

RICARDO JOSÉ PFITSCHER

**ANÁLISE DE CUSTO-BENEFÍCIO DE VIRTUALIZAÇÃO PARA
CONSOLIDAÇÃO DE SERVIDORES**

JOINVILLE - SC

2008

UNIVERSIDADE DO ESTADO DE SANTA CATARINA – UDESC
CENTRO DE CIÊNCIAS TECNOLÓGICAS – CCT
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO - DCC
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO - BCC

RICARDO JOSE PFITSCHER

**ANÁLISE DE CUSTO-BENEFÍCIO DE VIRTUALIZAÇÃO PARA
CONSOLIDAÇÃO DE SERVIDORES**

Trabalho de conclusão de curso submetido à
Universidade do Estado de Santa Catarina
como parte dos requisitos para a obtenção do
grau de Bacharel em Ciência da
Computação.

Orientador: Dr. Rafael Rodrigues
Obelheiro

JOINVILLE – SC

2008

RICARDO JOSÉ PFITSCHER

**ANÁLISE DE CUSTO-BENEFÍCIO DE VIRTUALIZAÇÃO
PARA CONSOLIDAÇÃO DE SERVIDORES**

Este Trabalho de Conclusão de Curso foi julgado adequado para a obtenção do título de Bacharel em Ciência da Computação e aprovada em sua forma final pelo Curso de Ciência da Computação Integral do CCT/UDESC.

Banca Examinadora:

Orientador:

Dr. Rafael Rodrigues Obelheiro

Membro:

MsC. Cinara Menegazzo

Membro:

Dr. Maurício Aronne Pillon

Joinville, Dezembro de 2008.

Aos meus familiares e amigos.

AGRADECIMENTOS

Para desenvolvimento deste trabalho algumas pessoas se destacam e tornam se merecedoras de reconhecimento, dentre elas podemos citar professores, amigos e familiares:

À minha mãe Elisete, pelo carinho, atenção e educação que a mim foram dedicados, somente com estes fui capaz de concluir mais esta etapa em minha vida.

Ao meu pai Paulo César, pelas palavras de força, pelo orgulho demonstrado, pela educação, e pelo carinho a mim dedicado.

Ao meu irmão Paulo Henrique, pelo incentivo, pelo apoio, e principalmente pelo afeto dedicado.

Ao meu irmão Pedro, pela atenção, pelo afeto, e principalmente por suportar a ausência do irmão e amigo.

Aos meus familiares pelo apoio e dedicação para com minha pessoa, e a força com a qual pude enfrentar este desafio.

Aos meus amigos que me animaram nos momentos de desânimo.

À minha namorada, Helena, pela paciência de suportar a ausência do namorado durante a execução deste trabalho, e pelas horas destinadas a ajudar a concepção do mesmo.

Ao meu professor Rafael pela orientação no desenvolvimento deste trabalho.

Aos demais professores da faculdade pelos ensinamentos repassados, estes fundamentais para o desenvolvimento deste trabalho.

SUMÁRIO

RESUMO	11
ABSTRACT	12
1. INTRODUÇÃO	14
1.1 Objetivos.....	17
1.2. Metodologia.....	18
1.3. Estrutura do Trabalho	19
2. VIRTUALIZAÇÃO	21
2.1 Introdução.....	21
2.2 Conceituação	23
2.3 Camadas de Virtualização	25
2.3.1 Camada de Virtualização para Aplicação	25
2.3.2 Camada de Sistema Operacional	26
2.3.3 Camada de Hardware	27
2.3.4 Paravirtualização	31
2.4 Camadas em Função da Consolidação	33
3. FERRAMENTAS.....	35
3.1 Ferramentas para a camada de sistema operacional	35
3.1.1 VMWare Workstation	36
3.1.2 VirtualBox.....	37
3.2 Ferramentas para a camada de hardware	39
3.2.1 Denali	39
3.2.2 Xen	40
3.3 Comparação entre Ferramentas	41
4. MÉTRICAS.....	45
4.1 Funcionamento de um Servidor Web	46

4.2 Requisitos	48
4.3 Definindo métricas	49
4.3.1 Métricas de avaliação interna.....	50
4.3.2 Métricas de avaliação externa.	50
4.4 Ferramentas	51
4.5 Definição dos testes	53
4.5.1 Ambiente de testes 1	55
4.5.2 Ambiente de testes 2	56
4.5.3 Ambiente de testes 3	57
4.6 Considerações.....	58

5. RESULTADOS..... 59

5.1 Metodologia de execução	59
5.2 Comparação entre XEN e máquina nativa.	61
5.3 Desempenho de rede nos diferentes ambientes	69
5.3.1 Cenário com um cliente.....	70
5.3.2 Cenário com dois clientes	71
5.3.3 Cenário com três clientes	73
5.4 Desempenho do servidor web nos diferentes ambientes	75
5.4.1 Cenário com um cliente.....	76
5.4.2 Cenário com dois clientes	85
5.4.3 Cenário com três clientes	94
5.5 Análise de custo-benefício da virtualização	104

6. CONCLUSÃO 108

7. ANEXOS 110

7.1 Configurando ambiente de virtualização.....	110
7.2 Criando as máquinas virtuais.....	111
7.3 Arquivo de configuração das máquinas virtuais.....	112
7.4 Plano de TCC aprovado.....	114

REFERÊNCIAS BIBLIOGRÁFICAS..... 123

LISTA DE FIGURAS

Figura 1: Comparativo de ambiente virtualizado com ambiente não virtualizado.....	15
Figura 2: Arquitetura Padrão.	22
Figura 3: Exemplo de Virtualização e Máquina Virtual [Adaptada de SMITH, 2005].	23
Figura 4: Virtualização na camada de aplicação.	25
Figura 5: Virtualização na camada de SO.	27
Figura 6: Virtualização Completa.....	28
Figura 7: Exemplos de virtualização [MAZIERO, 2008].	30
Figura 8: Exemplo de paravirtualização utilizando Xen [CROSBY, 2006].....	33
Figura 9: Execução de um SO Windows XP sobre um SO Ubuntu Linux..	38
Figura 10: Xen, Múltiplas plataformas Linux e NetBSD [XEN, 2008].	41
Figura 11: Funcionamento básico de um servidor WEB.....	47
Figura 12: Testes sobre o <i>hardware</i>	54
Figura 13: Ambiente de testes 1.	55
Figura 14: Ambiente de testes 2.	56
Figura 15: Ambiente de testes 3	57
Figura 16: Desempenho sistema de arquivos	61
Figura 17: Desempenho da criação de processos	63
Figura 18: Desempenho <i>Pipe</i>	65
Figura 19: Desempenho <i>signal handler</i>	66
Figura 20: Desempenho para execução de processos.....	67
Figura 21: Desempenho sistema operacional.	68
Figura 22: Transferência total para 1 cliente	70
Figura 23: Consumo de banda para 1 cliente	71
Figura 24: Transferência total para 2 clientes	72
Figura 25: Consumo de banda para 2 clientes.....	72
Figura 26: Transferência total para 3 clientes	73
Figura 27: Consumo de banda para 3 clientes.....	74
Figura 28: Tempo total <i>ApacheBench</i> com 1 cliente.....	76
Figura 29: <i>ApacheBench</i> : requisições suportadas para 1 cliente	77
Figura 30: <i>ApacheBench</i> - Taxa de transferência para 1 cliente	78
Figura 31: <i>Autobench</i> : taxa de requisição para 1 cliente.....	79
Figura 32: <i>Autobench</i> : Taxa de conexão para 1 cliente.....	80
Figura 33: <i>Autobench</i> : taxa de resposta para 1 cliente.	81
Figura 34: <i>Autobench</i> : Tempo de resposta para 1 cliente.	82
Figura 35: <i>Autobench</i> : Consumo de banda para 1 cliente.....	83
Figura 36: <i>Autobench</i> : número de erros para 1 cliente.....	84
Figura 37: Saturação do servidor para 2 clientes.....	85
Figura 38: <i>ApacheBench</i> : Tempo total dos testes para 2 clientes	86
Figura 39: <i>ApacheBench</i> : Requisições por segundo para 2 clientes	87
Figura 40: <i>ApacheBench</i> : Consumo de banda para 2 clientes.....	88
Figura 41: <i>Autobench</i> : Taxa de requisição para 2 clientes	89

Figura 42: <i>Autobench</i> : Taxa de conexão para 2 clientes	90
Figura 43: <i>Autobench</i> : Taxa de resposta para 2 clientes	90
Figura 44: <i>Autobench</i> : Tempo de resposta para 2 clientes	91
Figura 45: <i>Autobench</i> : Consumo de banda para 2 clientes	92
Figura 46: <i>Autobench</i> : número de erros para 2 clientes	93
Figura 47: Saturação do servidor para 3 clientes.....	95
Figura 48: <i>ApacheBench</i> : Tempo para realizar testes para 3 clientes	96
Figura 49: <i>ApacheBench</i> : Requisições por segundo para 3 clientes	97
Figura 50: <i>ApacheBench</i> : Consumo de banda para 3 clientes.....	98
Figura 51: <i>Autobench</i> : Taxa de requisição para 3 clientes	99
Figura 52: <i>Autobench</i> : Taxa de conexão para 3 clientes	100
Figura 53: <i>Autobench</i> : Taxa de resposta para 3 clientes	100
Figura 54: <i>Autobench</i> : Tempo de resposta para 3 clientes	101
Figura 55: <i>Autobench</i> : Consumo de banda para 3 clientes	102
Figura 56: <i>Autobench</i> : Número de erros para 3 clientes	103

LISTA DE TABELAS

Tabela 1: Comparativo entre ferramentas de virtualização.....	43
---	----

RESUMO

Este trabalho de conclusão de curso tem por objetivo promover um estudo referente ao uso de múltiplos servidores em um mesmo *hardware* com auxílio de máquinas virtuais. O foco do trabalho está em verificar aspectos como a perda de desempenho, e definir o custo/benefício de uma implantação dessa natureza. Para desenvolvimento deste trabalho também foi necessária a utilização de um ambiente de virtualização específico para serem realizados os testes. Este comparativo foi realizado com auxílio de ferramentas que forneceram dados e informações referentes aos experimentos propostos no decorrer do trabalho.

PALAVRAS-CHAVE: Máquina Virtual, Fornecimento de Serviços, Múltiplos Servidores, Redução de Custos, Virtualização, Desempenho.

ABSTRACT

This work aims to promote a study concerning the use of multiple servers on a single hardware with the help of virtual machines. The intention is to verify aspects such as loss of performance, and define the cost / benefit of a deployment of this nature. For development of this work will also be required to use a specific environment of virtualization to be performed the tests. This comparison has been conducted with the help of tools that has provided data and information relating to the experiments that has been proposed in the course of work.

KEYWORDS: Virtual Machines, Provision of Services, reduce costs, Virtualization, Performance.

*“Quanto maior for a batalha, maior será a
glória do vitorioso”*

Ricardo Jose Pfitscher

1. INTRODUÇÃO

A utilização de estruturas informatizadas para armazenamento de informações teve sua explosão a partir da década de 90, e cada vez mais tem aumentado o número de serviços necessários à sociedade, como por exemplo, o armazenamento de arquivos, o acesso à *web* e o compartilhamento de arquivos. Para suprir essa demanda de serviços, diversos artefatos informatizados são apresentados como soluções, entre estes pode-se encontrar diversos tipos de servidores [STALLINGS, 2005].

Servidores são máquinas físicas que tem como função principal prover serviços específicos aos usuários, ou até centralizar serviços, a fim de atingir diversos fatores, como segurança e redução de custos. Um exemplo seria o uso de um servidor para prover um serviço de arquivos centralizado, mantendo em apenas uma máquina física, dados que poderiam estar repetidos em diversas estações. Outro serviço que pode ser considerado é um *proxy* de acesso à *web*, que permite que um administrador tenha maior controle sobre os acessos externos realizados pelos usuários da sua rede interna.

Com o incremento na adoção desse tipo de solução, alguns desafios devem ser levados em consideração. Quanto maior a quantidade de serviços e servidores disponibilizados, maior é a complexidade de gerenciamento e maiores são as necessidades de utilização de plataformas e sistemas operacionais distintos [FIGUEIREDO, 2005].

Além disso, é comum que serviços distintos sejam implementados fisicamente em servidores separados. As razões pelas quais isso acontece incluem preocupações de segurança e confiabilidade (isolamento de serviços) e a diversidade de sistemas operacionais requeridos por diferentes serviços. Entretanto, esta forma tem se tornado cada vez mais cara. Estima-se que o custo de cada servidor no *data center* de uma empresa é de US\$ 10.000,00 por ano, incluindo gastos com administração, manutenção, energia elétrica, treinamentos,

atualizações, entre outros gastos. Uma possível solução para tais custos seria a utilização de virtualização a fim de diminuir a quantidade de *hardware* utilizada [CROSBY, 2006].

A virtualização consiste na inserção de uma camada de *software* entre o sistema operacional e o *hardware* que fornece ao usuário uma abstração dos recursos físicos da máquina, chamada de máquina virtual. Entre outras coisas, isto permite que sejam utilizadas diversas imagens do sistema operacional (SO) em um único sistema, sendo que cada SO possui a impressão de interagir diretamente com o *hardware*. O mecanismo responsável por propiciar estes acessos é o ambiente de virtualização, que distribui os recursos necessários de acordo com a necessidade das máquinas virtuais. Entre os recursos gerenciados pelo ambiente de virtualização encontram-se a unidade de gerenciamento de memória, os dispositivos de entrada e saída, e os controladores de acesso direto à memória [CROSBY, 2006].

Na consolidação de servidores, cada servidor lógico pode executar em uma máquina virtual separada como se tivesse todos os recursos de *hardware* à sua disposição. Isto facilita a consolidação, uma vez que múltiplas unidades de máquinas virtuais podem ser executadas, disponibilizando múltiplos serviços, ficando o ambiente de virtualização responsável por gerenciar os recursos compartilhados pelas máquinas virtuais. A figura 1 apresenta um exemplo desta aplicação [FRASER, 2004]:

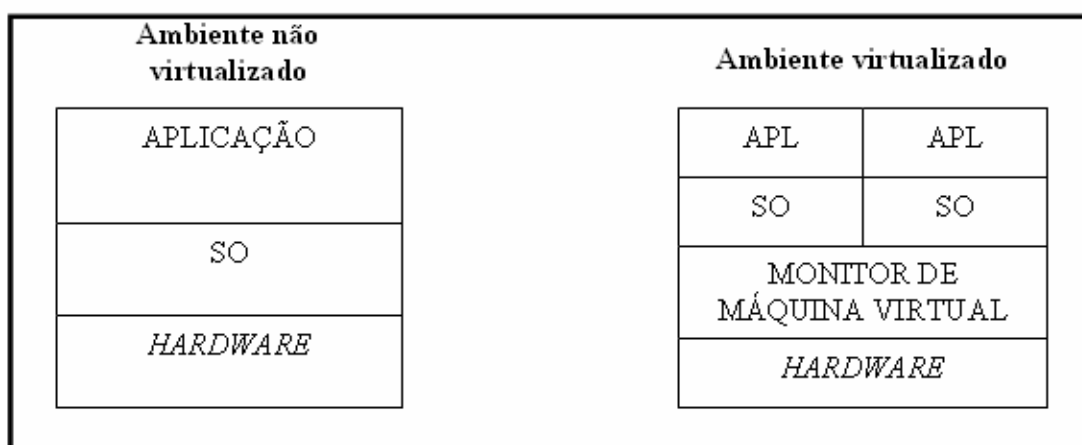


Figura 1: Comparativo de ambiente virtualizado com ambiente não virtualizado

Basicamente, em um ambiente onde não é utilizada a virtualização tem-se a utilização de um *hardware*, um sistema operacional e uma ou mais aplicações. Nesse tipo de ambiente, quando se deseja usar diferentes sistemas operacionais é preciso usar servidores físicos distintos. Quando se tem consolidação de servidores através de um ambiente virtualizado, pode-se utilizar diversos sistemas operacionais em um único servidor físico, e assim não seria necessária a utilização de um novo *hardware* para cada serviço [VOGELS, 2008].

Tendo em vista que a utilização de servidores virtuais sugere uma diminuição do custo relacionado à manutenção de múltiplos servidores, é necessário avaliar o real custo/benefício desta utilização: até que ponto a redução de custos irá igualar a balança com relação à perda de desempenho? Este é o principal problema a ser analisado neste trabalho, descobrir até que ponto pode-se utilizar a virtualização para que se tenha uma relação custo/benefício aceitável.

Como instrumento de análise, serão feitos testes comparativos entre serviços executando diretamente em máquina física (não virtualizada) e serviços executando em máquinas virtuais (virtualizadas). Esta virtualização dar-se-á de diferentes formas, utilizando um serviço unicamente virtualizado e também a utilização de várias réplicas do mesmo serviço virtualizado. É importante garantir também que os testes realizados com a máquina física e os testes realizados com as máquinas virtuais sejam feitos precisamente no mesmo ambiente, ou seja, utilizem o mesmo *hardware* e o mesmo ambiente de rede, mantendo a consistência dos dados recolhidos.

Com relação ao ambiente de testes, inicialmente teve-se a idéia de utilizar uma máquina física como provedora do serviço, onde este fez-se uso de interface de rede e CPU, um exemplo para uso destes aspectos é um servidor *web*, e duas ou três máquinas clientes realizando acessos a este serviço em uma rede local. Posteriormente, a mesma máquina física (*hardware*) servidora propiciou o uso do ambiente virtualizado.

A definição do ambiente de virtualização explorado foi direcionada ao uso de *software* livre, com o intuito de eliminar restrições impostas por licenças

proprietárias. Porém alguns ambientes proprietários também tiveram de ser analisados, a fim de se realizar um levantamento relacionado aos níveis de desempenho e compatibilidade das diversas ferramentas. Dentre as possibilidades levantadas cita-se: Xen, VirtualBox e VMWare ESX Server [FIGUEIREDO, 2005]. Dentre as características que se fazem necessárias para a definição das possibilidades foi destacado a capacidade de que a ferramenta tem de virtualizar sistemas operacionais distintos, a capacidade de virtualizar plataformas distintas, bem como foi analisada a complexidade do uso da ferramenta.

Os dados colhidos nos testes foram avaliados a fim de verificar qual o real emprego do conceito de virtualização, levando em consideração principalmente o custo e o desempenho das diferentes abordagens, viabilizando ou não o uso de máquinas virtuais para múltiplos servidores.

Algumas dificuldades foram encontradas para desenvolvimento deste trabalho, uma delas é a implantação dos serviços, sendo que se teve buscar um conhecimento específico sobre a utilização do ambiente de virtualização, pois somente através deste pode-se realizar uma implementação correta do serviço, sem falhas provenientes do mau uso do ambiente de virtualização utilizado. Outra dificuldade foi a definição das métricas e experimentos utilizados para realizar os comparativos referentes a custo e desempenho, pois existem diversas métricas disponíveis, contudo, fez-se necessário selecionar as mais representativas para o problema em estudo.

1.1 Objetivos

O objetivo principal deste trabalho é realizar um estudo referente à implantação de múltiplos servidores através de máquinas virtuais, visando determinar a relação custo/benefício desse tipo de solução. Para atingir o objetivo geral, alguns itens específicos devem ser explorados, primeiramente deve-se

realizar um estudo referente às tecnologias de virtualização existentes, a fim de se definir qual será utilizada e sobre esta realizar um estudo mais aprofundado.

