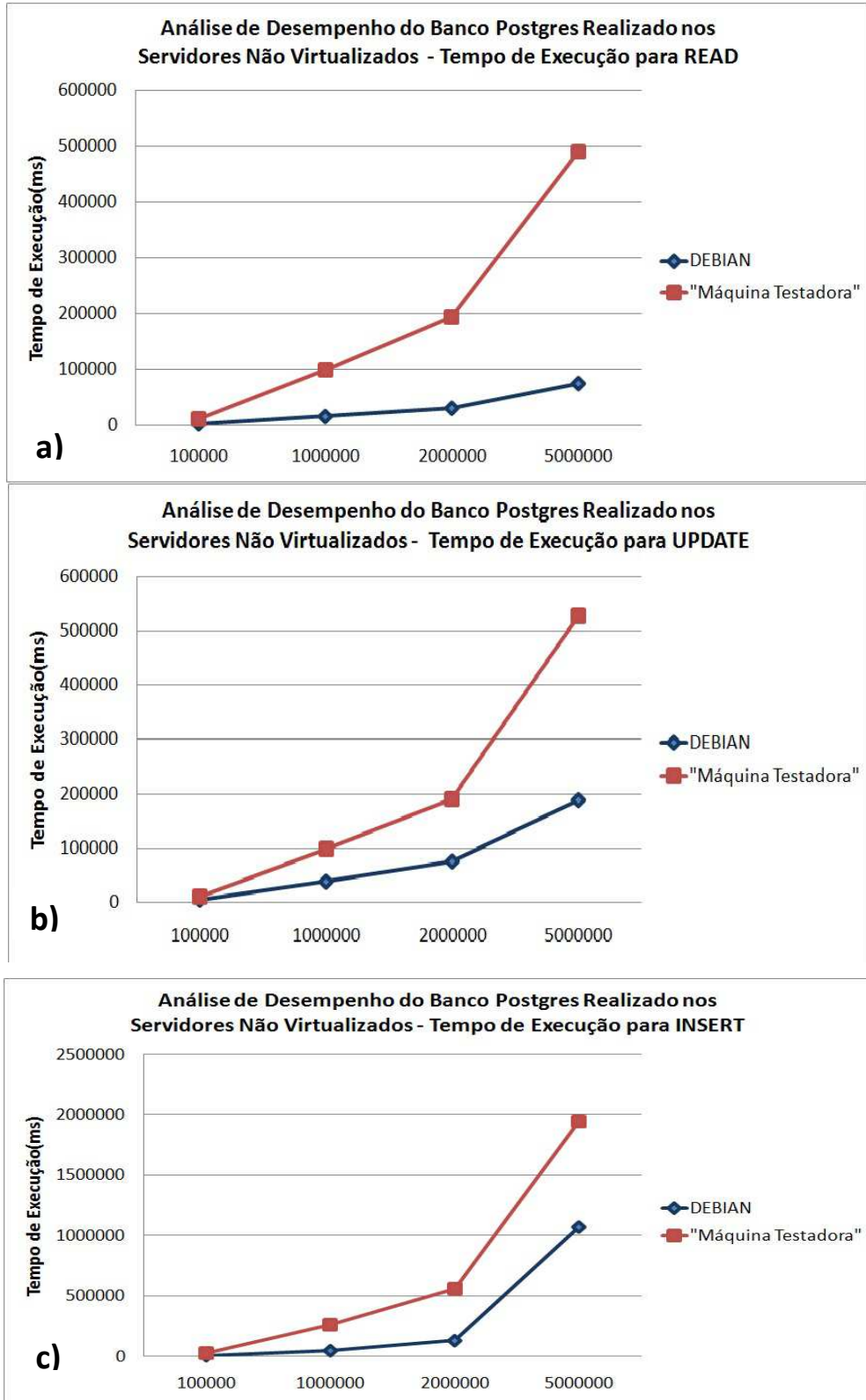


Figura 9 Tempo de execução em máquina não virtualizadas

(a) Insert, (b) Read, (c) Update

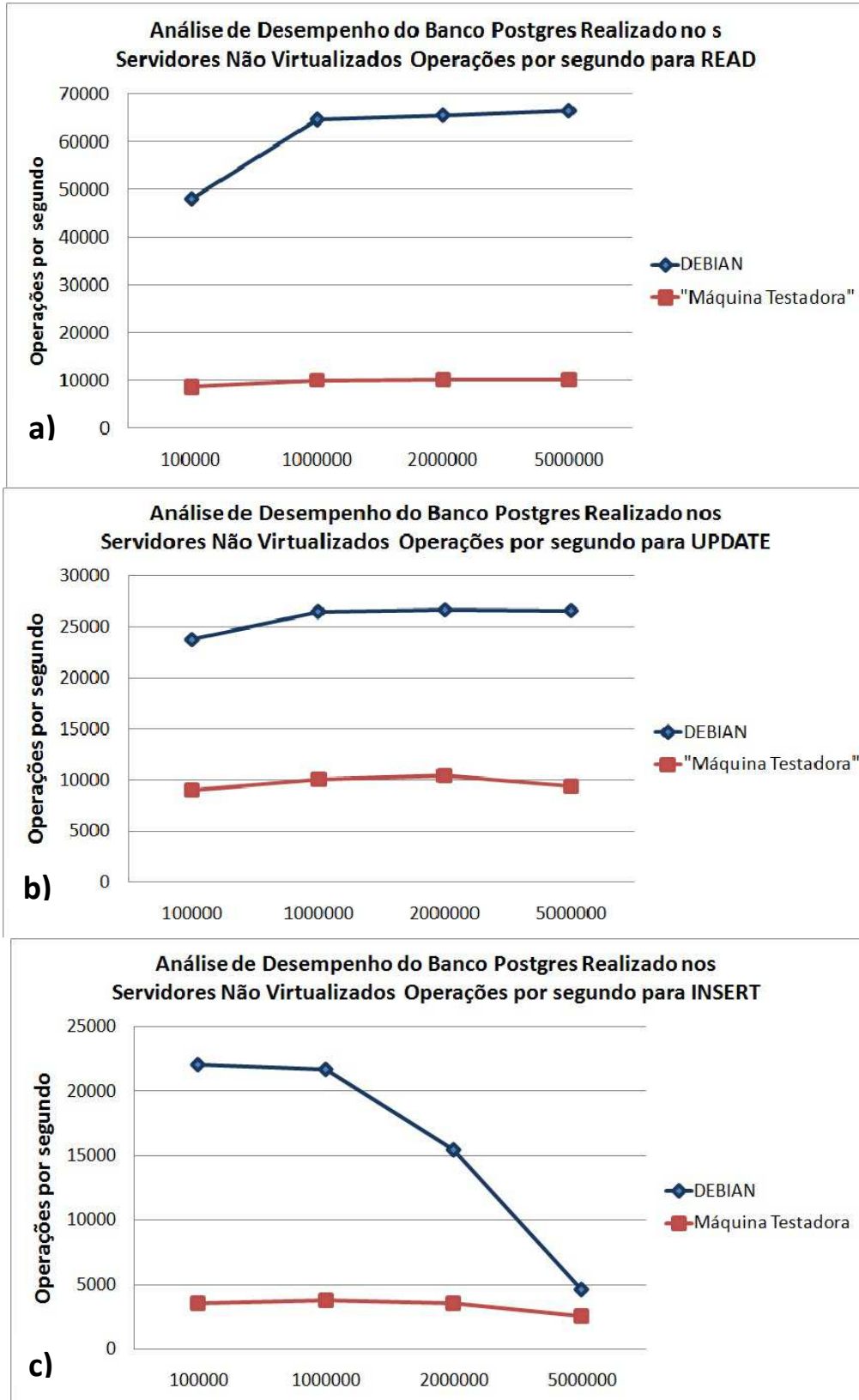
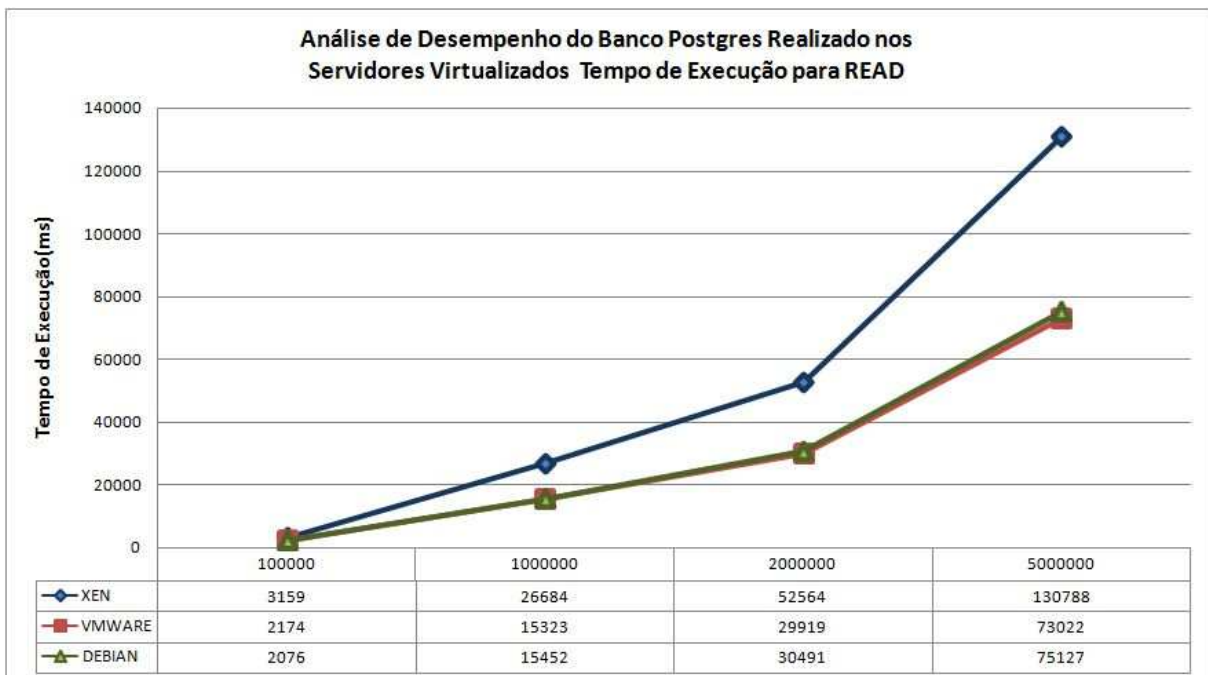