Outro item importante é a questão relacionada ao tipo de servidor e serviço a ser testado. É importante a definição e a implantação de um serviço específico para realização dos testes. Deve-se definir também, quais as métricas que serão utilizadas para comparar as diferentes situações analisadas.

A definição e elaboração dos testes é outro objetivo que deve ser realizado. É importante salientar quais os tipos de testes que serão realizados durante o processo de pesquisa, bem como o ambiente de sua execução.

Finalmente, tem-se a avaliação dos resultados e a obtenção da análise custo-benefício. Esta análise deve ser realizada com base nos dados obtidos nos testes, o resultado a ser apresentado deve indicar qual a melhor situação de uso de virtualização para consolidação de servidores, mostrando qual a perda de desempenho em diferentes situações e definir se existe a possibilidade de se reduzir custos utilizando estes conceitos.

1.2. Metodologia

Para que os objetivos propostos sejam alcançados será necessário o cumprimento de algumas etapas de desenvolvimento. A primeira etapa consiste em avaliar as tecnologias atuais referentes à utilização de máquinas virtuais e realizar uma pesquisa bibliográfica com o intuito de encontrar fontes e trabalhos dessa área, bem como na área de servidores. Outro item importante a ser pesquisado é relacionado às medidas de desempenho a serem utilizadas no trabalho, juntamente com a busca de ferramentas que possam auxiliar neste processo.

Depois de realizados os estudos referentes à utilização de máquina virtual junto a servidores, um novo estudo deve ser realizado para levantar informações referentes ao uso de um ambiente de virtualização específico, bem como definir aspectos importantes como capacidade de uso e requisitos necessários.

Após o levantamento dos métodos de análise e os pontos a serem analisados, deve-se passar para a definição dos aspectos a serem avaliados. Primeiramente serão definidas as métricas a serem utilizadas como medida de desempenho. A seguir, será definida qual ferramenta será utilizada para coletar tais métricas. Esta definição virá após uma comparação das ferramentas disponíveis, sendo que pontos como facilidade de uso e diversificação de testes serão de alta relevância para seleção da mesma.

Terminadas as definições referentes a métricas e ferramentas utilizadas, deve se pensar nos experimentos a serem realizados, e o objetivo desta etapa é justamente defini-los.

A última etapa consiste em analisar os resultados obtidos nos experimentos, e com estes dados, definir as perdas ocorridas, ou não, sobre a utilização de máquina virtual para a aplicação de múltiplos servidores. Esta informação será necessária para apresentar uma perspectiva completa sobre a viabilidade do propósito apresentado no trabalho.

1.3. Estrutura do Trabalho

Este trabalho está dividido em oito capítulos, sendo que quatro deles são escritos com ênfase conceitual, um com ênfase prática, um como conclusão de trabalho, um como exemplificação de uso e um como discriminação de referências.

No capítulo 1 é realizada uma introdução sobre o assunto, abordando conceitos básicos necessários ao desenvolvimento do trabalho. Neste capítulo também são definidos os objetivos e a metodologia do mesmo.

No capítulo 2 tem-se o início da descrição conceitual referente à virtualização, sendo que são utilizados conceitos aprofundados sobre a utilização desta tecnologia, neste também ocorre uma classificação referente aos tipos definidos para cada característica.

O capítulo 3 descreve e compara as principais ferramentas de virtualização, com base na classificação discutida no capítulo anterior.

No capítulo 4 é feita a definição do serviço a ser avaliado, bem como os requisitos a serem avaliados para o mesmo. Depois de levantados os requisitos são definidas quais as métricas a serem avaliadas de acordo com esses requisitos, as ferramentas utilizadas para obtê-las, e os diferentes ambientes de testes que serão considerados.

No capítulo 5 são apresentados e discutidos os resultados obtidos nos testes, e é realizada a análise final do custo-benefício de virtualização para consolidação de servidores.

No capítulo 6 são descritas as considerações finais, que indicam um breve resumo do realizado, além de dificuldades encontradas e novas propostas de trabalho na área.

Por fim, têm-se os anexos, onde estão apresentados o plano inicial deste trabalho de conclusão de curso, e os passos utilizados para instalação da ferramenta de virtualização XEN, bem como são apresentados os arquivos de configuração das máquinas virtuais.

2. VIRTUALIZAÇÃO

Este capítulo tem como objetivo a definição dos conceitos relacionados à utilização da virtualização. Além do estudo referente aos conceitos principais serão analisadas as classificações dos mesmos, e ocorre neste capítulo a abordagem sobre as características de cada classificação.

2.1 Introdução

Antes de apresentar os conceitos e analisar as características específicas da virtualização, é necessário compreender o seu histórico. O surgimento das tecnologias de virtualização veio com a evolução dos sistemas operacionais. Os avanços nas tecnologias de sistemas operacionais e *hardware* permitiram que cada processo do sistema fosse atribuído a um único usuário, depois disto, na maioria dos processadores existentes foram adicionados os conceitos de modo usuário e modo supervisor. Esta última tecnologia permitia que alguns processos e instruções do sistema fossem protegidos dos usuários comuns e só pudessem ser executadas durante o modo supervisor, este utilizado pelo sistema operacional para enviar instruções privilegiadas, como por exemplo, o acesso a memória. Isto fez com que o sistema operacional pudesse compartilhar o uso da memória física para diferentes processos, assim múltiplas aplicações poderiam ser executadas ao mesmo tempo [CROSBY, 2006].

Todas estas melhorias foram aplicadas ao sistema operacional, e seriam completas se não fosse o surgimento de um novo problema, onde múltiplos usuários e múltiplas aplicações necessitam utilizar diferentes sistemas operacionais. Como solução pode ser feito o uso de diferentes máquinas físicas para que estes sistemas operacionais fossem utilizados ao mesmo tempo

[CROSBY, 2006]. Entretanto, existe a necessidade do uso eficiente dos recursos de *hardware*. Apesar da utilização de múltiplas e avançadas tecnologias de *hardware* prover um grande nível de desempenho para os aplicativos, o custo do *hardware* está cada vez mais alto e alguns dos recursos necessários para seu uso estão se extinguindo, como é o caso da energia elétrica [CROSBY, 2006].

Além do custo proveniente do uso do *hardware*, frequentemente os seus recursos são mal ou pouco utilizados, tornando o sistema ocioso. Em algumas situações o uso dos recursos do *hardware* por determinado aplicativo, chega a apenas 10% do total, mantendo assim 90% dos recursos ociosos. O uso de um ambiente de virtualização pode distribuir de forma eficiente os recursos de *hardware*, entre diferentes máquinas virtuais que executam os diferentes aplicativos que antes eram utilizados através de múltiplas máquinas físicas [VOGELS, 2008].

Além do mau uso dos recursos de *hardware*, existem outros problemas que rondam a informatização atual. A incompatibilidade entre as diferentes plataformas é um deles: a maioria dos aplicativos desenvolvidos para uma plataforma específica como *MS – Windows*, não pode ser executada em outras plataformas, como *Linux*. Como solução para esta incompatibilidade pode-se inserir na arquitetura do sistema uma camada de *software* denominada virtualização. A inserção desta camada permite o funcionamento de interfaces distintas no mesmo *hardware*.



Figura 2: Arquitetura Padrão.

A figura 2 demonstra a arquitetura padrão de um sistema computacional. Basicamente, em um sistema com arquitetura padrão tem-se um *hardware* onde

um sistema operacional executa sobre ele, e diversas aplicações são executadas no sistema operacional, sendo que este é responsável por enviar instruções diretas ao *hardware* para o funcionamento de si mesmo e das aplicações.

2.2 Conceituação

Para que o funcionamento das diversas interfaces ou sistemas operacionais seja efetivado, o ambiente de virtualização cria outras interfaces de acordo com os recursos exigidos pelos sistemas virtuais que executarão sobre ela, estas interfaces criadas pelo ambiente de virtualização são denominadas de máquinas virtuais. Um exemplo de virtualização e uso de máquina virtual pode ser encontrado na figura 3.

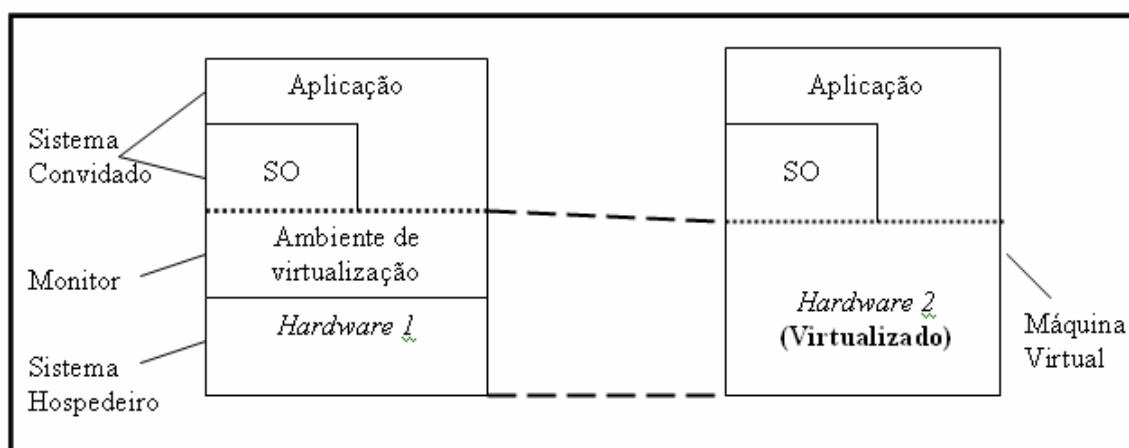


Figura 3: Exemplo de Virtualização e Máquina Virtual [Adaptada de SMITH, 2005].

Neste exemplo de virtualização a camada de *software* é inserida entre o *hardware1* (Sistema Hospedeiro) e o sistema operacional. O ambiente de virtualização cria uma máquina virtual que na figura é representada pelo *hardware2*, sendo que a máquina virtual referente ao *hardware2* é equivalente ao conjunto *hardware1*+Ambiente de Virtualização, ou seja, a máquina virtual apresentada neste exemplo é formada através da união entre ambiente de virtualização e *hardware1*. Esta abstração possibilita o uso de um sistema

operacional virtual (convidado). Outro conceito forte que deve ser empregado é o de que múltiplos sistemas convidados podem ser executados sobre um único ambiente de virtualização (Monitor) [SMITH, 2005].

A definição formal de virtualização foi enunciada por Gerald Popek e Robert Goldberg em 1974:

“Uma máquina virtual é vista como uma duplicata eficiente de uma máquina real. Essa abstração é construída por um monitor de máquina virtual” [POPEK, 1974].

Percebe-se que o conceito empregado por Popek e Goldberg ainda é válido para as arquiteturas e sistemas atuais. Para se elaborar uma virtualização algumas propriedades são definidas, dentre estas se encontra a equivalência, o controle de recursos, a eficiência, o isolamento e a recursividade [MAZIERO, 2008]:

- **Equivalência:** Um ambiente de virtualização deve permitir a execução de um sistema virtual de mesma capacidade e recursos contidos em um sistema físico real.
- **Controle de recursos:** O ambiente de virtualização é responsável pelo controle de acesso a recursos. Nenhum recurso do sistema real pode ser manipulado pela máquina virtual sem que se tenha o controle e o acesso definido pelo ambiente de virtualização.
- **Eficiência:** Depois de definidas as condições de acesso, a maioria das instruções passadas ao processador real devem ser executadas diretamente, sem que se haja a verificação pelo ambiente de virtualização. Esta propriedade é definida para que não ocorra uma excessiva perda de desempenho.
- **Isolamento:** As aplicações existentes em uma máquina virtual não podem interagir diretamente com as aplicações em uma máquina virtual distinta. Estas interações devem ocorrer somente nas situações equivalentes a máquinas físicas, como, por exemplo, por meio de interfaces de rede.
- **Recursividade:** Alguns sistemas virtualizados permitem a inserção de vários níveis de máquinas virtuais, ou seja, uma máquina virtual pode ser executada dentro de outra máquina virtual.

2.3 Camadas de Virtualização

A inserção do *software* de virtualização, ou ambiente de virtualização, pode ser feita nas três camadas de uma arquitetura padrão (figura 2), sendo que cada tipo de inserção constitui um novo conceito de virtualização. Os três conceitos são divididos em três camadas: virtualização de aplicação, virtualização de sistema operacional, e virtualização de *Hardware*.

2.3.1 Camada de Virtualização para Aplicação

Como o nome já diz, esta camada é responsável por gerar sistemas virtuais que têm como objetivo suportar o uso de um processo ou aplicação específica. O emprego do *software* de virtualização executa sobre o sistema operacional hospedeiro, e através deste as instruções são passadas ao *hardware*. A figura 4 demonstra a posição onde é instalado o ambiente de virtualização.

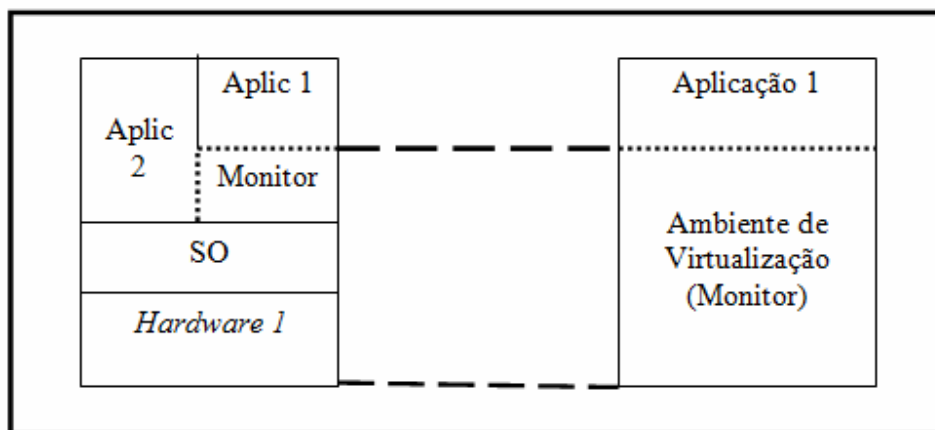


Figura 4: Virtualização na camada de aplicação.

Na figura 4 tem-se um exemplo de uso da camada de virtualização por aplicação, onde um *hardware1* tem sobre si um sistema operacional (SO) e sobre ele são executadas duas aplicações. A aplicação 1 é executada através de uma

máquina virtual na camada de aplicação, enquanto a aplicação 2 executa diretamente sobre o sistema operacional. Neste exemplo pode se entender que a máquina virtual é gerada através da união entre monitor de virtualização, sistema operacional e *hardware*¹. Um exemplo de uso deste sistema seria o uso da máquina virtual *Java*.

Na grande maioria dos monitores utilizados na virtualização na camada de aplicação, é permitido que as aplicações convidadas façam interações diretas com as demais aplicações do sistema, e muitas vezes também é permitido o acesso ao sistema de arquivos e outros recursos do sistema. Estas características infringem o conceito de isolamento, mas possibilitam que o usuário reconheça a aplicação convidada como uma aplicação normal [SMITH, 2005].

O uso da abstração de máquina virtual na camada de aplicação também pode ser visto, de certa forma, em sistemas operacionais conhecidos como multi-tarefas, são aqueles SO's onde é permitida a execução de múltiplos processos. Para esta múltipla execução o SO disponibiliza um processador virtual, uma memória virtual e os recursos físicos para cada processo. [TANENBAUM, 2006]

2.3.2 Camada de Sistema Operacional

O uso de virtualização na camada de sistema operacional funciona de maneira similar à virtualização na camada de aplicação, uma vez que são criadas máquinas virtuais que se utilizam do mesmo *hardware* do SO hospedeiro e o monitor de virtualização é inserido sobre o SO subjacente.

A grande diferença encontrada entre as duas classificações está relacionada à quantidade de máquinas virtuais suportadas: enquanto na aplicação só é possível gerar uma única máquina virtual sobre o monitor de virtualização, na inserção na camada de SO é possível executar múltiplas máquinas virtuais sobre o mesmo monitor de virtualização, sendo que estas são independentes e isoladas umas das outras. A figura 5 apresenta um exemplo desta abstração.

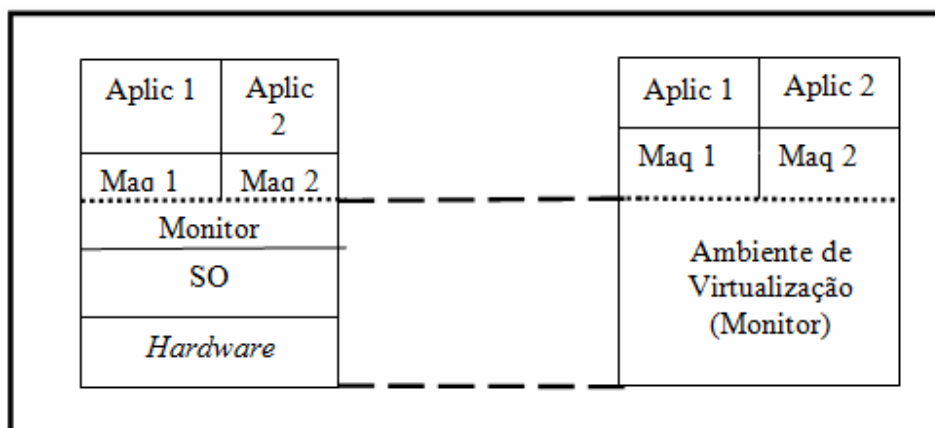


Figura 5: Virtualização na camada de SO.

A figura 5 é um exemplo de virtualização na camada de SO, onde duas aplicações são executadas sobre duas máquinas virtuais distintas, sendo que estas máquinas estão situadas sobre um mesmo ambiente de virtualização. Neste exemplo cada máquina virtual é equivalente à união entre o monitor de virtualização, o sistema operacional, o *hardware*, além do acréscimo das características das máquinas virtuais individuais, sendo que as aplicações 1 e 2 são executadas sobre suas respectivas máquinas virtuais.

O ambiente de virtualização desenvolvido para esta camada é responsável por compartilhar os recursos contidos no sistema operacional subjacente. Esta abstração pode ser analisada como um conceito de réplica, o principal objetivo deste conceito é que cada máquina virtual possa ter a percepção de que está executando com acesso completo sobre um SO isolado e independente. Outro aspecto importante está relacionado ao sistema operacional convidado, que deve ser igual ao SO hospedeiro [NANDA, 2005].

2.3.3 Camada de Hardware

Os ambientes de virtualização desenvolvidos para a camada de *hardware* têm como objetivo a execução de sistemas operacionais virtuais completos, sobre os quais são executadas diversas aplicações [MAZIERO, 2008]. Um exemplo básico da inserção desta camada de virtualização pode ser visualizado na figura 6.

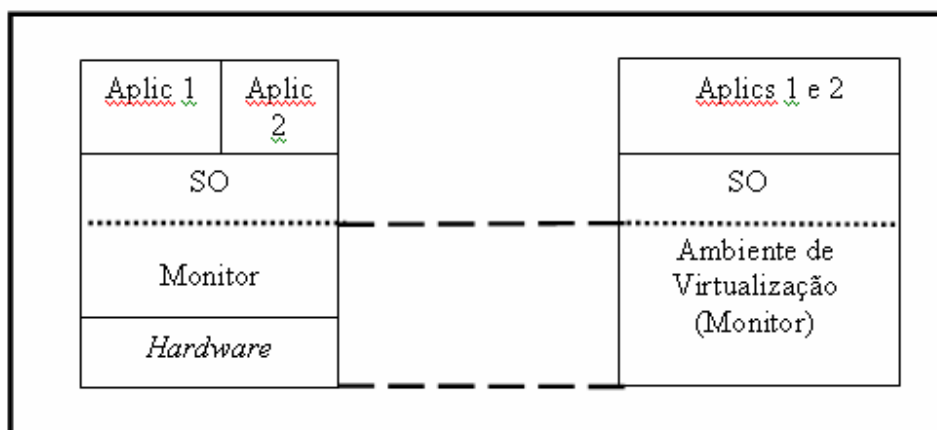


Figura 6: Virtualização Completa.