Figura 10 Número de Operações por segundo em máquina não virtualizadas (a) Insert, (b) Read, (c) Update

Observando os resultados das Figuras 9 e 10, nota-se “Máquina Testadora” teve desempenho inferior em todos os testes realizados, tanto houve mais tempo para a execução dos testes quanto a capacidade de realizar um grande número de testes.

Na primeira abordagem ocorreu uma perda de desempenho, e devido a essa falta de capacidade da máquina testadora em enviar os testes de sobrecarga e optou-se por dar ênfase no servidor testando ele mesmo. Assim, considerando que a troca de mensagens no mesmo *hardware* não influenciou significativamente nos resultados de desempenho.

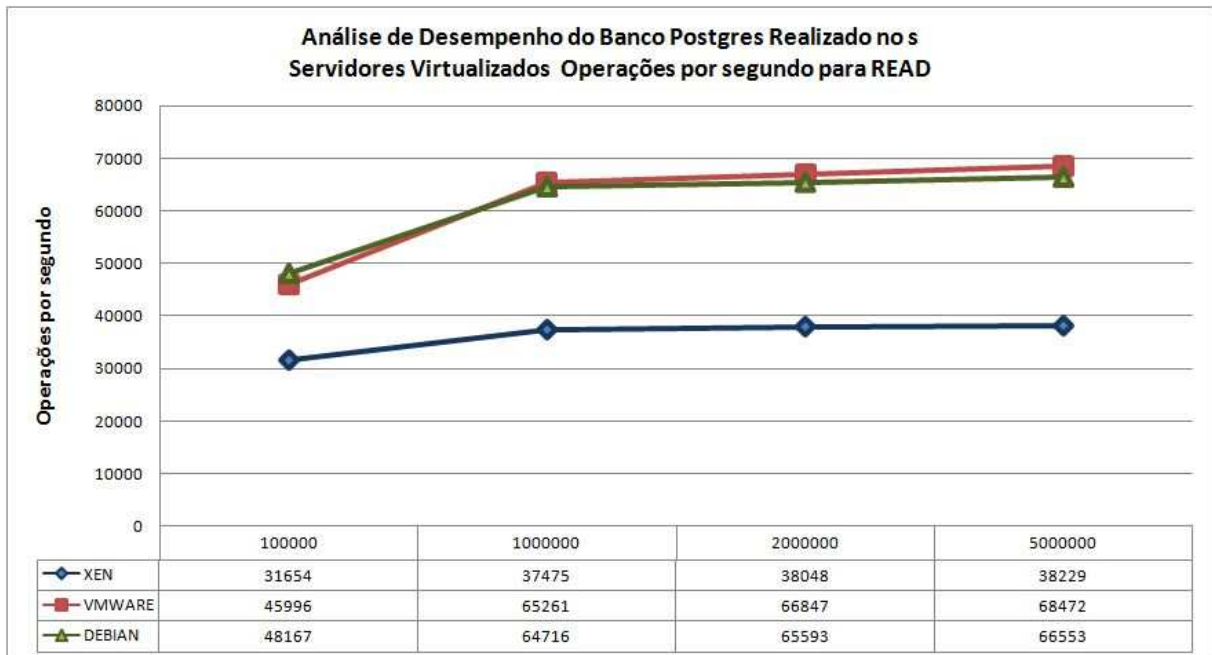
Analisando as 3 operações de Banco de Dados (Read, Insert e Update) foi possível quantificar e apresentar a diferenças sobre o "tempo de execução" das sobrecargas em relação às diferentes quantidades de operações e as diferenças das VM referente as "operações por segundo". Iniciou-se primeiramente pela operação READ conforme pode ser observado na Figura 11.



**Figura 11 Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Tempo de Execução para READ.**

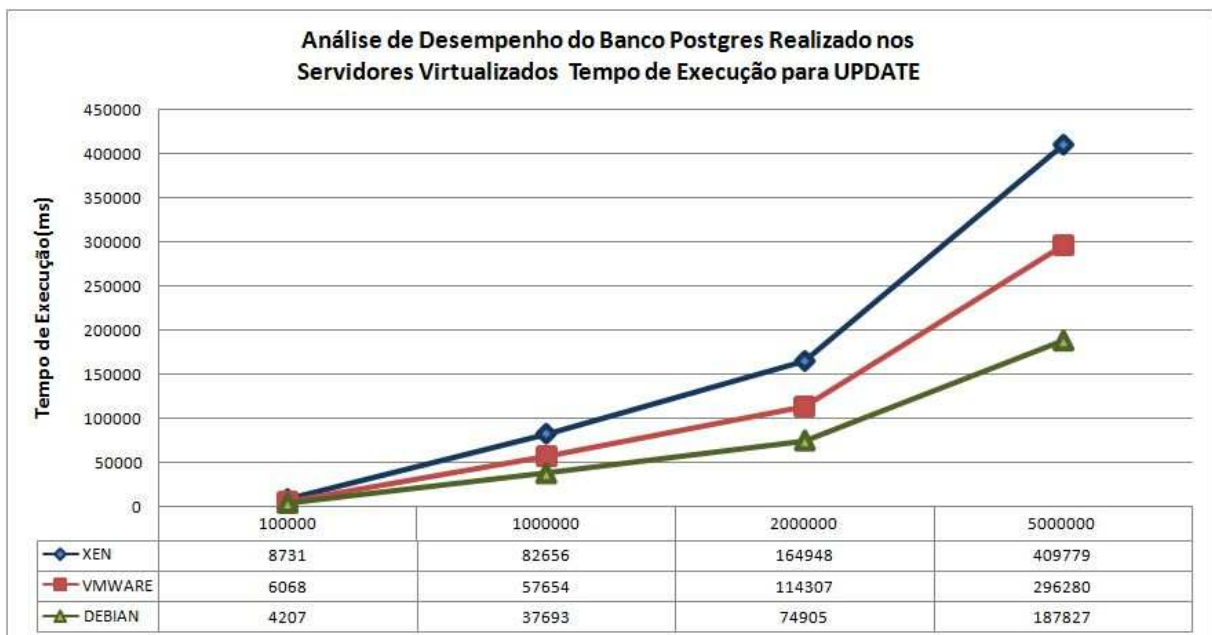
Observando os resultados obtidos, Figura 11 sobre READ, e comparados com as duas máquinas virtuais (Vmware e XEN) e o Sistema Operacional Linux (DEBIAN) é possível observar que no Tempo de Execução, o tempo gasto para a realização da sobrecarga da operação READ foi muito superior no virtualizador Xen

em relação a máquina não virtualizada e o VmWare, as quais tiveram resultados bem próximos.