A figura 6 consiste em um exemplo de virtualização completa, onde o monitor de virtualização é inserido entre a camada de *hardware* e a camada de SO. Neste exemplo, as aplicações necessitam enviar suas instruções para o sistema operacional virtual, e este através do monitor de virtualização envia as instruções ao *hardware*, sendo que a máquina virtual gerada é uma equivalência da união entre recursos do monitor de virtualização e *hardware*.

A virtualização na camada de *hardware* é a mesma abordagem utilizada por Popek e Goldberg na década de 1970, claro que com as devidas evoluções tecnológicas, mas o conceito básico utilizado é o mesmo conceito apresentado na seção 2.2. Este tipo de virtualização permite que os sistemas utilizem uma interface virtual de comunicação idêntica ou distinta da interface real de comunicação com o *hardware*. Com isto cada máquina virtual criada possui o seu próprio conjunto de recursos virtuais, como por exemplo, disco rígido, interface de rede e memória. O uso deste modo de virtualização também permite a execução de múltiplos sistemas operacionais.

A virtualização na camada de *hardware* pode ser classificada em vários tipos de acordo com as características arquiteturais e o nível de virtualização do *hardware*. Quando se fala em arquitetura são encontrados dois tipos básicos de monitores virtuais, o monitor nativo e o monitor convidado. E ao se falar de abrangência ou profundidade de virtualização de *hardware*, também são definidos

dois tipos básicos, a virtualização completa e a virtualização parcial [SMITH, 2005].

- Monitor nativo: Para que um monitor de virtualização seja considerado um monitor nativo, a sua execução deve ser direta sobre o *hardware*, sem que haja interação com um sistema operacional subjacente. Sua função é replicar todos os recursos de *hardware*, de forma que cada máquina virtual tenha a imagem de estar executando independentemente.
- Monitor convidado: Para que um monitor de virtualização seja considerado um monitor convidado, a sua execução deve ser realizada sobre um sistema operacional subjacente, ou seja, a sua execução deve ser como um processo deste SO. Neste tipo de inserção só é permitida a execução de uma máquina virtual sobre cada monitor; para que haja a execução de múltiplas máquinas virtuais, devem ser executados outros processos de monitores.
- Virtualização completa: Quando um ambiente de virtualização se encaixa nesta categoria, significa que todos os recursos de *hardware* são virtualizados, incluindo-se aqui as instruções ao processador e dispositivos. Com isso, é possível executar SO's convidados escritos para plataformas de *hardware* distintas do *hardware* real subjacente.
- Virtualização parcial: Nesta categoria a interface real de comunicação com o *hardware* é mantida, apenas as instruções privilegiadas e os recursos são virtualizados, sendo que o SO convidado e as aplicações convidadas têm acesso direto ao *hardware* real. Esta abstração permite um desempenho próximo a uma execução sobre uma plataforma real.

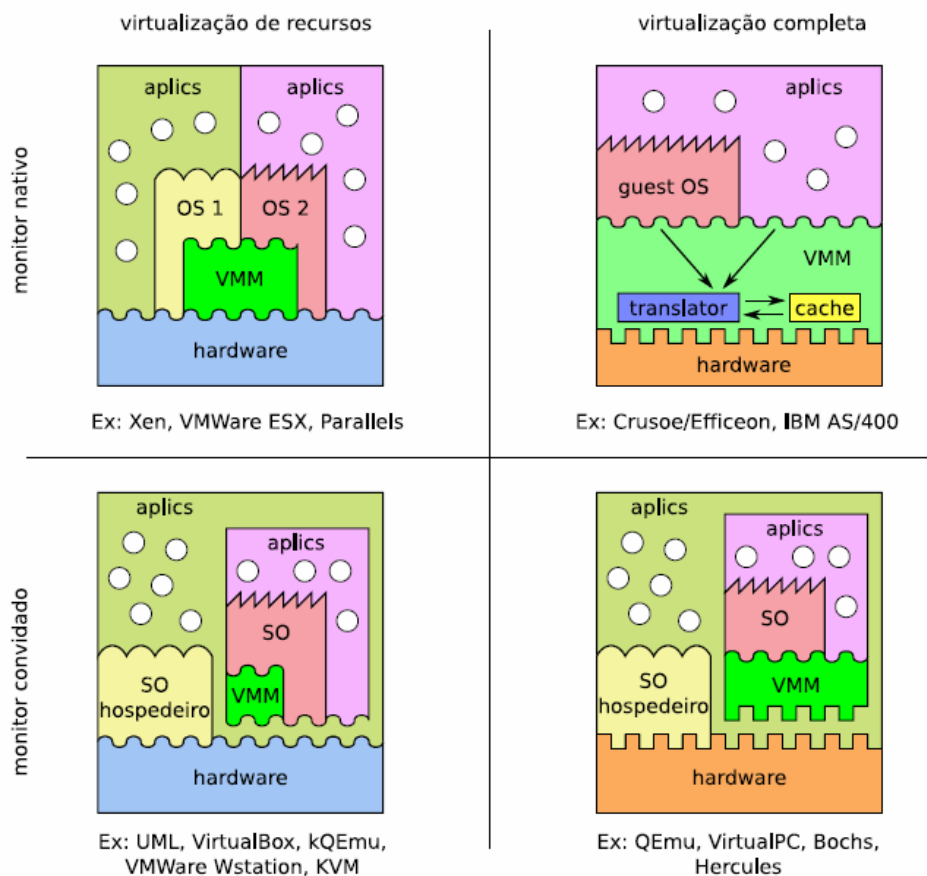


Figura 7: Exemplos de virtualização [MAZIERO, 2008].

Ao analisar a figura 7 pode se perceber que alguns conceitos são complementares. A inserção de um monitor nativo pode conter características que classifiquem essa virtualização como completa, para melhor entendimento deve se dividir esta figura em quatro quadrantes, no primeiro quadrante é apresentado um exemplo de virtualização de recursos (parcial) junto ao conceito de monitor nativo, neste quadrante a inserção da máquina virtual é feita diretamente sobre o *hardware* e os acessos das aplicações aos recursos podem ser feitos de forma direta ou indireta. No segundo quadrante, também tem se um exemplo de monitor nativo onde a inserção da máquina virtual é feita diretamente sobre o *hardware*, porém esta virtualização utilizada é considerada uma virtualização completa, pois todos os recursos são virtualizados. No terceiro quadrante tem-se outro exemplo de virtualização parcial onde o monitor de virtualização é caracterizado como convidado, pois ele é considerado como uma aplicação do SO. Já no quarto e

último quadrante tem-se um exemplo de virtualização completa onde o monitor de virtualização é considerado convidado, pois todos os recursos de *hardware* são virtualizados, e a máquina virtual é considerada como uma aplicação sobre o SO.

2.3.4 Paravirtualização

O conceito de paravirtualização pode ser visto como um tipo de virtualização em nível de *hardware*, porém neste conceito algumas características não têm intersecção com as características deste nível. Contudo, a paravirtualização será denotada como um conceito diferenciado de virtualização.

Durante o processo de desenvolvimento de tecnologias de virtualização algumas dificuldades foram encontradas com relação a algumas arquiteturas de *hardware*. No caso dos processadores com arquitetura x86 existe uma restrição quanto às instruções privilegiadas do processador, algumas das instruções são deste tipo, e por definição um programa usuário (ambiente de virtualização por emulação) não pode enviar tais instruções. Outra dificuldade observada está relacionada à gerência de memória, que é feita diretamente pelo *hardware* e não permite a intervenção direta do monitor de virtualização.

Outro empecilho envolvendo a completa virtualização de um *hardware* é que em alguns casos esta abstração pode causar alguns problemas chamados de “Buracos de Virtualização”. Algumas instruções podem ser enviadas tanto pelo modo usuário quanto pelo modo supervisor, podendo produzir diferentes resultados dependendo do modo em que estão sendo executadas [CROSBY, 2006].

Diante destas dificuldades, algumas possíveis soluções foram levantadas. Uma delas é modificar a *interface* entre o monitor e o sistema operacional convidado, de tal forma que apresente ao sistema convidado um *hardware* virtual semelhante ao *hardware* real. Este novo conceito empregado é denominado de paravirtualização. A paravirtualização aproxima o monitor do sistema virtual convidado, evitando a incompatibilidade entre as instruções e aumentando assim o desempenho entre as máquinas virtuais.

O uso de paravirtualização aproxima-se muito da virtualização na camada de *hardware*, a diferença é que na paravirtualização um sistema operacional convidado deve sofrer modificações impostas pelo monitor de virtualização, modificações estas voltadas aos seguintes recursos:

- O mecanismo de entrega de interrupções passa a usar um serviço de eventos oferecido pelo monitor.
- As operações de entrada e saída são realizadas através de uma *interface* simplificada, a qual independe de dispositivos [MAZIERO, 2008].
- Durante as consultas do núcleo convidado, são permitidos os acessos às tabelas de páginas de memória, porém as modificações devem ser solicitadas ao monitor [CROSBY, 2006].
- O nível de privilégio dos núcleos convidados deve ser inferior ao nível do monitor. [MAZIERO, 2008].
- O tratamento das chamadas de sistemas das aplicações deve ser implementado pelo núcleo do sistema convidado, evitando que estas chamadas passem pelo monitor antes de chegar ao núcleo convidado em questão.

Ao utilizar o conceito de virtualização na camada de *hardware* o monitor de máquina virtual deve buscar os *drivers* adequados para o funcionamento dos núcleos convidados, que realizam acessos nos dispositivos virtuais, portanto eles não necessitam de *drivers* específicos. Para evitar que um conjunto de múltiplos *drivers* específicos sejam desenvolvidos para o monitor de máquina virtual, alguns *softwares* de virtualização utilizam o conceito de réplica de *drivers*. A primeira máquina virtual criada é responsável por dispor os *drivers* necessários ao monitor, e as máquinas virtuais subseqüentes devem utilizar os mesmos *drivers* disponibilizados pela primeira máquina [SMITH, 2005]. A exemplificação deste conceito pode ser visto na figura 8.

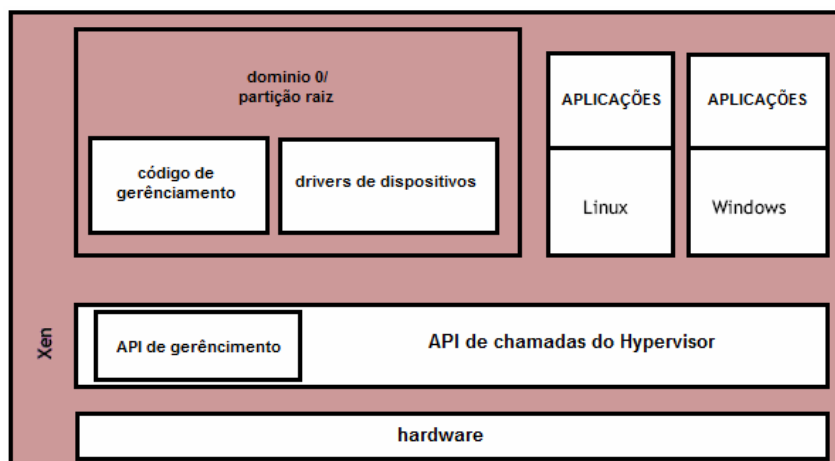


Figura 8: Exemplo de paravirtualização utilizando Xen [Traduzido de CROSBY, 2006].

Na figura 8 pode se perceber que o ambiente de virtualização (Xen) executa diretamente sobre o *hardware*, os núcleos convidados executam sobre o ambiente de virtualização acessando os devidos recursos, sendo que também é disponibilizado um setor específico para os *drivers* (*device drivers*) a serem utilizados pelos núcleos convidados.

2.4 Camadas em Função da Consolidação

Quando se trata de consolidação de servidores é necessário definir quais são as camadas interessantes para se realizar tal objetivo. Neste capítulo foi visto que a camada de virtualização para aplicação não tem seu uso destinado ao fim de consolidação, pois não é possível através dela se virtualizar um sistema operacional completo e também através desta forma não é possível realizar a virtualização de recursos físicos, como por exemplo, a placa de rede. Portanto fica descartado o uso deste tipo de inserção para a consolidação.

Para definir qual das duas formas de inserção remanescentes será utilizada, faz-se necessário uma avaliação das ferramentas existentes para cada

um destes dois tipos de inserção de virtualização, pois, percebe-se que através de ambos os tipos de inserção é possível se realizar a consolidação de servidores. A avaliação das características presentes nas ferramentas bem como a definição de quais delas tem importância relevante para a definição de qual ferramenta será utilizada posteriormente é realizada no capítulo 3, que tem por foco a pesquisa relacionada às ferramentas de virtualização.

3. FERRAMENTAS

Depois de definidos os conceitos e as classificações de virtualização é importante realizar um levantamento sobre as ferramentas existentes para cada camada citada no capítulo 2. Este levantamento sobre as características de cada ferramenta será importante para definição do ambiente de virtualização a ser utilizada na análise comparativa proposta para este trabalho.

É válido frisar que o foco da análise comparativa deste trabalho é o uso de ferramentas na camada de Sistema Operacional e *Hardware*, pois somente através destas se pode efetuar a criação de máquinas virtuais com características de servidor, ou seja, máquinas virtuais com características de máquinas físicas provedoras de serviços.

Para se definir qual tipo de ferramenta será utilizado para os testes, deve-se verificar alguns itens importantes como: a capacidade de que a ferramenta de virtualização tenha para virtualizar sistemas operacionais distintos, se a ferramenta possui a capacidade de virtualizar plataformas distintas e também é importante verificar os níveis de desempenho de cada ferramenta.

3.1 Ferramentas para a camada de sistema operacional

Entre as ferramentas com características de virtualização referentes as da camada de sistema operacional, pode-se identificar a ferramenta *VMWare Workstation* e a ferramenta *VirtualBox*. As características relevantes sobre cada um destes ambientes de virtualização serão apresentadas nas seções 3.1.1 e 3.1.2.

Normalmente, as ferramentas existentes para a virtualização inserida na camada de sistema operacional possuem a característica que lhes permite virtualizar diferentes sistemas operacionais, porém não possuem a característica

de virtualizar plataformas diferentes da qual está inserida. Outra desvantagem encontrada neste tipo de ferramenta é a questão de desempenho, pois ela deve executar como uma aplicação dentro do sistema operacional, não tendo privilégios de administrar muitos recursos [NANDA, 2005].

3.1.1 VMWare Workstation

A ferramenta de virtualização *VMWare* possui tanto características para a camada de *hardware* quanto para a camada de sistema operacional, estas características dependem da versão desenvolvida. Sendo que a última versão lançada, o *VMWare ESX Server*, possui características de camada de *hardware*, e a versão *VMWare Workstation* possui características de camada de SO [VMWARE, 2007].

O conjunto de ferramentas *VMWare* é um *Software* proprietário e foi desenvolvido para as arquiteturas de *hardware* x86 e x86-64, sendo compatível com os sistemas operacionais *MS-Windows*, *Linux* e *Mac OS X*. O *VMWare Workstation* foi desenvolvido em 1999. Com relação à classificação da ferramenta, ela possui as características de virtualização completa e virtualização parcial. No entanto, a ferramenta dispõe de versões gratuitas, porém estas não são versões que dispõe da totalidade dos recursos.

Basicamente, seu funcionamento ocorre da seguinte forma: Em um sistema operacional específico é executado um novo processo, um ambiente de virtualização *VMWare*. Através deste *software* é criada uma máquina virtual executando um outro sistema operacional virtualizado, no qual são geradas as diversas aplicações.

A utilização do *VMWare Workstation* também provê a propriedade da portabilidade, uma vez que através desta ferramenta é possível executar aplicações diferentes daquelas que o sistema operacional real permitiria, contudo é importante lembrar que esta ferramenta é desenvolvida para a plataforma x86 e não pode executar aplicações ou SO's de *hardwares* diferentes.

3.1.2 VirtualBox

O *VirtualBox* possui características semelhantes ao *VMWare Workstation*, ambos são sistemas proprietários, assim como no *VMWare*, no *VirtualBox* existem licenças gratuitas para usuários privados (não comerciais). Além das características voltadas à propriedade do *software*, outra semelhança é a classificação do ambiente de virtualização, o *VirtualBox* também possui propriedades que caracterizam a virtualização parcial, bem como, o monitor é identificado como um sistema convidado [VIRTUALBOX, 2008].

Esta ferramenta de virtualização foi desenvolvida por um grupo alemão denominado *innotek*, o qual agora é uma subsidiária da *Sun Microsystems*. Em janeiro de 2007 uma nova versão do *VirtualBox* foi lançada com sistema de código aberto, para tanto algumas restrições foram impostas: foi retirado o protocolo de acesso remoto, o suporte à porta USB e também foi retirado um protocolo que fazia o controle de acesso ao disco rígido denominado iSCSI. Estes recursos continuam presentes nas versões proprietárias.

Da mesma forma que *VMWare Workstation*, o *VirtualBox* também foi desenvolvido para as plataformas x86, sendo que o monitor de virtualização pode ser executado como aplicação dos sistemas operacionais *MS-Windows*, *Linux* e *Mac OS X* e pode gerar máquinas virtuais *FreeBSD*, *Linux*, *Windows*, *Mac OS X* e *Solaris*.

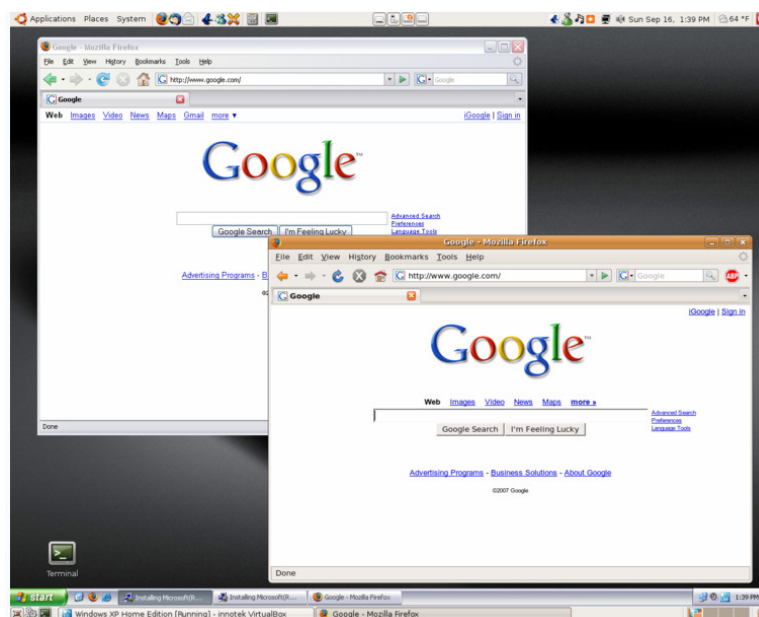


Figura 9: Execução de um SO Windows XP sobre um SO Ubuntu Linux. [VIRTUALBOX, 2008].