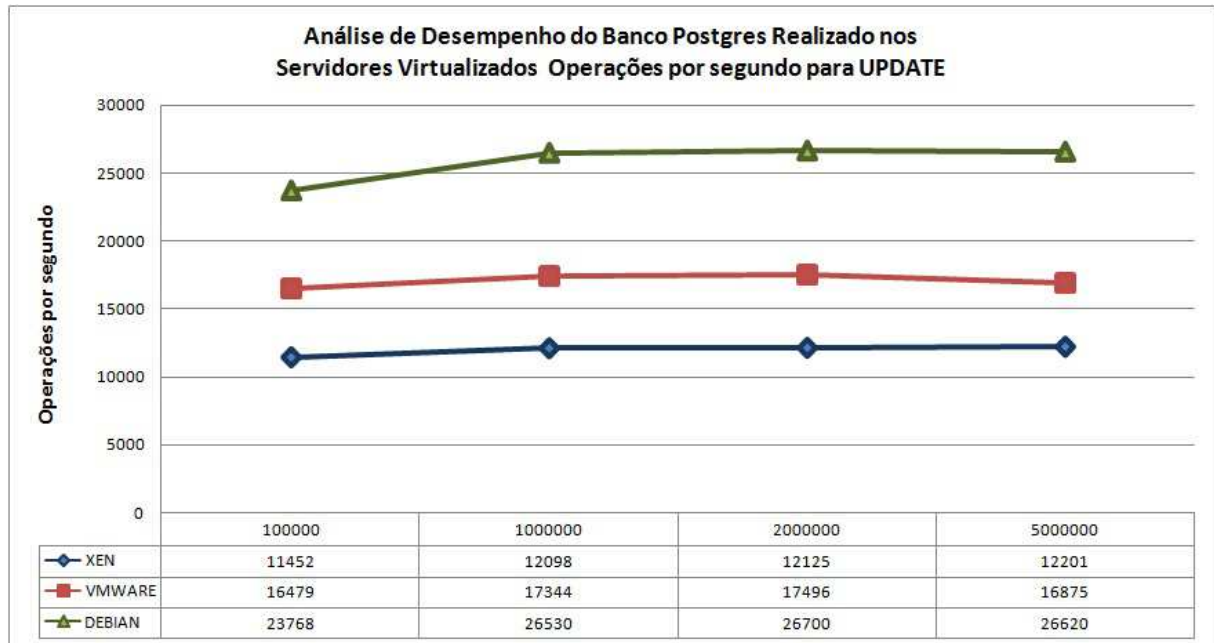
**Figura 12** Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Operações por Segundo para READ.

O VmWare e o Debian na raiz tiveram quase o mesmo desempenho de operações por segundo, enquanto o Xen teve um desempenho inferior. Mostrado na Figura 12.



**Figura 13** Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Tempo de Execução para UPDATE.

A Figura 13, mostra os resultados de Tempo de Execução para as operações de UPDATE.



**Figura 14** Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Operações por Segundo para UPDATE.

Como mostra a Figura 14, o Debian na raiz manteve a média de 26000 operações por segundo, o VmWare com 17 mil, Xen com 12 mil.

A Figura 15, mostra o tempo de execução para as sobrecargas de INSERT.



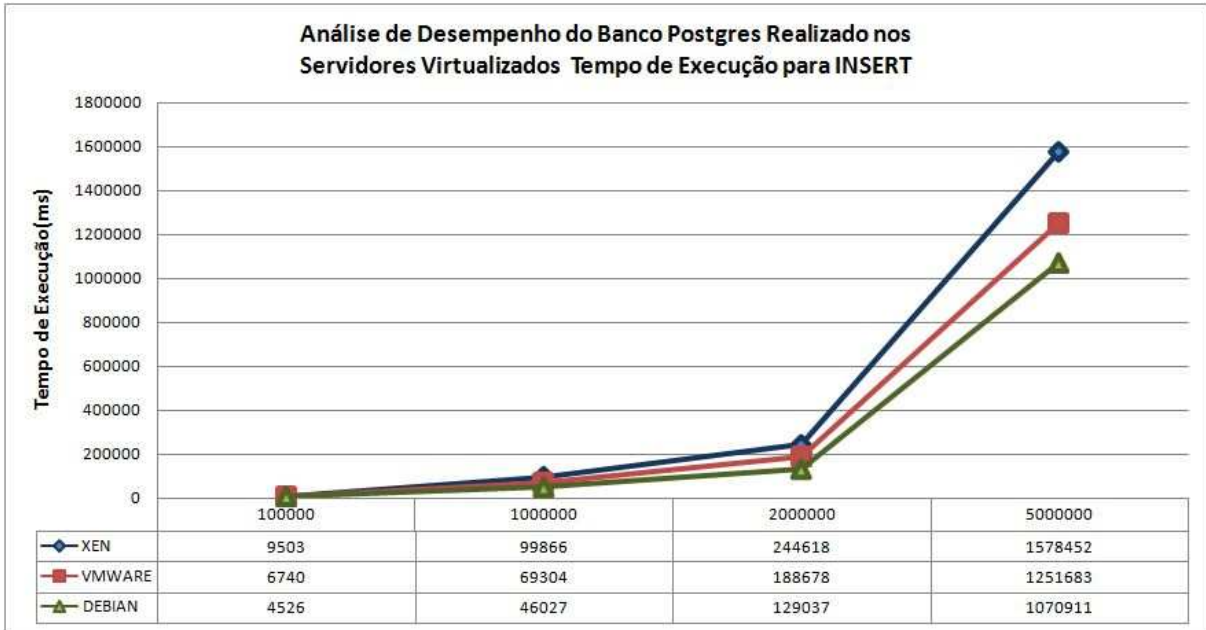


Figura 15 Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Tempo de Execução para INSERT.

A Figura 16, mostra os resultados para as operações por segundo para INSERT.

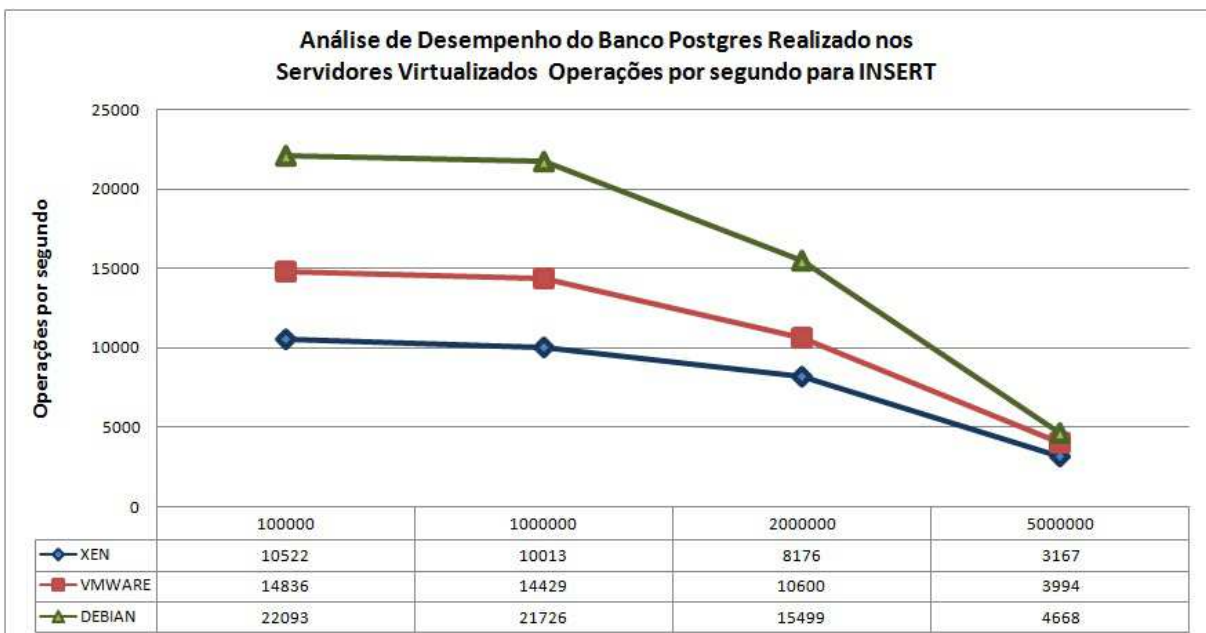


Figura 16 Análise de Desempenho do Banco Postgres Realizado nos Servidores Virtualizados Operações por segundo para INSERT.

O teste do *insert* quando ultrapassa a marca das duas milhões de operações, mostra a redução drástica do seu desempenho, mantendo-se com uma média de quatro mil operações por segundo, sendo que no começo do teste se manteve em treze mil operações por segundo.

Os resultados obtidos permitem concluir que o virtualizador VmWare teve o melhor desempenho comparado com o Xen. Os resultados obtidos com o virtualizador VmWare foram muito próximos aos testes feitos em uma máquina não virtualizada.

Embora a paravirtualização (Xen) apresente uma de desempenho significativo frente à virtualização total (VMware), devido a arquitetura do servidor suportar a virtualização e o Sistema Operacional ser x64, favorecendo assim a virtualização total, pois evita os possíveis erros decorrentes de uma arquitetura x86.

## 5 Conclusão

Nesse trabalho foi possível reafirmar as conclusões obtidas por Minhas, 2008 onde o autor defende a utilização de Banco de Dados em servidores virtualizados, havendo mais vantagens do que desvantagens em se utilizar a técnica de virtualização.

Há superioridade do virtualizador VmWare utilizando a virtualização total em todos os testes realizados comparado ao Xen (paravirtualização), isso por causa da favorável arquitetura do Sistema Operacional x64 e do servidor dar suporte à virtualização. Os resultados do Debian sem virtualização para o Tempo de Execução tiveram um desempenho maior de 8% comparado ao VmWare e nas maiores sobrecargas chegou a 15%. Os resultados são mostrados nas Figuras 17 e 18 para o Tempo de Execução e Operações por Segundo.