A figura 9 mostra um exemplo da utilização do ambiente de virtualização *VirtualBox*, uma máquina virtual é gerada e nesta máquina é executado um sistema operacional *MS – Windows*, sendo que o SO hospedeiro do ambiente de virtualização é o Ubuntu Linux.

A propriedade de portabilidade também é mantida no *VirtualBox*, já que é possível virtualizar um sistema operacional convidado em uma plataforma diferente daquela para qual ele foi projetado. Outra vantagem que pode ser encontrada no *VirtualBox* é a possibilidade de redução de *hardware*, pois existe o atributo de que diversas máquinas virtuais podem ser criadas em um mesmo ambiente de virtualização.

Além dos ganhos referentes à economia e portabilidade, o *VirtualBox* também é de fácil utilização, pois sua instalação é realizada através de interface gráfica e a sua configuração é auto-explicativa.

3.2 Ferramentas para a camada de *hardware*

As ferramentas estudadas nesta seção são aquelas que possuem características inerentes aos ambientes de virtualização para a camada de *hardware*, ou seja, aqueles que atingem as necessidades citadas na seção 2.3.3. Entre as ferramentas de virtualização pode-se encontrar a ferramenta *Xen* e a *Denali*.

Com relação as características destes ambientes de virtualização pode se dizer que eles possuem um desempenho um pouco maior do que aquele inserido sobre o sistema operacional. Este desempenho está relacionado com a técnica que é utilizada para se realizar a virtualização, se a técnica utilizada é a *paravirtualização* o desempenho é bem superior ao das ferramentas inseridas sobre a camada de SO. Além do desempenho é importante frisar que algumas destas ferramentas permitem que sejam virtualizadas plataformas diferentes daquela onde o ambiente é inserido, podendo assim virtualizar, uma gama maior de sistemas operacionais [NANDA, 2005].

3.2.1 *Denali*

O ambiente de virtualização *Denali* foi desenvolvido pela Universidade de Washington, inicialmente seu projeto surgiu com o intuito de suportar múltiplas máquinas virtuais sobre uma única arquitetura x86. Esta ferramenta tem seu uso voltado para a utilização de pequenas máquinas virtuais e não tem suporte aos sistemas operacionais convencionais como *Windows* e *Linux*, sendo que cada máquina virtual executa um serviço de rede [MAZIERO, 2008].

As características relacionadas a esta ferramenta se fazem presente junto ao conceito de *paravirtualização*. Como o *Denali* foi desenvolvido para pequenas aplicações e serviços, o propósito do uso da *paravirtualização* nesta ferramenta é obter uma arquitetura com maior escalabilidade, desempenho e simplicidade. [WHITAKER, 2002]

3.2.2 Xen

Este ambiente de virtualização é uma ferramenta de código aberto, portanto é um *software* gratuito, a sua execução é possível em diversas arquiteturas de *hardware* como a arquitetura IA-32, a x86-64 e PowerPC 970. Por ter uma grande diversidade de arquiteturas suportadas, a ferramenta *Xen* também possui grande variedade de sistemas operacionais suportados tais como *MS-Windows* (este apenas como convidado), *Linux*, *Solaris* e derivações de sistemas operacionais *Unix*. [XEN, 2008]

A ferramenta *Xen* foi desenvolvida na universidade de Cambridge, e seu projeto foi liderado por *Ian Pratt*, mais tarde, em outubro de 2007, o projeto foi adquirido pela empresa *Citrix Systems*.

O monitor de máquina virtual gerado pela ferramenta *Xen* tem como características os conceitos referentes à virtualização na camada de *hardware*. O princípio de virtualização utilizado é a *paravirtualização*. A Figura 8 inserida na seção 2.3.4 apresentou o funcionamento da ferramenta *Xen*. A figura 10 mostra um exemplo da execução da ferramenta *XEN*, demonstrando também a propriedade de portabilidade: em um mesmo *hardware* são executadas três sistemas operacionais *Linux* (janelas marcadas pelos números 1,3 e 4) e um SO *NetBSD* (janela marcada pelo número 2). É válido lembrar que a ferramenta *Xen* também permite o acesso remoto. Outro ponto importante está relacionado à instalação e configuração, que é realizada através de linhas de comando.

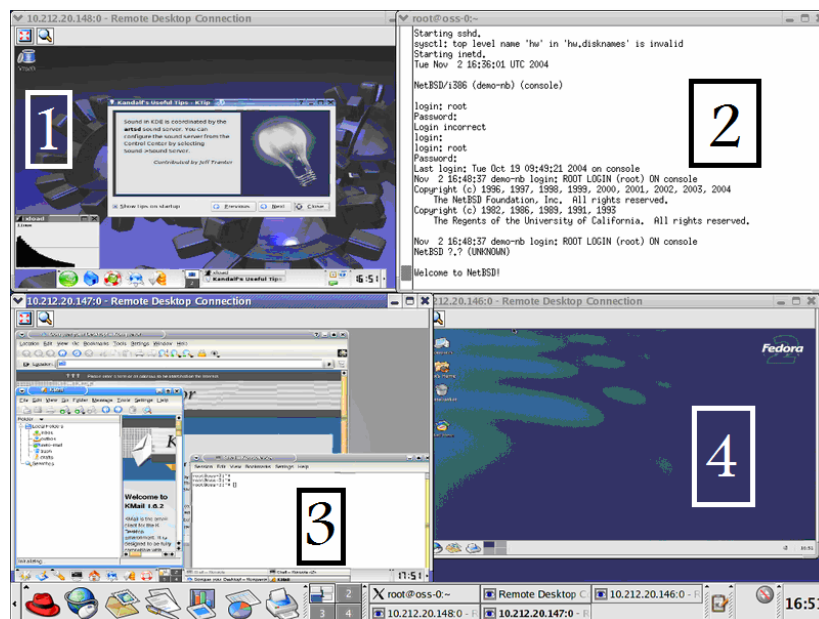


Figura 10: Xen, Múltiplas plataformas Linux e NetBSD [XEN, 2008].

3.3 Comparação entre Ferramentas

Depois de realizados os estudos sobre cada ferramenta individualmente, é necessário realizar uma comparação entre as mesmas, levando principalmente em consideração os aspectos relevantes para a consolidação de servidores, percebe-se que a comparação será realizada com todas as ferramentas pesquisadas: VMWare, VirtualBox, Denali e XEN, porém a ênfase maior é sobre aquelas que possibilitem a virtualização de servidores, propiciando assim a consolidação.

Antes de apresentar quais os aspectos relevantes e qual a decisão final referente a utilização das ferramentas, é apresentado um resumo em forma de tabela sobre as características das ferramentas avaliadas, sendo que este será feito com base nas seguintes características:

- **Camada de inserção.** Este item identifica em qual das camadas de virtualização estudadas que a ferramenta é aplicada, pode ser classificada em *Hardware* ou sistema operacional;

- **SOs Distintos.** Esta característica identifica se através do uso desta ferramenta é possível virtualizar sistemas operacionais distintos daquele para o qual a ferramenta foi criada, por exemplo, ao executar esta ferramenta sobre um ambiente *Windows* seria possível gerar uma máquina virtual de um SO *Linux*. As ferramentas podem ser classificadas em ‘sim’ ou ‘não’;
- **Plataformas distintas.** Através deste item é possível definir se a ferramenta pode virtualizar plataformas de *hardware* distintos, como por exemplo, através da ferramenta inserida sobre uma plataforma x86, se realizar a virtualização de uma plataforma *SPARC*. Também classificadas por ‘sim’ ou ‘não’;
- **Licença.** Se a ferramenta necessita de pagamento de licença para ser utilizada. A indicação deste item no comparativo é feita por ‘Sim’, ‘não’, ou ‘Depende’. Nas ferramentas classificadas com ‘depende’, a licença varia conforme a versão das mesmas;
- **Consolidação.** Aspecto que indica se com o uso da ferramenta é possível realizar a consolidação de servidores, ou seja, se existe a possibilidade de virtualizar múltiplas máquinas físicas através de máquinas virtuais. Classificam-se as ferramentas em ‘Sim’ ou ‘Não’;
- **Documentação.** Aqui é avaliada a quantidade de documentos disponibilizados pelas ferramentas. Considera-se ‘bom’, ‘ruim’ ou ‘médio’.

A tabela 1 traz um resumo da comparação entre as ferramentas existentes, onde as mesmas são classificadas de acordo com os aspectos anteriormente definidos:

Tabela 1: Comparativo entre ferramentas de virtualização.

Aspectos	Ferramentas			
	VMWare	VirtualBox	Denali	Xen
Camada de Inserção	Hardware ou SO	SO	Hardware	Hardware
SOs Distintos	Sim	Sim	Não	Sim
Plataformas Distintas	Não	Não	Não	Sim
Licença	Depende	Depende	Não	Não
Consolidação	Sim	Sim	Não	Sim
Documentação	Bom	Médio	Ruim	Bom

Ao realizar uma análise sobre a tabela 1 é possível definir que:

- O *VMWare* é uma ferramenta que possui características tanto de virtualização de *hardware* como de sistema operacional e esta classificação depende da versão utilizada. Percebe-se também que com o uso desta ferramenta é possível realizar a virtualização de sistemas operacionais distintos, mas plataformas distintas não podem ser virtualizadas. A classificação quanto a licença é de que a ferramenta necessita de pagamento de licença para algumas versões da mesma. No entanto esta é uma boa ferramenta para se realizar a consolidação de servidores, uma vez que permite a virtualização de múltiplos SOs, considera-se também que esta é uma ferramenta bem documentada;
- O *VirtualBox* é uma ferramenta que é inserida sobre a camada de sistema operacional e também permite a virtualização de sistemas operacionais distintos daquele sobre o qual a ferramenta é executada, percebe-se que como no *VMWare* ela não pode virtualizar plataformas distintas. Quando se fala em licença, o uso desta ferramenta é permitido para ambientes de pesquisa, ou seja, não é permitido em ambientes comerciais. Observa-se também que ao utilizar esta ferramenta é possível realizar a consolidação de

servidores, uma vez que também existe aqui a possibilidade de se criar máquinas distintas. Ao se falar em documentação, esta ferramenta pode ser considerada medianamente documentada, uma vez que tem uma boa explicação de como se instalar a ferramenta, porém o seu uso não é muito documentado;

- A *Denali* é uma ferramenta inserida sobre a camada de *Hardware* que não permite a virtualização de sistemas operacionais distintos e nem de plataformas distintas, porém ela não necessita de pagamento de licença para ser utilizada. Observa-se que a consolidação não é uma característica presente nesta ferramenta, pois a maioria dos sistemas operacionais utilizados em servidores não podem ser virtualizados. Outro item importante é relacionado a documentação, que é classificada como ruim, por não detalhar bem a instalação e uso da mesma;
- A ferramenta *Xen* tem características de virtualização para a camada de *hardware* e possui o atributo de virtualizar múltiplos SOs e plataformas. Percebe-se que para se utilizar esta ferramenta não é necessário adquirir licença e ela também pode ter seu uso voltado para a consolidação de servidores, pois múltiplas máquinas virtuais podem ser criadas a fim de virtualizar máquinas físicas. Outro item importante percebido é que a documentação da ferramenta é boa, bem detalhada tanto para instalação quanto para o uso.

Com os dados obtidos é possível definir que a ferramenta a ser utilizada durante os testes será a ferramenta *Xen*, pois ela compõe de forma completa todos os atributos relevantes para se realizar testes referentes ao uso de virtualização para consolidação de servidores, e ao se comparar com a ferramenta *VMWare*, ela (*Xen*) não traz restrições quanto ao uso de licenças de *software*.

4. MÉTRICAS

Durante este processo de definição de métricas e testes uma ordem cronológica deve ser seguida, pois antes de se atingir um objetivo é necessário que um anterior ao mesmo seja atingido. Primeiramente é necessário decidir qual o serviço que será virtualizado, após devem ser listados os requisitos presentes na avaliação do mesmo, quando o levantamento de requisitos é concluído é necessário selecionar ou associar dentre estes requisitos quais são as métricas relevantes para os mesmos. Depois de definidos estes passos iniciais deve-se propor os testes e os ambientes nos quais os mesmos serão realizados.

Existem muitos tipos de serviços que podem ser implementados a fim de se fazer uma análise referente a virtualização, porém como o foco deste trabalho é a consolidação de servidores, a escolha é direcionada para um serviço que tenha o seu uso de forma freqüente. Percebe-se que, dos vários serviços existentes que podem se beneficiar da consolidação, servidores *Web* e de banco de dados estão entre os mais populares. Dentre eles, apenas um será utilizado, pois o foco do trabalho é a definição do custo/benefício da virtualização em um ambiente específico, portanto não se faz necessário aqui a virtualização e execução de testes sobre diferentes serviços.

Dentre os serviços citados, faz-se um direcionamento maior para o uso do serviço de servidor *web*, pois este serviço é um dos mais utilizados entre as diferentes instituições informatizadas. E, além de ser um serviço popular, o servidor *web* é de fácil implementação quando comparado ao serviço de banco de dados. Outra vantagem encontrada é que ele não necessita de um ambiente computacional muito complexo, adequando-se assim ao ambiente computacional do grupo de pesquisa GRADIS.

É válido frisar que o serviço aqui definido possui uma gama de diferentes implementações, ou seja, ele pode ser implementado de várias formas e pode conter diferentes recursos, como por exemplo, o controle de acessos, porém o

serviço a ser implementado será feito de forma básica e simples, apenas serão implementados os recursos necessários para um funcionamento básico do mesmo.

4.1 Funcionamento de um Servidor *Web*

Um servidor *web* ou HTTP é uma máquina que tem por objetivo armazenar páginas, e disponibilizá-las a diversos usuários em uma rede interna ou externa. Nesta seção serão descritas as características de um servidor *web*, indicando quais são os processos básicos relacionados ao seu funcionamento. Basicamente, um cliente (geralmente um navegador *web*) utiliza o protocolo HTTP sobre um canal TCP para se comunicar com o servidor e realizar as suas requisições. Para efetuar as requisições o cliente envia ao servidor um cabeçalho HTTP, sendo que neste cabeçalho estão inseridas as informações que o cliente necessita.

Em um servidor *web* pode-se encontrar dois tipos de dados, aqueles de conteúdo estático e aqueles de conteúdo dinâmico. Os conteúdos estáticos são armazenados como arquivos de disco, já os dados com conteúdo dinâmico são gerados com apoio de aplicações auxiliares que são executadas no servidor [PAI, 1999].

Depois que o servidor recebe a requisição do cliente, ele envia um cabeçalho HTTP de resposta, que é seguido pelos dados solicitados pelo cliente. O processo básico de funcionamento de um servidor HTTP (*web*) com conteúdos estáticos pode ser visualizado na figura 11.

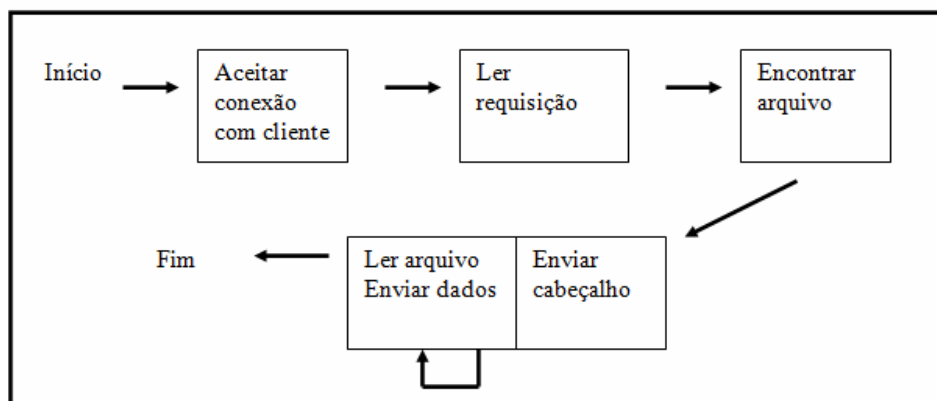


Figura 11: Funcionamento básico de um servidor WEB

Na figura 11 pode se ter uma boa abstração sobre como é o funcionamento básico de um servidor *web*, e a explicação de cada um dos passos é realizada através dos seguintes itens:

Aceitar conexão com cliente: Neste passo o servidor deve aceitar a conexão solicitada pelo cliente, esta solicitação é realizada em um *socket* de escuta que foi criado pelo servidor, depois de aceito, é criado um novo *socket* para comunicação com o cliente.

Ler requisição: O servidor lê o cabeçalho de solicitação emitido pelo cliente, e analisa qual a pagina solicitada e os atributos da solicitação.

Encontrar arquivo: Procura no sistema de arquivos para encontrar o arquivo solicitado pelo cliente e verifica se o mesmo existe e se o cliente possui as devidas permissões. O tamanho do arquivo e data da última modificação são incluídos no cabeçalho HTTP de resposta.

Enviar cabeçalho: Transmite o cabeçalho de resposta HTTP para o cliente.

Ler arquivo: Carrega todo o arquivo ou parte dele para posteriormente enviá-lo.

Enviar dados: Transmite o conteúdo requerido pelo usuário através da rede, para grandes arquivos este procedimento de envio e o procedimento de leitura do arquivo são repetidos até que todo o arquivo seja enviado.

Percebe-se também que uma abstração do funcionamento de um servidor *web* pode ser feita de forma mais complexa, pois outros itens devem ser levados em consideração, como por exemplo, a arquitetura do processador contido na máquina servidora. Em máquinas multiprocessoras pode-se utilizar a arquitetura apresentada na figura 11, apenas é necessário destacar que é possível executar múltiplos processos. Já em arquiteturas de um único processo é necessário fazer uma abstração diferenciada. Como o foco deste trabalho é realizar a análise da virtualização, não serão caracterizadas aqui as diferentes abstrações existentes. Portanto se tem por definido que o modelo a ser utilizado para implementação de servidor *web* é aquele apresentado na figura 11.

4.2 Requisitos

Depois de concluído o estudo sobre o serviço a ser implementado, é importante definir quais são os aspectos ou requisitos computacionais que são necessários ou que são essenciais para o funcionamento de um servidor *web*.

Como foi dito na seção 4.1, durante o processo completo de funcionamento um servidor *web* básico realiza diferentes tipos de acessos a recursos de *hardware*, e algumas funcionalidades são executadas, como a leitura e gravação de arquivos, a resposta enviada ao usuário, o uso da *interface* de rede e a busca de arquivos. Com base nestas funcionalidades, a análise de alguns requisitos se faz necessário, para que posteriormente, sejam definidas as métricas:

- Uma das métricas a ser utilizada deve avaliar o desempenho do disco rígido durante os testes, este requisito está relacionado com a funcionalidade de leitura e gravação.

- Faz-se necessário também definir uma métrica para avaliação dos recursos de rede, ou da *interface* de rede, uma vez que este requisito está relacionado com múltiplos acessos dos clientes.
- Outro item que é importante para a definição de métrica, é referente a capacidade de processamento do servidor, uma vez que o tempo relacionado ao processo completo desde o início até o fim do mesmo também deve ser analisado durante os testes, entende-se aqui início como o momento em que o servidor recebe a requisição do cliente e fim o momento em que o servidor entrega a ele a resposta.

Além dos requisitos relacionados com as métricas também é importante definir requisitos físicos importantes para o projeto dos testes:

- É necessária a implantação de uma máquina que possua um *software* responsável por prover o serviço *web*.
- Um ou mais clientes realizando acessos sobre o servidor.
- E uma rede que conecta os clientes ao servidor, rede esta que pode ser local.

Com base nestes requisitos será necessário especificar quais as métricas que estão relacionadas com os mesmos, para então definir quais as ferramentas que irão auxiliar na captura dos dados que estejam em conformidade com tais métricas.

4.3 Definindo métricas

Definidos os requisitos que necessitam ser avaliados para que se tenha uma análise concreta de um servidor *web*, é necessário definir quais as métricas que estão relacionadas com estes requisitos, definir de forma específica quais variáveis serão relevantes para a pesquisa. Percebe-se que as métricas definidas são classificadas em dois tipos: as internas e as externas.

4.3.1 Métricas de avaliação interna.

As métricas de avaliação interna são aquelas responsáveis por avaliar alguns aspectos relativos ao servidor que disponibiliza o serviço. É importante avaliar estas métricas para que se tenha uma abordagem sobre os aspectos interessantes para o administrador de informática de um meio corporativo, como por exemplo, uso de processador, uso de memória, uso de disco e de interface de rede.

- **Processador:** É necessário avaliar, nos diferentes testes efetuados, qual o uso do processador da máquina sobre a qual estão sendo executados os serviços, tanto virtuais como reais, deve ser analisada pelos tempos gastos para realizar operações específicas, como execução de processos, criação de processos e comunicação entre processos.
- **Memória:** Nesta métrica é avaliada qual a porcentagem dos recursos da memória que estão sendo utilizados pelos serviços executados sobre a máquina física.
- **Disco:** Esta métrica também é conhecida como vazão ou *throughput* de arquivos, e define a quantidade de *bytes* que podem ser transferidas no disco por unidade de tempo.
- **Rede:** Outro aspecto importante é definir qual o *throughput* da rede, ou seja, a quantidade de *bytes* que a interface de rede pode transmitir em um determinado período de tempo.

4.3.2 Métricas de avaliação externa.

As métricas externas são aquelas que têm relação com a percepção do usuário sobre o serviço, são avaliados por ela dados como o tempo de resposta, o número de conexões simultâneas suportadas e a disponibilidade real do serviço. Para tanto as seguintes métricas foram definidas, a fim de que se tenha uma melhor conformidade com os requisitos apontados, e também para que as

mesmas tenham relevância para a análise final proposta no trabalho [MIDGLAY, 2001]:

- **Tempo de resposta:** Esta métrica deve indicar o tempo de resposta para uma determinada requisição do cliente. Pode se considerar esta métrica como uma boa medida de desempenho relacionada com a percepção do usuário.
- **Requisições por segundo:** Esta métrica deve indicar o número de requisições por segundo que o servidor *web* é capaz de sustentar, pode se considerar esta métrica como uma medida de desempenho bruta, estando relacionada com a avaliação total do desempenho do servidor.
- **Número de conexões simultâneas:** Esta métrica avalia o número de conexões simultâneas que o servidor é capaz de suportar, para tanto é necessária a utilização de uma ferramenta de testes que simule múltiplas conexões. Esta métrica atinge a conformidade com o requisito de avaliação de desempenho de recursos de rede.

4.4 Ferramentas

Definidas as métricas, é necessário definir quais as ferramentas que irão auxiliar no processo de captura dos dados compatíveis com as mesmas. O objetivo desta seção é descrever quais as ferramentas que serão utilizadas, bem como denotar as características das mesmas e os princípios de funcionamento das mesmas. Já a explicação e exemplificação deste funcionamento serão realizadas nos capítulos que descrevem a execução dos testes.

Dentre as ferramentas encontradas pode-se citar as seguintes [PAI, 1999]:

Iperf: É uma ferramenta simples que executada sobre um sistema operacional recupera dados de desempenho do *hardware*, mais especificamente sobre a interface de rede. Durante o seu funcionamento uma rede é simulada e

múltiplas tentativas de acesso e troca de arquivos são efetuadas. O resultado é dado como uma espécie de *throughput* máximo da placa, este é representado pelo número de bits por segundo transferidos.

Httpperf: É uma ferramenta simples para cálculo de desempenho de um servidor *web*, onde os clientes realizam uma série de requisições para um mesmo arquivo no servidor e a avaliação da ferramenta é sobre o número de requisições respondidas e o tempo de resposta destas requisições. O uso desta ferramenta está relacionado com a definição dos dados referentes à métrica que avalia o número de requisições por segundo, bem como, a de retorno de arquivos (pode ser chamada de tempo de resposta). Os dados escolhidos, entre os apresentados por esta ferramenta, estão nas seguintes unidades de medida: Requisições por segundo; Respostas por segundo (indica os valor mínimo e o máximo obtido); Tempo de resposta em *milissegundos*.

LMBench: Esta ferramenta realiza uma série de testes sobre o sistema computacional (*Hardware* + SO). Cada função executada representa uma métrica diferente, o uso desta irá apresentar os dados utilizados na comparação entre o uso da virtualização e o não uso da mesma. Dentre os diferentes testes provenientes, pode-se citar como selecionados os que avaliam o desempenho do processador, do sistema operacional e do sistema de arquivos.

Apachebench: Outra ferramenta utilizada para analisar o desempenho de um servidor *web*, porém esta apresenta um dado diferente daqueles apresentados pela *httpperf*, ela tem como resposta o número de requisições e o número de conexões simultâneas. É recomendável que ela seja utilizada em testes rápidos em servidores, pois esta ferramenta não é muito estável para testes muito complexos ou que demandem uma carga alta de requisições. O uso desta ferramenta está relacionado com as métricas: Número de conexões simultâneas e Retorno de arquivos. Os dados apresentados não indicam o número de conexões suportadas, mas sim os resultados obtidos sobre variados números de conexões. Dentre os diferentes resultados apresentados, os escolhidos para utilização na análise apresentam as seguintes unidades de medida: Tempo de duração das

conexões (médio, mínimo e máximo) em *milisegundos*; Nível de transferência em *KBytes* por segundo.

Autobench: Ferramenta de apoio ao *httperf*, durante a sua avaliação ela executa o *httperf* muitas vezes aumentando assim o número de requisições realizadas. O autobench é uma boa ferramenta para se realizar a comparação entre dois servidores *web*, ela é capaz gerar um comparativo entre os dois na mesma tabela, sendo que os dados apresentados por esta ferramenta são os mesmos da ferramenta *httperf*.

4.5 Definição dos testes

Definidas as ferramentas de apoio, é necessário definir quais os testes que serão executados durante o processo de análise. O objetivo destes testes é recolher dados em diferentes situações para que se tenha uma boa qualidade e quantidade de dados para se realizar a análise de custo/benefício da virtualização para consolidação de servidores.

Os testes serão realizados sobre um mesmo ambiente computacional, ou seja, o *hardware* utilizado e o ambiente de rede serão os mesmos em todos os testes. A diferenciação entre os testes virá do número de máquinas virtuais criadas pelo ambiente de virtualização, sendo que cada máquina será responsável por consolidar um servidor separado. A máquina na qual será implantado o servidor de forma nativa e os servidores virtualizados tem as seguintes características de *hardware*:

- Processador: AMD Sempron™ 2300+ 1900Mhz, 256K Cache
- Memória RAM total: 1,768 GB
- Disco Rígido: 40 GB

É importante definir também que os testes enviados pelos clientes para esta máquina servidora deverão ser realizados em um laboratório disponibilizado pela Universidade, uma vez que múltiplos clientes serão necessários para efetuar os testes.

Para que o resultado apresentado seja válido é necessário que o ambiente de testes seja idêntico em todas as realizações, para tanto, todas as máquinas virtuais devem manter as mesmas características de recursos. E as mesmas foram definidas para:

- Memória RAM total: 128 MB
- Disco Rígido Virtual: 2 GB

Antes de definir os ambientes de testes é necessário também realizar um teste inicial para se verificar a capacidade do *hardware* servidor e a possível perda de desempenho decorrente da inserção da ferramenta de virtualização, para tanto a abordagem apresentada na figura 12 será utilizada.

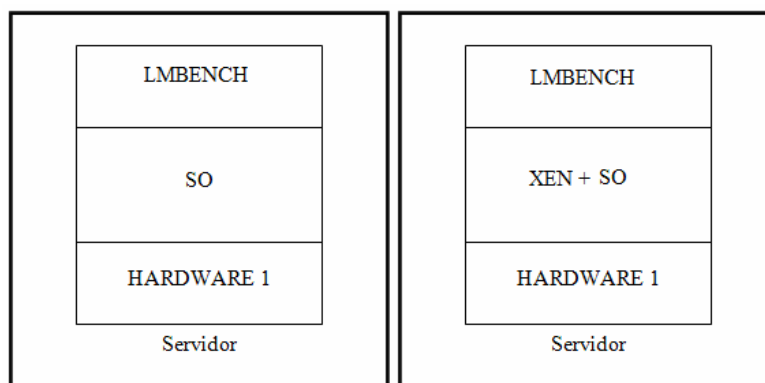


Figura 12: Testes sobre o *hardware*

Este teste será realizado para que dados sobre a capacidade do *hardware* sejam levantados, no exemplo apresentado pela figura 12 é possível perceber que a ferramenta *Lmbench* que executa sobre SO irá coletar os dados. A mesma execução será realizada sobre o ambiente no qual está inserida a ferramenta de virtualização, sendo que os dados obtidos devem ser comparados junto da análise final.

4.5.1 Ambiente de testes 1

O primeiro ambiente de testes definido terá como característica a não utilização de virtualização, este teste irá recolher os dados referentes à aplicação das ferramentas de apoio sobre um servidor *web* implementado diretamente sobre uma máquina física. Depois de obtidos os dados através das ferramentas, serão separados aqueles que são relevantes para a análise, ou seja, aqueles que correspondem às métricas citadas na seção 4.3. A arquitetura do ambiente de testes 1 pode ser visualizada na figura 13.

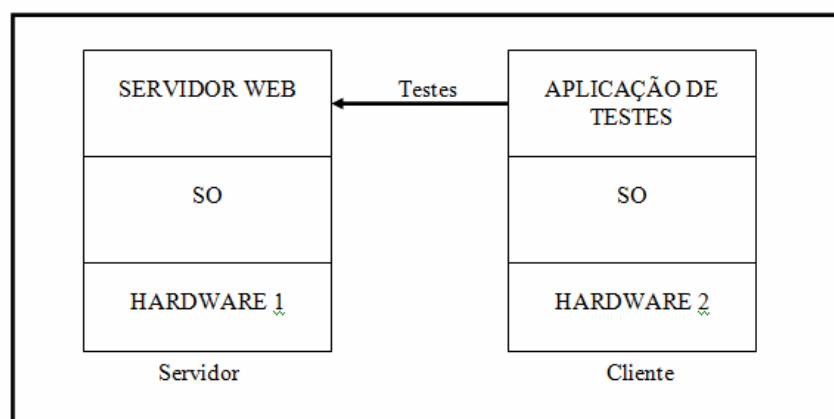


Figura 13: Ambiente de testes 1.

Na figura 13 tem-se dois *hardwares* distintos, o *hardware* primário (simbolizado por *hardware 1*) é aquele onde será implantado o servidor *web*, ou seja, esta é a máquina servidora, para tanto é necessária a execução de um sistema operacional que permita tal atividade. Já a máquina secundária é aquela onde estão instaladas as ferramentas de apoio, durante a sua execução as mesmas irão executar uma série de operações sobre o servidor, recolhendo assim os dados obtidos com as mesmas. Esta máquina secundária é denominada de cliente.

O objetivo de se utilizar tal abstração é obter dados concisos que servirão de referência durante a comparação com os demais testes realizados, portanto

pode-se considerar o primeiro teste como um marco inicial para realização da análise.

4.5.2 Ambiente de testes 2

O ambiente de testes 2 começa a utilizar o conceito de virtualização, sendo que durante a execução deste apenas uma máquina virtual será criada, ou seja, apenas um servidor *web* será virtualizado. Da mesma forma que no ambiente de testes 1, serão utilizadas duas máquinas físicas, onde uma delas (servidora) irá executar o ambiente de virtualização e a máquina virtual. Já a segunda máquina física irá desempenhar a mesma função daquela apresentada no primeiro ambiente de testes, ou seja, irá realizar testes sobre o servidor *web*. A arquitetura deste ambiente pode ser visualizada na figura 14.

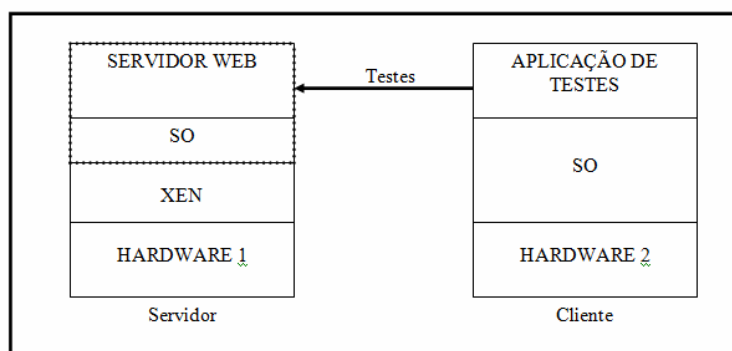


Figura 14: Ambiente de testes 2.

Ao observar a figura 14 pode se perceber que ela possui uma arquitetura similar àquela que foi apresentada para o ambiente de testes 1 entretanto, neste segundo ambiente, há a inserção de mais uma camada no servidor, onde é executado o ambiente de virtualização definido (XEN). Neste exemplo é criada apenas uma máquina virtual pelo ambiente de virtualização (representada pela caixa pontilhada), sendo que nela é executado o sistema operacional junto do servidor *web*. Da mesma forma que no ambiente de testes 1, a máquina cliente através das aplicações de teste realiza diversos testes ao servidor, recolhendo dados que serão utilizados posteriormente para análise.

O objetivo de utilizar apenas uma máquina virtual é que se possa coletar dados referentes a esta abstração e compará-los com os dados obtidos no primeiro ambiente para determinar se existe perda de desempenho para uma máquina virtual apenas. Percebe-se também, que esta abstração não traz economia de *hardware*, pois mantém-se a execução de apenas um servidor.

4.5.3 Ambiente de testes 3

O terceiro ambiente de testes mantém a camada de virtualização inserida sobre o *hardware*. A diferença em relação ao teste 2 é a execução de duas ou mais máquinas virtuais sobre o ambiente de virtualização e a presença de mais de um cliente na execução (um para cada servidor virtualizado), sendo que os diversos clientes deverão realizar múltiplos testes sobre os ambientes virtualizados. Os dados coletados serão utilizados para se estimar a perda de desempenho existente quando se compara a virtualização com diferentes números de máquinas virtuais. O exemplo desta abstração pode ser realizado na figura 15.

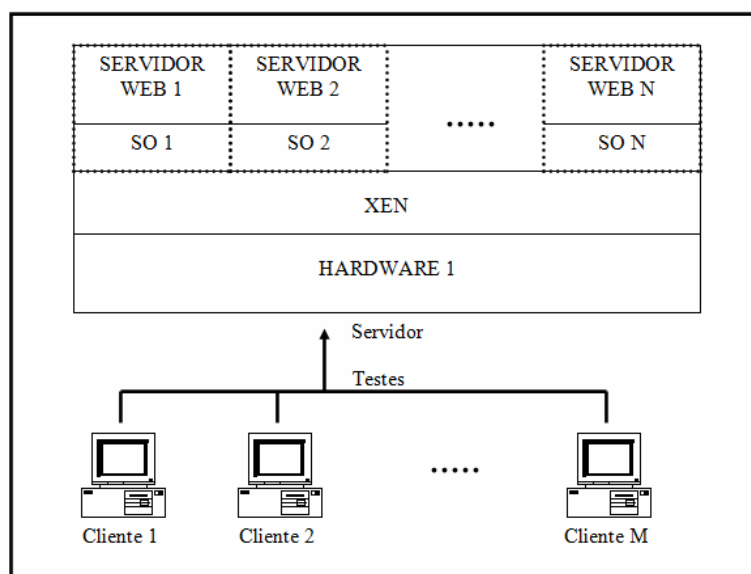


Figura 15: Ambiente de testes 3

Ao observar a figura 15 percebe-se que o número de máquinas virtuais criadas pelo ambiente de virtualização Xen (N) é de no máximo 6, sendo que para cada máquina virtual serão executadas um sistema operacional e um servidor *web*. Outro aspecto importante verificado é de que múltiplos clientes também serão necessários, onde cada cliente será responsável por realizar os testes sobre um servidor, o número de clientes (M) utilizados é 3.

O objetivo de se utilizar ambiente de testes com múltiplas máquinas é de que se possa perceber qual a real perda de desempenho e qual a proporção em que isto ocorre, para diferentes números em N (1,2,3) e M (1,2,4,6). Portanto, para este nível de testes serão coletadas informações referentes a 12 ambientes distintos.

4.6 Considerações

Este capítulo de métricas foi descrito em uma ordem cronológica, que se inicia com a definição do serviço testado para este trabalho, seguido da avaliação dos requisitos para este serviço. Também foram abordadas as métricas que estão relacionadas para estes requisitos e as ferramentas utilizadas para obter tais dados. Por fim, tem-se a descrição dos ambientes de testes realizados necessários para que se consiga o conjunto de informações relevantes para o trabalho.

Depois de realizada as propostas de testes é necessário realiza-los, o detalhamento da execução destes e a apresentação e avaliação dos resultados é realizada no capítulo 5, onde, depois de exemplificados os resultados, tem-se a análise final do custo-benefício da virtualização para consolidação de servidores.

5. RESULTADOS

Este capítulo tem como objetivo a descrição das características específicas dos testes realizados sobre os diferentes ambientes citados no capítulo 4, bem como apresentar os resultados obtidos com os mesmos, sendo que a análise final sobre o uso de virtualização para consolidação de servidores também é descrita ao final deste capítulo.

A descrição dos resultados obtidos será feita de acordo com os ambientes de testes propostos no capítulo 4, portanto este capítulo será dividido em cinco seções, a primeira irá descrever a metodologia utilizada na execução dos testes, as três seguintes descrevem os resultados obtidos, e a última seção irá apresentar a análise final sobre o tema deste trabalho.

5.1 Metodologia de execução

Uma metodologia específica foi definida para a realização dos testes, esta faz-se necessária, pois o resultado final deste trabalho é uma análise que deve ser baseada em números estáveis e idôneos. Para alcançar estes objetivos realizaram-se 10 testes distintos para cada servidor (físico ou virtual) e cada ferramenta, com o intuito de se alcançar valores equilibrados. O uso desta metodologia permitiu um desvio padrão de 10%, resultado este aceito estatisticamente para testes de elevado porte, como o que está em questão.

Além de definir um valor estatisticamente aceitável, também foi necessário manter o mesmo local de testes, uma vez que variações aqui poderiam modificar o

resultado final do trabalho. Portanto a execução de todas as aplicações de testes sobre os diferentes ambientes foi realizada no laboratório do grupo de pesquisa GRADIS, pertencente à Universidade do Estado de Santa Catarina.

Outro ponto importante definido nesta metodologia foi a utilização dos mesmos clientes (máquinas físicas) nos diferentes testes, com o intuito de manter invariáveis os dados, pois clientes com diferentes características poderiam afetar o desempenho dos mesmos.

Outra variável que poderia afetar o resultado dos testes é o sincronismo, a solução encontrada para solucionar tal questão, foi configurar os clientes com o mesmo horário, e utilizar chamadas programadas do sistema para executar os testes simultaneamente. Porém, não foi possível manter os clientes 100% sincronizados, pois, mesmo que os testes iniciassem juntos, eles terminavam em tempos distintos (devido à diferença de desempenho dos clientes), portanto para que os dados fossem aceitáveis, os melhores e os piores resultados obtidos nos testes foram eliminados, a fim de que se mantivessem dados estatisticamente corretos.