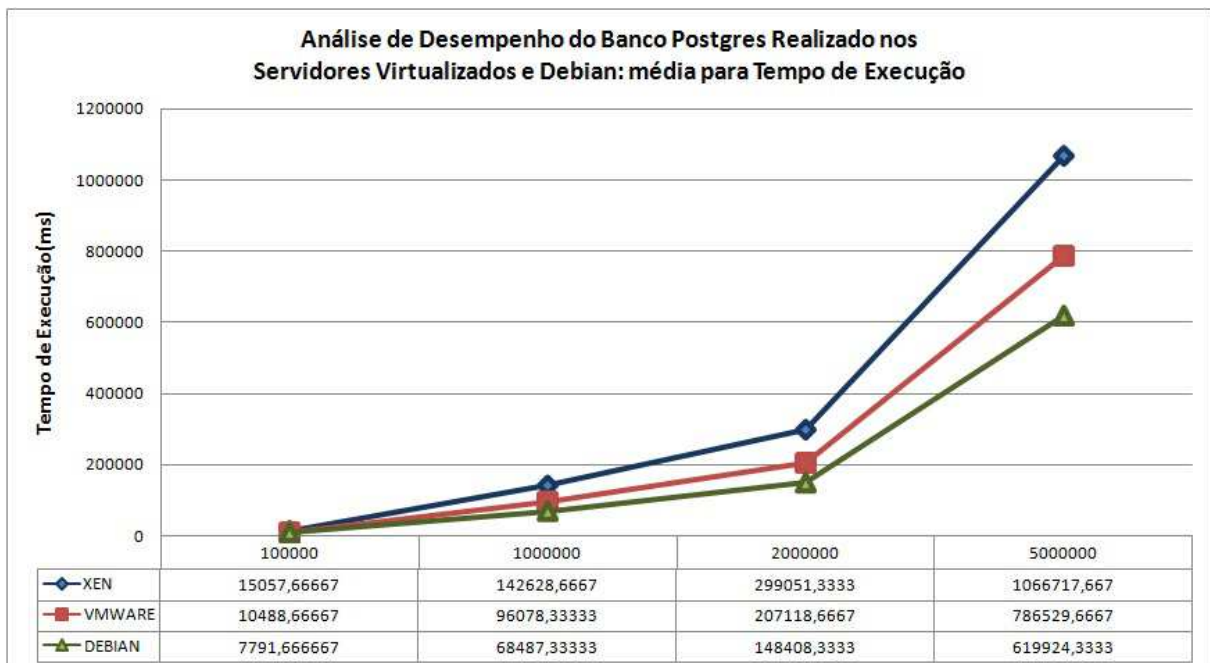
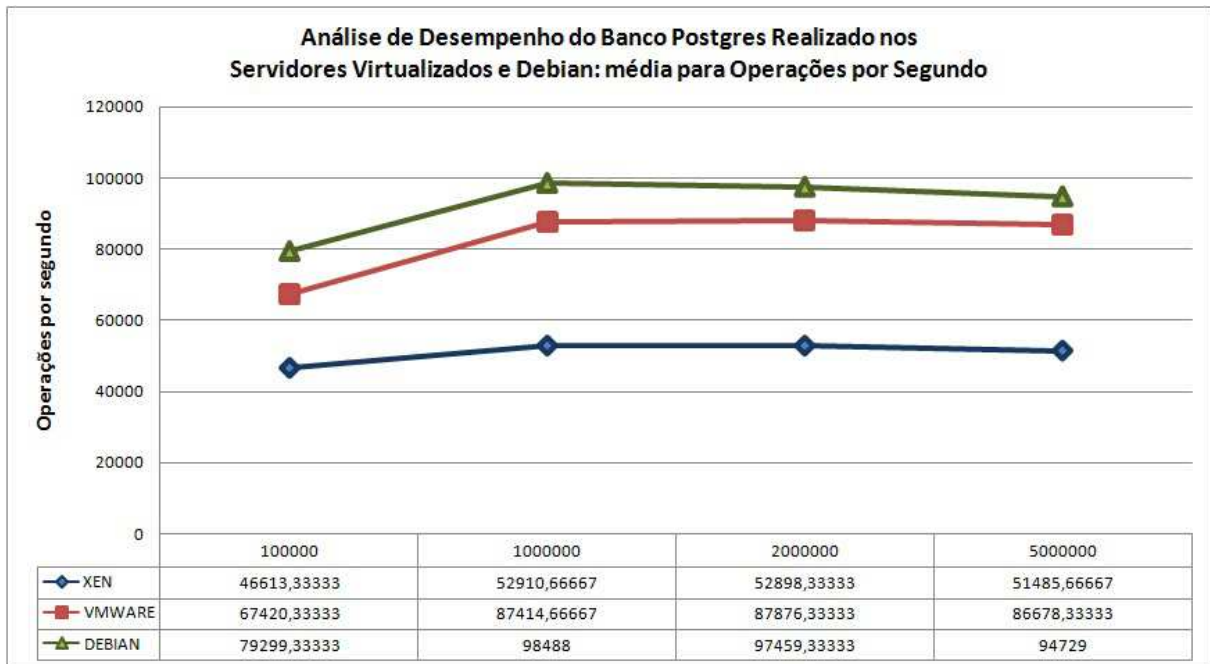


Figura 17. Média para Tempo de Execução



**Figura 18 Média para Operações por Segundo**

Os resultados para Operações por Segundo mostraram superioridade do virtualizador VmWare comparado ao Xen e uma perda de apenas 7% comparado ao servidor não virtualizado.