Por fim, é importante frisar também que as seis máquinas virtuais utilizadas possuíam características idênticas de recursos como memória, sistema operacional e espaço em disco. Para que o servidor utilizado nos testes, sem uso de máquina virtual, tivesse características idênticas às dos virtualizados, também foi mantida a mesma quantidade de memória, o mesmo sistema operacional e o mesmo aplicativo de serviço *web*. Como citado no capítulo 4, a quantidade de memória das máquinas virtuais testadas foi de 128 MB, o sistema operacional utilizado foi o Debian GNU/Linux 4.0 (Kernel 2.6.18), e o aplicativo de serviço *web* utilizado foi o *Apache* 1.3.

5.2 Comparação entre XEN e máquina nativa.

O primeiro teste foi realizado com o intuito de comparar o desempenho da máquina física em duas diferentes situações:

- Sem a inserção do ambiente de virtualização XEN (Máquina nativa);
- Com a compilação e instalação do ambiente.

Este teste foi feito para verificar a hipótese de que a inserção de uma camada entre o *hardware* e o sistema operacional (virtualização) por si só reduz o desempenho da máquina física.

Para verificar essa possível queda de desempenho foi utilizada a ferramenta de testes denominada *LMBench*, e dentre os resultados apresentados pela mesma, foram selecionados aqueles que avaliam os recursos de processador, sistemas de arquivos e sistema operacional [BARHAM,2003].

O gráfico com os dados para avaliação do sistema de arquivos nos dois diferentes ambientes pode ser observado na Figura 16.

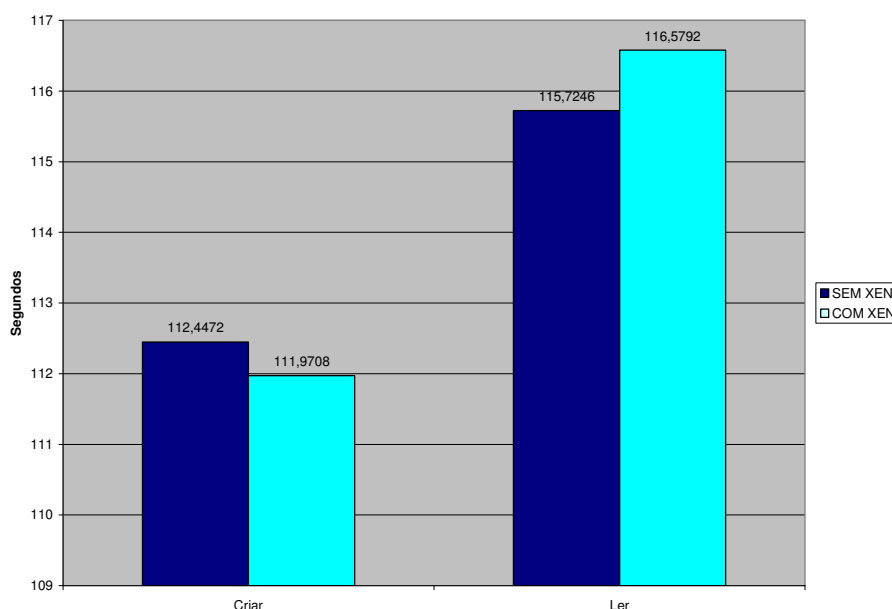


Figura 16: Desempenho sistema de arquivos

Na Figura 16 tem se a avaliação do desempenho do sistema de arquivos nos ambientes: com ambiente de virtualização representado pelos dados da

coluna em verde; e sem Xen representados pelas colunas em azul. Neste teste foi mensurado o tempo necessário para criar um arquivo de 3,5 GB, bem como realizar a leitura do mesmo.

Percebe-se que o tempo para criar o arquivo no ambiente 'SEM XEN' é de 112,45 segundos, valor este estatisticamente equivalente aos 111,97 segundos do ambiente 'COM XEN', uma vez que o desvio padrão obtido neste teste foi de 1,8 segundos. O tempo de leitura também é considerado equivalente entre os dois, uma vez que o ambiente 'SEM XEN' obteve a marca de 115,72 segundos e o 'COM XEN' a marca de 116,58 segundos. Portanto, o desempenho do sistema de arquivos no ambiente com a inserção da camada de virtualização é totalmente comparável ao desempenho daquele sem a inserção desta camada.

Para verificar o desempenho do processador foram mensurados os tempos gastos para realizar três diferentes processos:

- *fork+exit* que é o tempo gasto para dividir um processo em dois, com o processo filho encerrando imediatamente sua execução.
- *fork+execve* neste caso é o tempo gasto para criar um novo processo e fazer com que o mesmo execute um programa do tipo 'hello world';
- *fork + /bin/sh -c* onde é mensurado o tempo utilizado para criar um novo processo que invoca o *shell* para que este execute um programa. Este corresponde à criação e execução de um processo complexo, se comparado aos anteriores.

Os resultados destes testes são representados pelo gráfico representado na Figura 17.

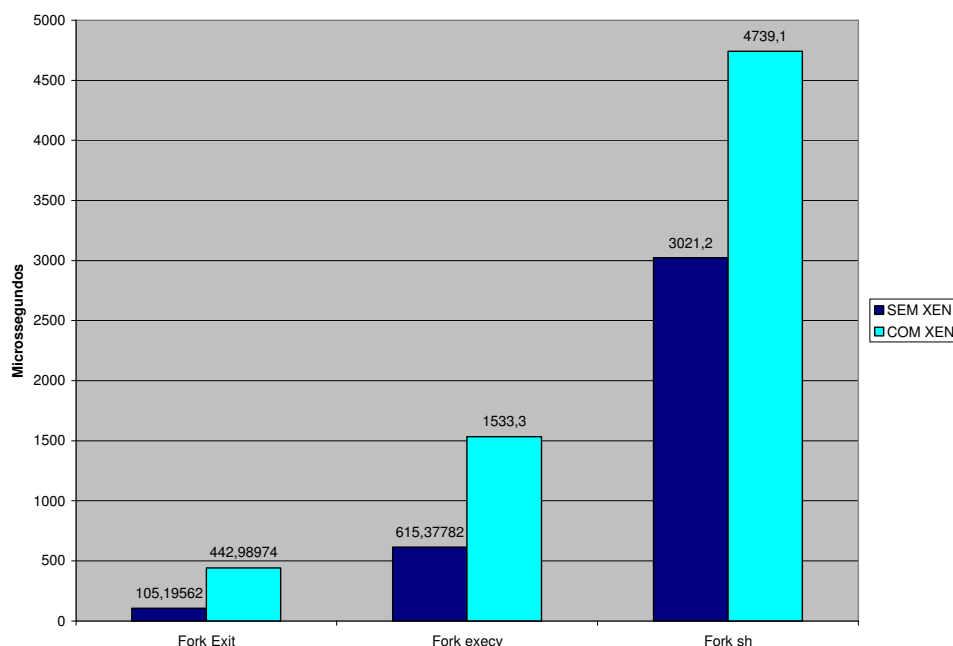


Figura 17: Desempenho da criação de processos

Ao observar a Figura 17 pode-se perceber que o desempenho das chamadas *fork* cai drasticamente após a inserção da ferramenta de virtualização (dados representados pela cor verde), é possível observar esta queda nos três diferentes tipos de execução. No teste com *fork+exit*, o tempo para execução no ambiente 'SEM XEN' (azul) é de 105,19 microssegundos, e no ambiente 'COM XEN' (verde) a mesma execução é realizada em 442,99 μs. O mesmo desequilíbrio segue nos outros testes, o ambiente 'SEM XEN' executou o *fork+execv* em 615,38 microssegundos, e o ambiente 'COM XEN' a mesma execução é dada em 1533,3 microssegundos. Por fim, o teste *fork+sh* obteve nos ambientes 'SEM XEN' e 'COM XEN' respectivamente 3021,2 e 4739,1 microssegundos.

Ao analisar estes resultados é possível afirmar que a inserção da ferramenta XEN reduz drasticamente o desempenho da criação de processos no sistema, em algumas situações o mesmo é reduzido em até três vezes do valor do ambiente sem a inserção da ferramenta. Portanto, serviços que envolvam a

criação freqüente de processos podem esperar uma degradação no seu desempenho quando executarem sobre o Xen.

Além das chamadas *fork*, outras métricas também foram utilizadas para avaliar o desempenho do processador, principalmente com foco na comunicação realizada entre os processos. A primeira avaliação é realizada sobre o mecanismo *pipe*, onde a comunicação é realizada de forma unilateral, e o tempo avaliado é resultante da operação de criação de dois canais de comunicação e o envio e retorno de uma palavra nesses canais.

A segunda avaliação é feita sobre o mecanismo *signal handler*, que trata sinais de comunicação, enviado para um processo informando que o mesmo deve lidar com um evento. Para este mecanismo são avaliados dois itens, o tempo utilizado para instalar um tratador de sinal (*signal Handler Installation*), e o tempo para realizar a expedição do sinal em dois *loops* no mesmo processo, denominado *signal handler overhead*.

No entanto, se faz necessária a avaliação do desempenho do processador durante a execução de processos, para tanto são avaliados dados referente às operações entre variáveis, sendo que os tipos de operações e os tipos de variáveis definidas para comparação são a multiplicação e a divisão entre valores do tipo inteiro e tipo *float*.

A Figura 18 apresenta o gráfico com os resultados sobre os testes com o mecanismo *pipe*, realizado com o apoio da ferramenta *LMBench*.

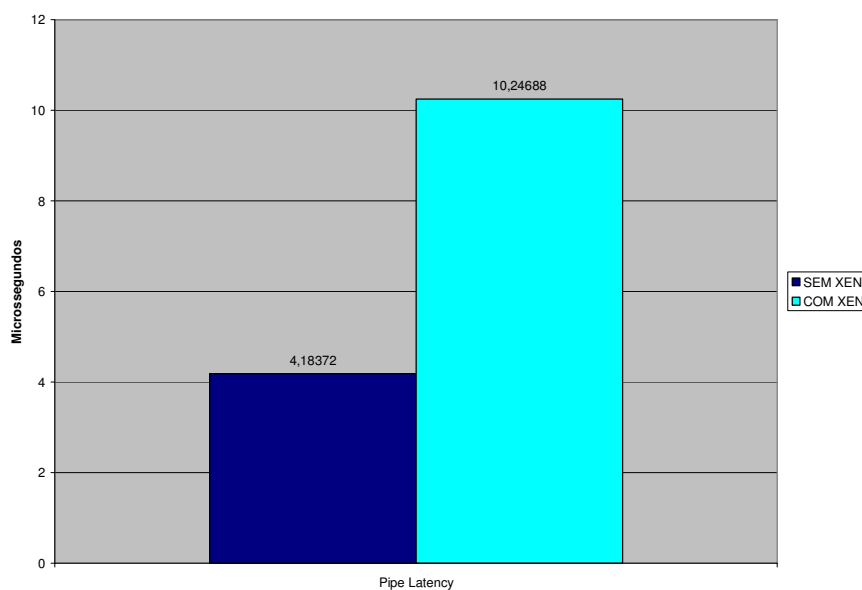


Figura 18: Desempenho Pipe

Ao analisar a Figura 18 é possível perceber que o ambiente 'SEM XEN', em azul, executa o teste em 4,18 microssegundos, e com desempenho muito inferior o ambiente 'COM XEN' executa o mesmo teste em 10,25 microssegundos. Pode-se então afirmar que o recurso de comunicação entre processos sofre uma queda de desempenho quando se é inserida a ferramenta de virtualização XEN.

Na Figura 19 têm-se o gráfico com os resultados obtidos para os testes com *signal handlers*.

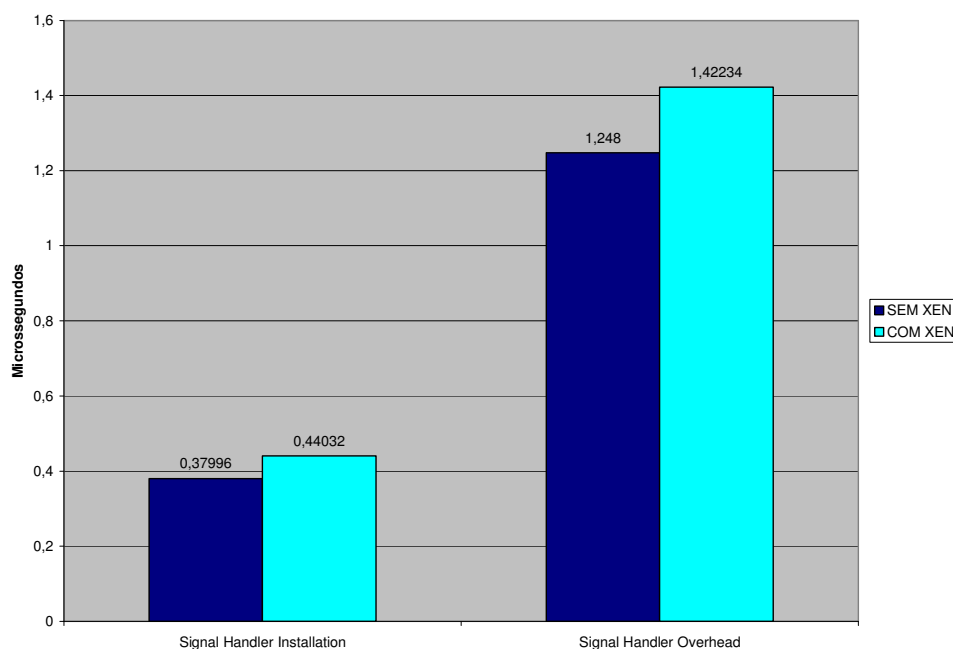


Figura 19: Desempenho *signal handler*

Analisando os dados apresentados pela Figura 19, pode-se perceber que o desempenho apresentado pelo ambiente 'COM XEN', em verde, é levemente inferior ao ambiente 'SEM XEN', em azul. Os dados obtidos indicam que o tempo para criação do tratador de sinal (*Signal Handler Installation*) no ambiente sem XEN é de 0,38 microssegundos, e no ambiente com XEN o mesmo teste é executado em 0,44 microssegundos. Ao avaliar o tempo para realizar a expedição do sinal, também se percebe a leve queda de desempenho, enquanto o ambiente 'SEM XEN' realiza o teste em 1,25 microssegundos, o ambiente 'COM XEN' realiza o mesmo teste em 1,42 microssegundos.

O próximo item avaliado está relacionado à execução de processos, e os resultados obtidos podem ser visualizados na figura 20.

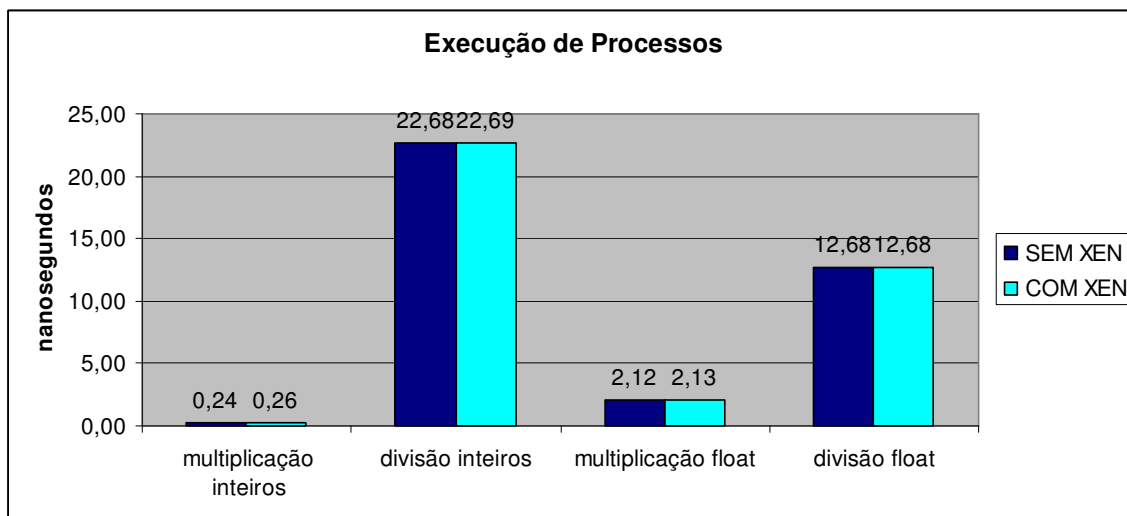


Figura 20: Desempenho para execução de processos

Ao se avaliar os resultados apresentados pela figura 20, pode-se perceber que o desempenho obtido para os ambientes 'SEM XEN' (em azul) e 'COM XEN' (em verde) se mantém equivalente para as quatro operações testadas, onde alguns processos como a divisão de *float* (realizado em 12,68 nanossegundos) apresentou resultados idênticos para os dois ambientes e nas demais operações percebeu-se variações praticamente imperceptíveis.

Portanto, ao avaliar o recurso de processador pode-se afirmar que, quando se fala em criação e comunicação de processos, existe uma queda de desempenho na inserção da ferramenta XEN, porém, o desempenho da execução de processos não é interferido pela inserção da mesma.

Outro recurso que deve ser avaliado é o desempenho do sistema operacional e esta avaliação é feita através da execução de três chamadas básicas do sistema:

- *simple syscall*, mensura o tempo gasto para realizar uma chamada do sistema;
- *simple stat*, avalia o tempo necessário para solicitar o *status* de um arquivo;
- *simple open/close*, informação referente ao tempo necessário para abrir e fechar um arquivo sem realizar nenhum acesso a ele.

Os resultados referentes aos testes sobre os recursos do sistema operacional podem ser observados na Figura 21.

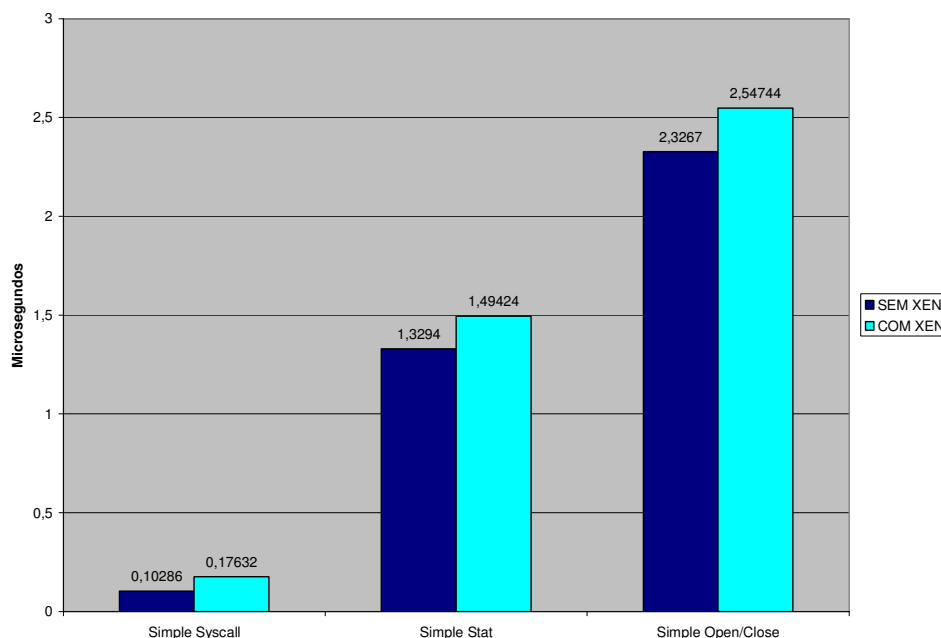


Figura 21: Desempenho sistema operacional.

Os dados da Figura 21 indicam que o desempenho do ambiente 'COM XEN', em verde, é levemente inferior àquele apresentado pelo ambiente 'SEM XEN', em azul. Os dados do teste da chamada do sistema apontam que o ambiente 'SEM XEN' executa este teste em 0,10 microssegundos, e o ambiente 'COM XEN' realiza o mesmo teste em 0,18 microssegundos. Esta diferença se mantém na avaliação do teste sobre a solicitação do *status* de um arquivo onde os valores são para 'SEM XEN' e 'COM XEN', respectivamente, 1,33 microssegundos e 1,49 microssegundos. O mesmo ocorre para o teste de abertura e fechamento de arquivo, onde os valores indicados são de 2,33 microssegundos para ambiente 'SEM XEN' e 2,55 para ambiente 'COM XEN'. Portanto a queda de desempenho pode ser considerada leve, não afetando bruscamente o desempenho do sistema operacional.

Ao concluir a comparação entre o ambiente onde a ferramenta de virtualização está inserida com o ambiente nativo, pode-se afirmar que existe sim uma queda de desempenho, resultado este já esperado, uma vez que a camada

de virtualização inserida interfere, de alguma forma, sobre o funcionamento normal do sistema. Sendo que a queda é mais significativa quando se fala no recurso de processador, mais especificamente na criação de processos, portanto é válida a indagação de que aplicações que necessitem de muita criação de processos preferivelmente não devem ser utilizadas junto à ferramenta XEN, porém, foi visto que a execução dos mesmos não sofre interferência de desempenho.

5.3 Desempenho de rede nos diferentes ambientes

Depois de analisada a diferença de desempenho, em aspectos de processamento, proveniente da inserção da ferramenta XEN em um servidor nativo, é necessário avaliar qual o desempenho da *interface* de rede nos diferentes ambientes testados. Para isso, nesta seção será avaliada a transferência total em 10 segundos e o consumo de banda obtido nos ambientes: sem virtualização, com uma máquina virtual, com duas máquinas virtuais, com quatro máquinas virtuais e com seis máquinas virtuais.

Esta medição é realizada através de três diferentes cenários e com o uso da ferramenta *lperf*. No primeiro cenário os testes são realizados através de um cliente, no segundo cenário os testes são duplicados, ou seja, através de dois clientes, e no terceiro cenário os testes são feitos com três clientes. É válido lembrar que o número de clientes é igual para cada servidor (nativo ou virtual); por exemplo, no ambiente com seis máquinas virtuais cada cliente efetua seis testes, um para cada máquina virtual.

5.3.1 Cenário com um cliente

Para execução dos testes neste cenário, foram utilizados dois computadores, a máquina servidora e máquina do cliente, sendo que a ferramenta de testes é executada no cliente e dispara múltiplos acessos para o servidor. É importante mencionar que o servidor foi testado com cinco ambientes diferentes (sem virtualizar, uma máquina virtual, duas máquinas virtuais, quatro máquinas virtuais e seis máquinas virtuais). Os resultados obtidos durante a execução de testes neste cenário estão apresentados pelos gráficos das Figuras 22 e 23.

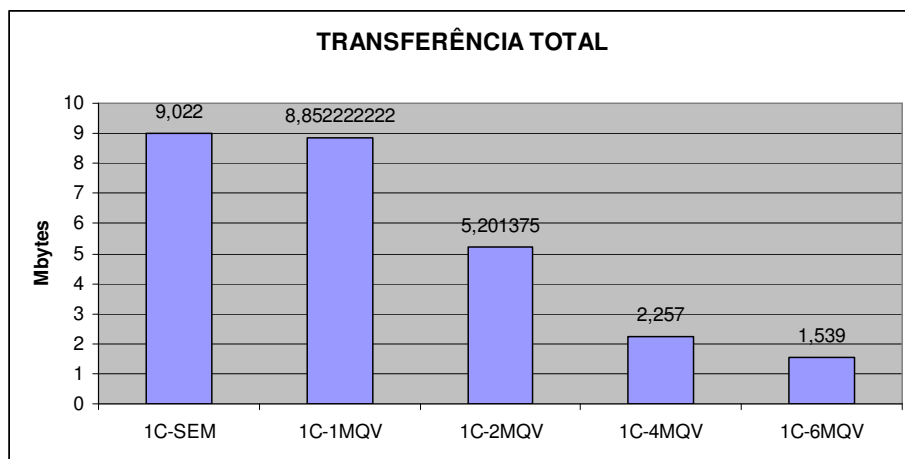


Figura 22: Transferência total para 1 cliente

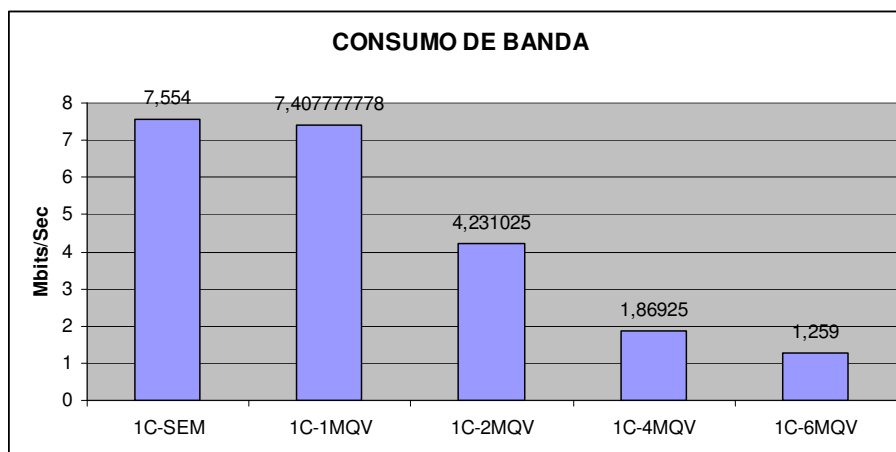


Figura 23: Consumo de banda para 1 cliente

Ao observar as Figuras 22 e 23 é possível perceber que o desempenho da rede nos dois itens avaliados se mantém bastante próximo entre os ambientes sem virtualização (1C-SEM) e com a execução de uma máquina virtual (1C-1MQV), e o mesmo cai linearmente quando são executadas mais de uma máquina virtual. Ao se considerar o desvio padrão de 10% nos testes pode-se afirmar que os recursos de rede avaliados são divididos igualmente para cada cliente, por exemplo, o resultado obtido para o consumo de banda com uma máquina virtual foi de 7,41 Mbits/seg, já o consumo para duas máquinas virtuais foi de 4,23 Mbits/seg, considerando-se o desvio, pode se avaliar uma divisão aproximadamente pela metade, o que indica que o ambiente de virtualização possui uma boa capacidade de controlar os recursos de rede em testes com um cliente.

5.3.2 Cenário com dois clientes

Os testes executados neste cenário seguem os mesmos princípios utilizados no cenário com um cliente, a diferença é que agora serão utilizados três computadores, um para o servidor que é responsável por abrigar as máquinas virtuais, e dois para os clientes que são responsáveis por executar os múltiplos

testes com a ferramenta *Iperf*. Os dados obtidos para este cenário podem ser visualizados nas Figuras 24 e 25, que indicam, respectivamente, transferência total e consumo de banda.

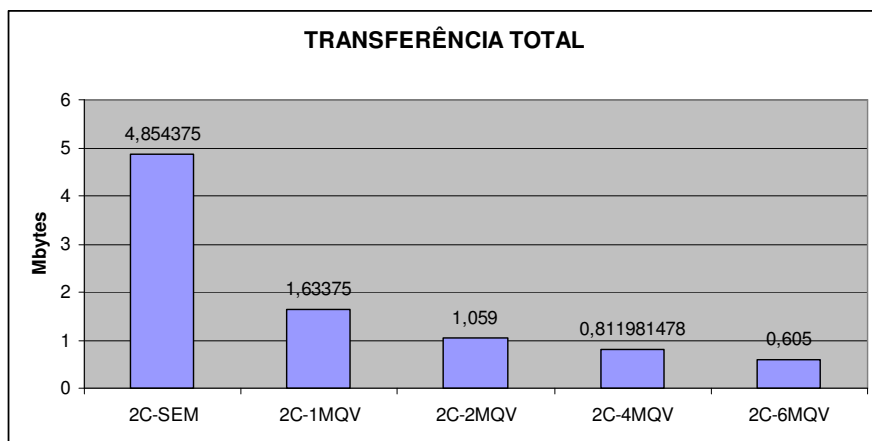


Figura 24: Transferência total para 2 clientes

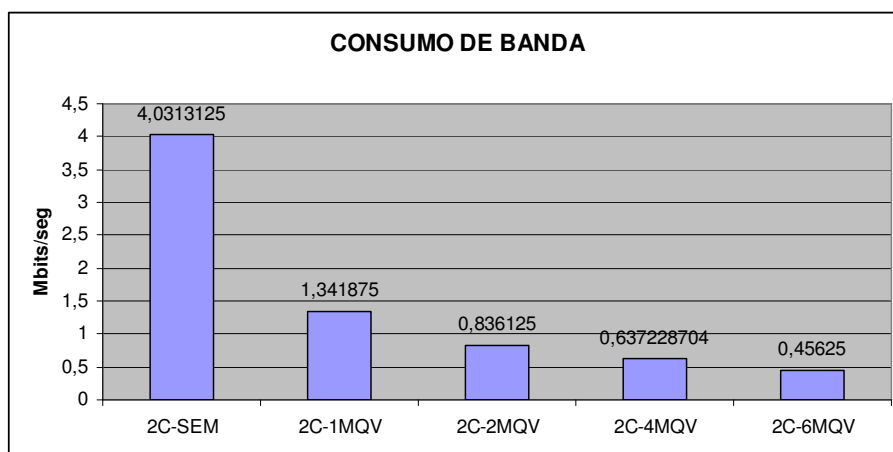


Figura 25: Consumo de banda para 2 clientes

Ao analisar as Figuras 24 e 25, é possível perceber que a inserção de mais um cliente na realização dos testes reduz abruptamente o desempenho dos recursos de rede, se comparado ao cenário com um cliente. Percebe-se que, sem a utilização de ambiente de virtualização o desempenho cai relativamente pela metade, o consumo de banda para o servidor nativo (2C-SEM) cai para 4,03 Mbits/seg, enquanto o teste para um cliente apresentava um valor igual a 7,41 Mbits/seg, valor normal, uma vez que o sistema divide o recurso pelos dois clientes. Porém, ao realizar o mesmo teste com uma máquina virtual o

desempenho cai de forma bastante relevante, atingindo a marca de 1,34 Mbits/seg de consumo de banda. No entanto, observa-se que ao se inserir novas máquinas virtuais a queda de desempenho não é tão abrupta, o que indica que o ambiente de virtualização consegue manter os recursos de rede equilibrados entre as diferentes máquinas virtuais.

5.3.3 Cenário com três clientes

A metodologia dos testes se repete para o cenário com três clientes, a diferença é que aqui se acrescenta mais um cliente para realizar acesso aos servidores (virtuais ou físicos). Os resultados também são dados por transferência total e consumo de banda, e podem ser visualizados, respectivamente, nas Figuras 26 e 27.

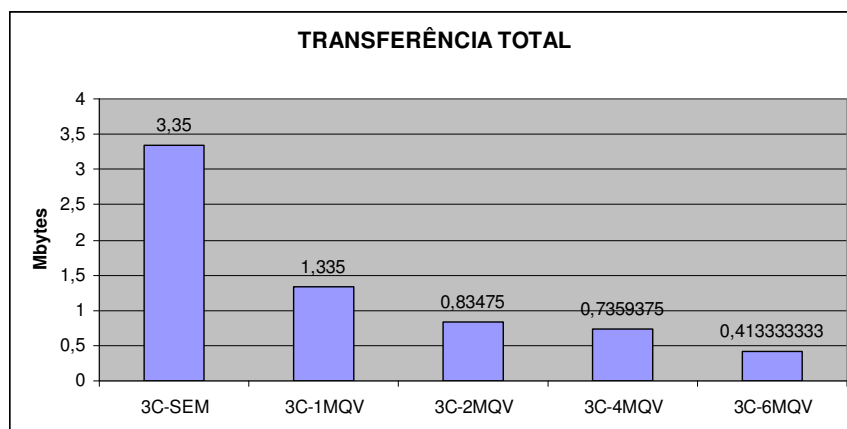


Figura 26: Transferência total para 3 clientes

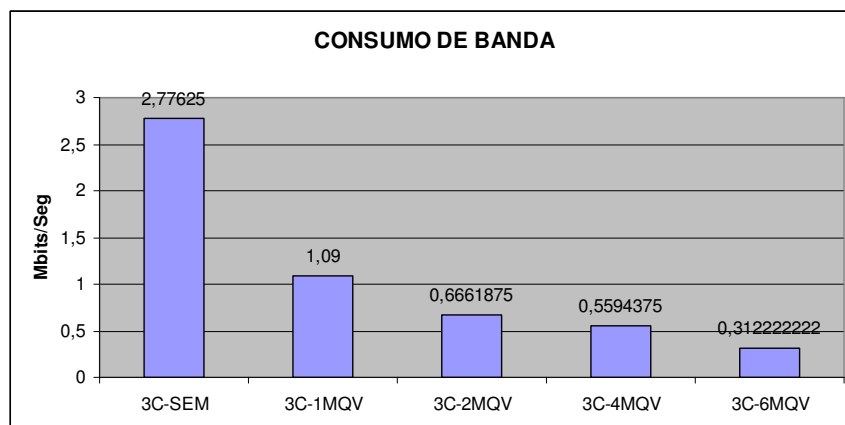


Figura 27: Consumo de banda para 3 clientes

Ao analisar as Figuras 26 e 27 é possível perceber que a inserção de mais um cliente nos ambientes de testes não interfere tão incisivamente no desempenho, quando se comparado ao cenário com dois clientes. Porém a queda abrupta de desempenho ainda se mantém ao se comparar os ambientes sem virtualização (3C-SEM) e com uma máquina virtual (3C-1MQV), que apresentaram respectivamente, 2,78 Mbits/seg e 1,09 Mbits/seg para a avaliação de consumo de banda. Percebe-se também que o mesmo nível de queda é mantido quando são inseridas duas máquinas virtuais, onde o índice atinge 0,67 Mbits/seg. No entanto o nível de queda se mantém equilibrado para o ambiente com quatro (0,56 Mbits/seg) e seis máquinas (0,31 Mbits/seg).

Portanto, percebe-se que a ferramenta XEN consegue equilibrar os recursos de rede entre os diferentes clientes e máquinas virtuais, fazendo que se mantenha um nível equilibrado de desempenho para as mesmas.

5.4 Desempenho do servidor *web* nos diferentes ambientes

Depois de analisado o desempenho do ambiente de virtualização XEN e o desempenho do recurso de rede nos diferentes ambientes, é necessário avaliar os resultados obtidos com os testes realizados sobre o serviço de servidor *web*, pois este foi o serviço definido para consolidação neste trabalho. Para tanto, foram utilizadas duas ferramentas específicas, o *ApacheBench* e o *Autobench*. Com a primeira foram obtidos os dados referentes ao tempo total dos testes (em segundos), ao número de requisições suportadas (nº/segundo) e a taxa de transferência do servidor *web*.

A segunda ferramenta, além de mensurar o número de requisições e a taxa de transferência, também mede a taxa de conexão (conexões por segundo), a taxa de resposta (respostas por segundo), o tempo de resposta (em milissegundos), e o número de erros. O uso dos dados referente a esta ferramenta, se faz necessário porque ela exige uma capacidade maior de processamento para sua execução, portanto a mesma foi capaz de saturar o uso do processador do servidor durante alguns testes. É válido frisar que para esta ferramenta, os testes ocorrem em tentativas de requisições por segundo que variam de 200 a 2000, e os dados utilizados são aqueles onde se tem certeza de que há o sincronismo, ou seja, no início (200 requisições por segundo) e na metade dos testes (1000 requisições por segundo).

É necessário ressaltar que os dados obtidos com a ferramenta *Httpperf* não serão analisados, pois eles se utilizam das mesmas métricas que a ferramenta *Autobench*.

É importante frisar que a apresentação dos dados obtidos nos testes será feita da mesma forma como aqueles apresentados pela seção 5.3, ou seja, classificados de acordo com o número de clientes que executam as aplicações sobre os servidores (nativo ou virtuais).

5.4.1 Cenário com um cliente

Os resultados apresentados nesta seção estão relacionados aos testes efetuados sobre o servidor através de um cliente, que dispara execuções das aplicações de testes para cada servidor (nativo ou virtual). Primeiramente, serão apresentadas as informações obtidas com a ferramenta *ApacheBench*, em seguida, tem-se as informações referentes aos testes com a ferramenta *Autobench*.

A Figura 28 traz um gráfico que compara o tempo total dos testes com a ferramenta *ApacheBench* nos cinco diferentes ambientes deste cenário, onde este tempo total está relacionado à execução da ferramenta, que dispara 100 requisições e simula 5 clientes simultâneos.

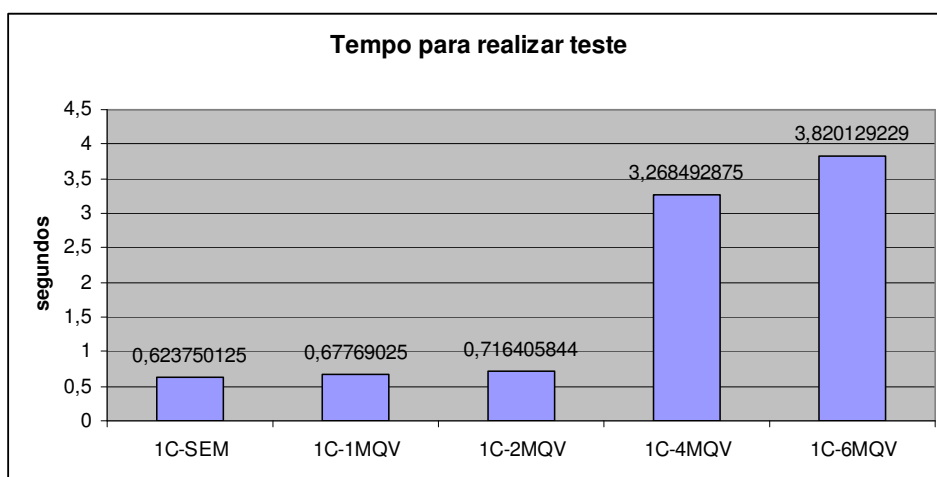


Figura 28: Tempo total *ApacheBench* com 1 cliente

Ao analisar a Figura 28 é possível perceber que houve um equilíbrio quanto ao tempo utilizado pelos ambientes: sem virtualização (1C-SEM); com uma máquina virtual (1C-1MQV); e com duas máquinas virtuais (1C-2MQV). Estes realizaram os testes em respectivamente: 0,62 segundos; 0,68 segundos; e 0,72 segundos. No entanto, após a execução do teste sobre quatro máquinas virtuais (1C-4MQV), observou-se uma grande queda de desempenho, o tempo total para a

execução do teste foi de 3,27 segundos, e como esperado o baixo desempenho se manteve para os testes com seis máquinas virtuais.

Outro dado avaliado pela ferramenta *ApacheBench* é o número de requisições por segundo suportadas pelo servidor, e os resultados obtidos podem ser visualizados no gráfico apresentado pela Figura 29.