Através do levantamento bibliográfico foi possível concluir que os erros de paginação os quais são costumeiramente encontrados em Banco de Dados Virtualizados podem ser ajustados com a configuração do buffer de memória. Em alguns casos os ajustes podem ser implementados sem muito esforço, em outros, é necessária uma reengenharia completa da aplicação. Esse trabalho concluiu que o problema em virtualizar um Banco de Dados não está na técnica da virtualização em si, mas na necessidade de ajustes os quais devem ser analisados caso a caso. Esse trabalho também reforça o uso de Banco de Dados em servidores virtualizados.

É relevante que trabalhos futuros utilizando outros virtualizadores sejam realizados, para que se otimize a comparação entres os virtualizadores, facilitando a escolha de desenvolvedores.

É de grande consideração que se descubra as razões pelas quais os testes de *insert* se mostraram com um resultado final semelhantes para todos os testes, com uma redução significativa de desempenho.

## Referências Bibliográficas

ABOULNAGA, A; SALEM, K; SOROR. A, A. **Database Virtualization: A New Frontier for Database Tuning and Physical Design**. University of Waterloo. p 1 – 7, 2007.

ANASTASOPOULOS, A., **Memory Virtualization in Database Systems**, Master of Science School of Informatics. University of Edinburgh, August 2008.

BENEDICTO, M; BEZERRA, A; BOSCARIOLI, C. DELMIRO, G. **Uma reflexão sobre Banco de Dados Orientados a Objetos**. Universidade Estadual do Oeste do Paraná. p 1 - 12, 2006.

CAMPOS, V; VASCONCELOS B. A. **Teste de desempenho em Desktops Virtuais**. Centro Universitário Ritter dos Reis. p 10 - 15, 2010.

COSTA, C. L. L. da; **Estudo de Caso dos Siftwares Para Virtualização no IESAM**, Instituto de Estudos Superiores da Amazônia, 2010.

JIN, S. **VMware VI and vSphere SDK: Managing the VMware Infrastructure and vSphere**. Prentice Hall, 2009.

LIMA, I; SANTOS, I. **Virtualização em Servidores de Banco de Dados**. [S.l]. p 1 - 11, 2010.

MINHAS, U.F.; YADAY, J.; ABULNADA, A.; SALEM, K. **Database systems on virtual machines: How much do you lose?**, Data Engineering Workshop, 2008. ICDEW 2008. IEEE 24th International Conference on , vol., no., pp.35,41, 7-12 April 2008.

ORACLE, 2011 **UserManual**, <http://www.VirtualBox.org/manual/UserManual.html> acessado em 29/06/2014

PAPE C., TROMMER R. **Monitoring VMware-based Virtual Infrastructures with OpenNMS** – 2012.

PERKUSICH, L. B. M; TURNELL, F.Q.V.M. de; PERKUSICH, A. **Modelagem de Banco de Dados em Tempo-real**. Universidade Federal da Paraíba, 1999.

POLLON, V. **Virtualização de servidores em ambientes heterogêneos e distribuídos**. Universidade Federal do Rio Grande do Sul. p 20 - 35, 2008.

POSTGRESQL . Site Oficial 2005. <http://www.postgresql.org/> Acessado em: 27/06/2014.

SANTOS, F, C. **Ambiente de Virtualização: Uma Análise de Desempenho**. Universidade Federal de Lavras. p 20 – 35, 2011.

Sistemas Operacionais. <http://www2.ic.uff.br/~aconci/SistemasOperacionais.html> Acessado em 16/04/2014.

TPC-H: An Ad-hoc, Decision Support Benchmark. <http://www.tpc.org/tpch/>. Acessado em 19/11/2013

VERAS, M. **Virtualização Componente Central do Datacenter**. ed 2, Brasport, p 30 - 130, 2011.

VMware, Inc. 2011. VMware vSphere Basics. VMware, Inc., URL: <http://pubs.vmware.com/vsphere-50/topic/com.vmware.ICbase/PDF/vsphere-esxi-vcenter-server-50-basics-guide.pdf>. Acessado 17/11/2013

Xen Archives. <http://www.xenproject.org/downloads/xen-archives.html>. Acessado em 19/11/2013.