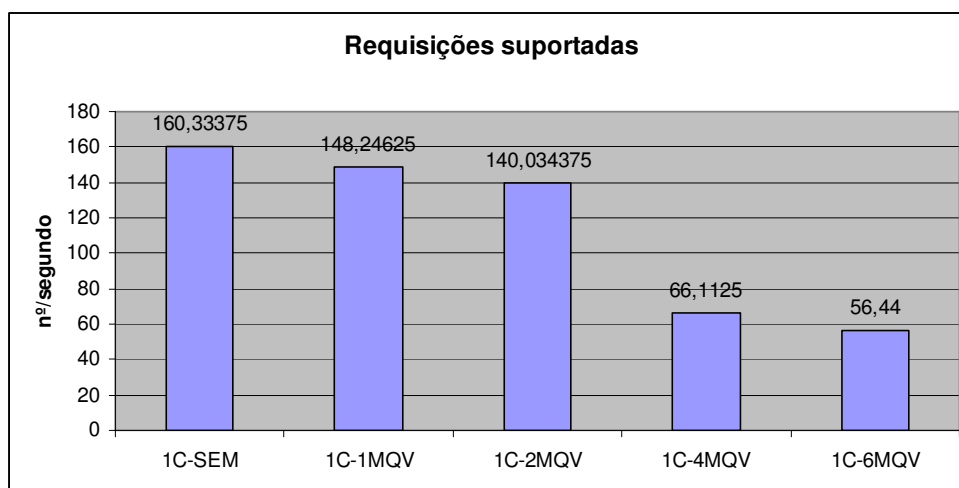


Figura 29: ApacheBench: requisições suportadas para 1 cliente

Com os dados obtidos com a figura 29, assim como no tempo total para realizar, é possível perceber que os resultados se mantêm com baixa queda até o ambiente com duas máquinas virtuais (1C-2MQV), sendo que a queda maior é percebida no ambiente com quatro máquinas virtuais (1C-4MQV). Portanto, mais uma vez, identifica-se como aceitável a queda de desempenho para o uso de duas máquinas virtuais. Porém é válido lembrar que estes testes são realizados para acessos de um cliente, onde o servidor não está saturado.

O último teste realizado com a ferramenta *ApacheBench* avalia a taxa de transferência do servidor *web*, os resultados deste teste podem ser visualizados no gráfico da Figura 30.

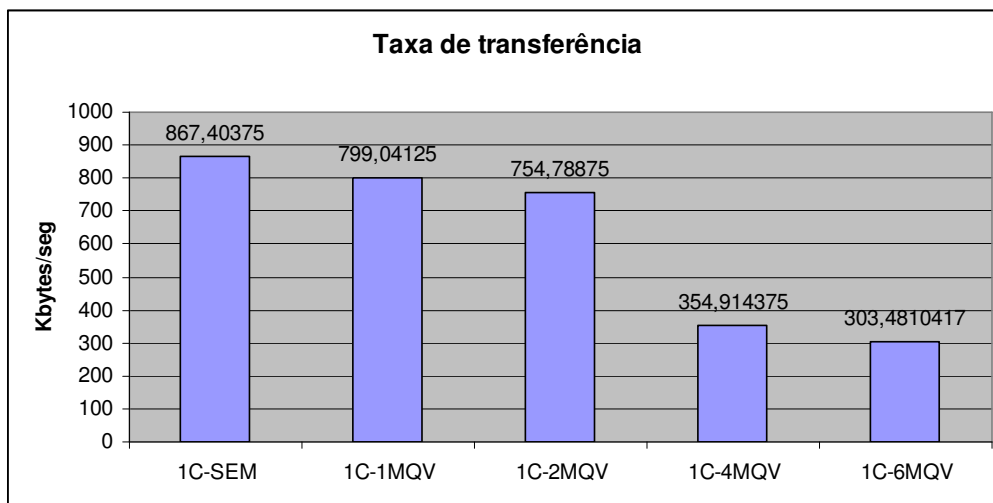


Figura 30: *ApacheBench* - Taxa de transferência para 1 cliente

Ao observar a Figura 30, percebe-se que os dados apresentados pelo teste mantêm o padrão dos outros testes executados pela ferramenta, ou seja, até a utilização de duas máquinas virtuais é possível observar uma leve queda de desempenho, e a queda maior deste desempenho ocorre com os resultados para quatro máquinas virtuais.

Depois de observados os resultados com a ferramenta *ApacheBench* é necessário avaliar os resultados obtidos com a ferramenta *Autobench*, que impõem um maior custo de processamento para o servidor. Durante a sua execução são disparadas 5000 conexões simultâneas, com a transferência de um arquivo de 10Kbytes, sendo que para cada execução, a ferramenta repete o mesmo teste 10 vezes. O primeiro resultado obtido com a ferramenta é a taxa de requisição obtida nos cinco ambientes.

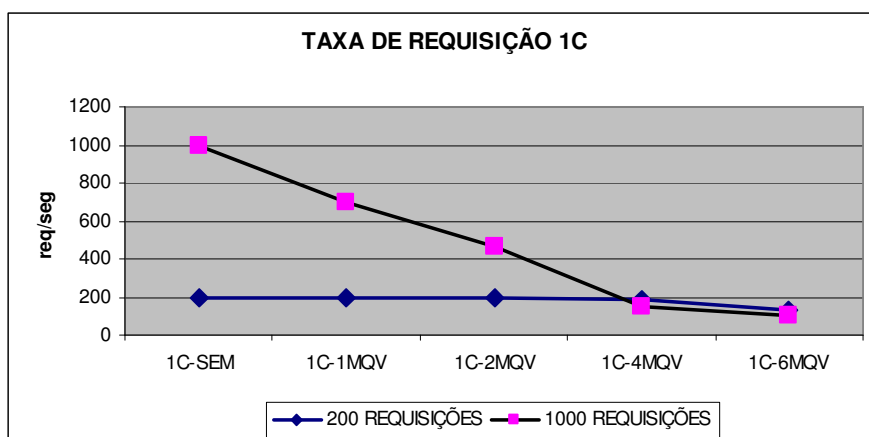


Figura 31: *Autobench*: taxa de requisição para 1 cliente

Ao analisar os dados apresentados pela Figura 31, é possível observar que os valores recuperados são aqueles obtidos para tentativa de 200 requisições (linha azul) e também, para tentativa de 1000 requisições (linha preta). Percebe-se que em níveis de 200 requisições, o desempenho se mantém equilibrado para todos os ambientes, já a avaliação de 1000 requisições, o desempenho cai linearmente, e se iguala à linha de 200 requisições ao utilizar quatro servidores virtuais (1C-4MQV). Percebe-se que o desempenho nesta situação é tão baixo que se equipara ao teste com 200 requisições. Ou seja, com quatro máquinas virtuais o nível de requisições que o servidor consegue suportar é equivalente a aquele apresentado para os testes com 200 requisições.

Outra medida de desempenho avaliada pela ferramenta é a taxa de conexão, os dados obtidos da execução deste teste podem ser observados na Figura 32.

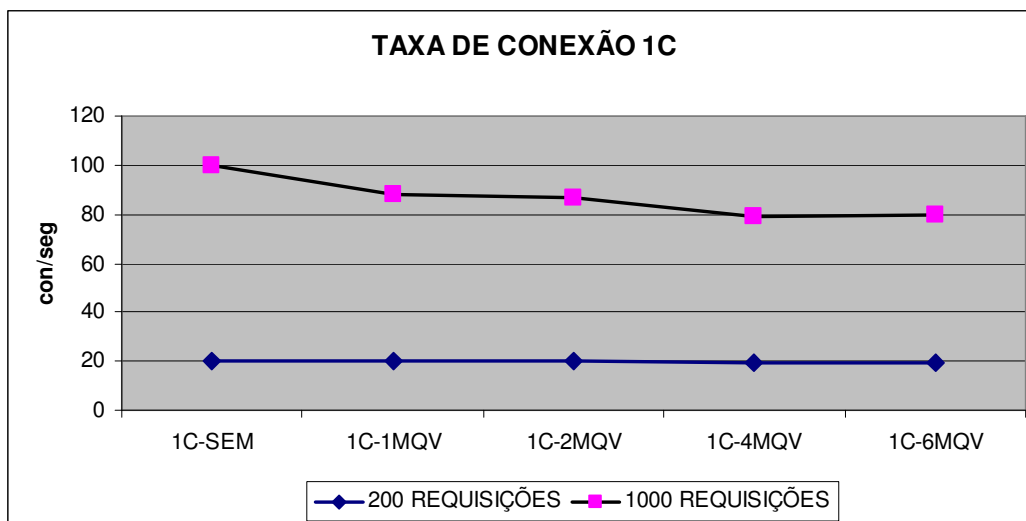


Figura 32: Autobench: Taxa de conexão para 1 cliente

Ao analisar o número de conexões por segundo suportadas pelo servidor, informação contida Figura 32, é possível definir que esta métrica não sofre grande interferência do uso de máquinas virtuais, uma vez que a queda de desempenho ocorrida do ambiente sem virtualização (1C-SEM) até o ambiente com seis máquinas virtuais (1C-6MQV) é de apenas 20%, considerando-se o desvio padrão de 10% pode-se dizer que a queda é mínima, tanto para a abordagem de 200, quanto para a abordagem de 1000 requisições.

Outro item escolhido entre os avaliados pela ferramenta é a taxa de resposta, mensurada em respostas por segundo. Os dados obtidos com este índice podem ser observados na Figura 33.

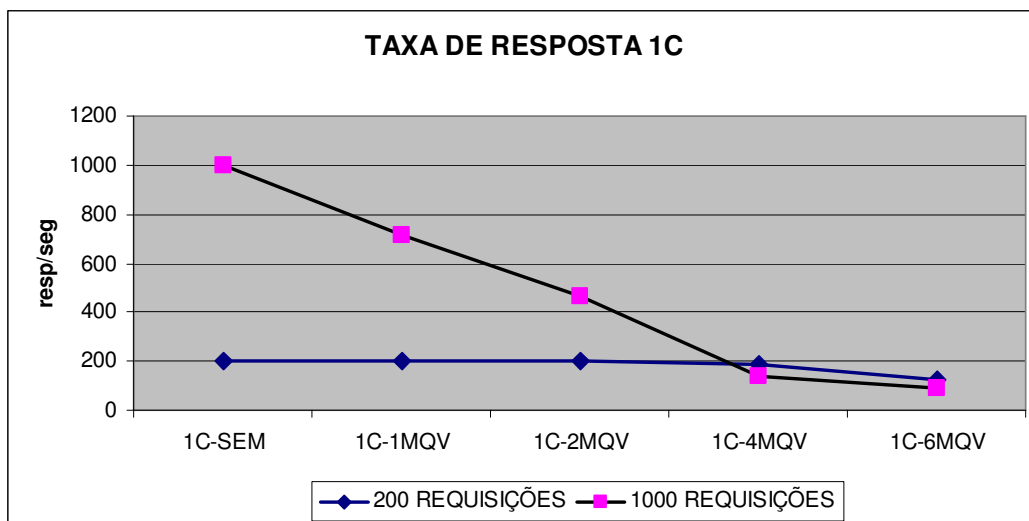


Figura 33: Autobench: taxa de resposta para 1 cliente.

Ao analisar os resultados apresentados pela Figura 33, é possível perceber que as linhas de desempenho seguem o mesmo comportamento dos resultados apresentados para o teste de taxa de requisição, onde os testes com 200 requisições simultâneas (linha azul) apresentam resultados estáveis e equilibrados para todos os ambientes. Percebe-se que o comportamento para 1000 requisições (linha preta), mantém uma queda de desempenho linear, e a curva de 1000 requisições se equivale estatisticamente à outra curva no ambiente com quatro máquinas virtuais.

A avaliação do tempo de resposta também é importante quando se fala de desempenho de servidor *web*, uma vez que está diretamente relacionado à percepção do usuário. Os resultados obtidos desta avaliação são representados pelo gráfico da Figura 34.

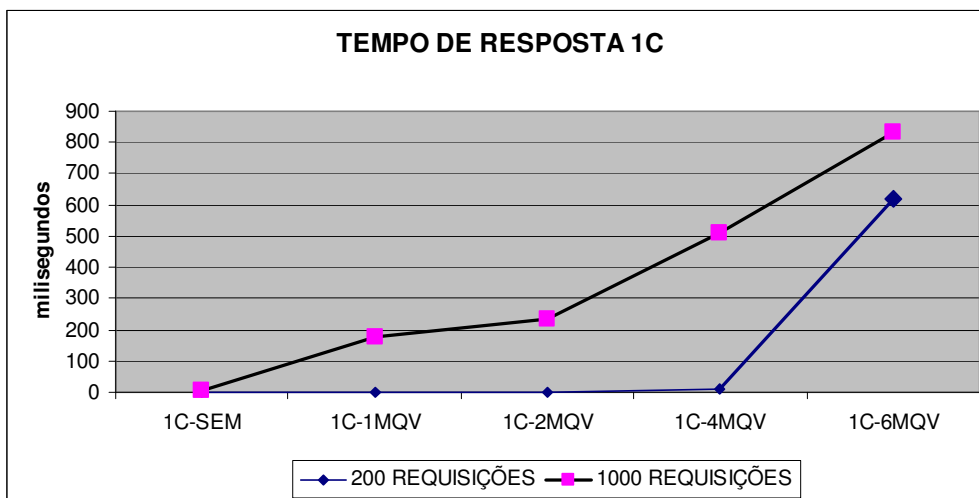


Figura 34: *Autobench*: Tempo de resposta para 1 cliente.

A Figura 34 apresenta o tempo de resposta atingido nos diferentes ambientes em duas situações, na primeira são executadas 200 requisições simultâneas (linha azul) e na segunda são executadas 1000 requisições simultâneas (linha preta). Percebe-se que o resultado dado em milissegundos (ms) varia de acordo com o número de máquinas virtuais utilizadas, e a queda de desempenho ocorre de forma diferente para os dois números de requisições. Para 200 requisições, esta queda é imperceptível até o ambiente com quatro máquinas virtuais, e é com seis máquinas virtuais que esta queda ocorre drasticamente. No entanto, com o número de requisições igual a 1000, a queda ocorre linearmente, enquanto o tempo de resposta para o ambiente sem virtualização (1C-SEM) é nulo, o mesmo teste revela que para uma máquina virtual o tempo é de 200 ms, para duas máquinas virtuais este tempo é de 250 ms, para quatro máquinas é de 500 ms e finalmente com seis máquinas este tempo é de aproximadamente 800 ms.

Contudo, apesar de apresentar uma queda, é possível afirmar que este não chega a ser um mau resultado, pois a queda não ocorre exponencialmente, mas sim linearmente, isto significa que o ambiente de virtualização consegue entregar de forma equilibrada os recursos às máquinas virtuais.

Além dos dados já obtidos com a ferramenta *Autobench*, para testes com um cliente, outros dados ainda devem ser avaliados, como o número de erros

apresentados e o consumo de banda. Portanto, a Figura 35 apresenta os resultados obtidos para a medida de consumo de banda nos servidores *web*.

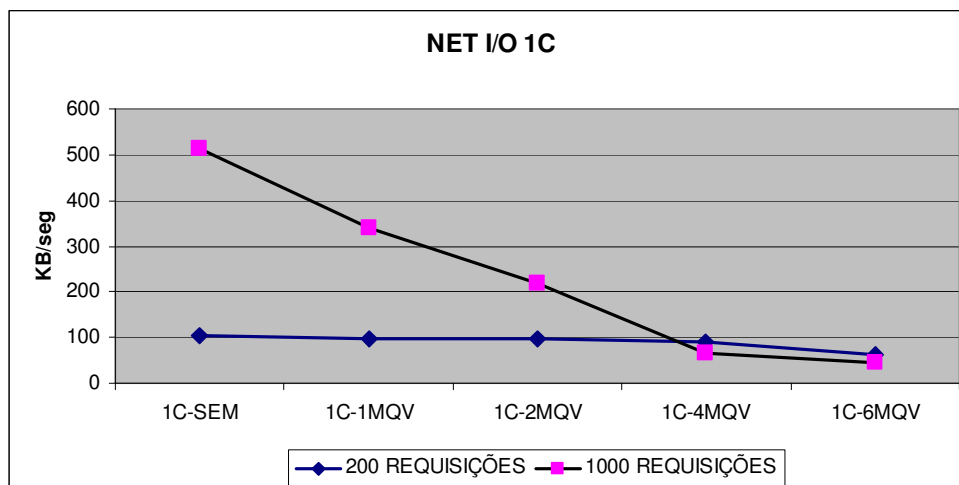


Figura 35: Autobench: Consumo de banda para 1 cliente.

Ao observar os comportamentos das curvas de desempenho na métrica apresentada pela Figura 35, se faz uma associação direta com o comportamento das métricas que avaliavam a taxa de resposta e a taxa de requisição, ou seja, o comportamento das curvas é semelhante. Neste teste é avaliado o consumo de banda do servidor *web* e o mesmo foi mensurado em Kbytes por segundo. Mais uma vez percebe-se que os resultados obtidos para 200 requisições (linha azul) seguem equilibrados em todos os ambientes. A degradação do desempenho somente é percebida quando ocorre uma maior saturação do servidor, e esta ocorre com os testes em 1000 requisições, onde esta curva se equivale com a curva de 200 requisições no ambiente com quatro máquinas virtuais (1C-4MQV), sendo que as duas permanecem com equivalência estatística para o ambiente com seis máquinas virtuais.

O número de erros ocorridos também é avaliado para os cinco ambientes testados, sendo que os dados obtidos podem ser visualizados na Figura 36.

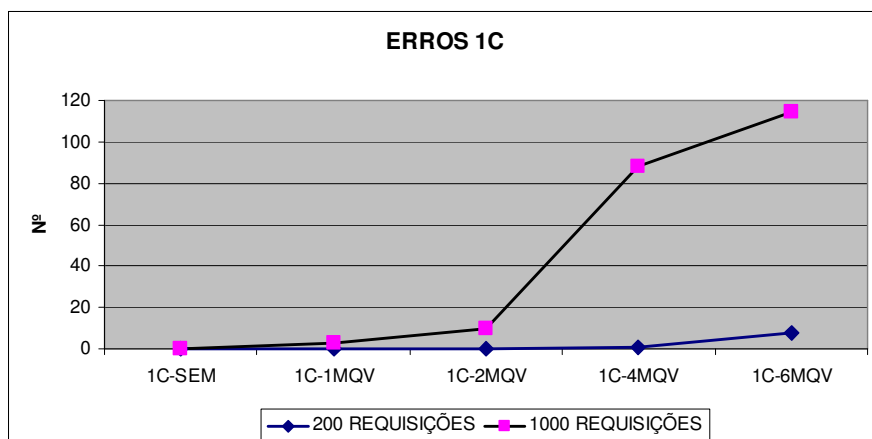


Figura 36: Autobench: número de erros para 1 cliente

Ao analisar o gráfico contido na Figura 36, pode-se afirmar que o número de erros é considerado nulo para os testes com 200 requisições (linha azul). Já nos testes com maior saturação, como o de 1000 requisições, percebe-se que o número de erros segue equilibrado até o ambiente com duas máquinas virtuais (1C-2MQV), onde o número de erros é igual a 10 (1% do total de requisições), e cresce drasticamente no ambiente com quatro máquinas virtuais (1C-4MQV), onde o número de erros é igual a 88 (quase 8,8% do total de requisições), no entanto, este número de erros pode ser considerado baixo, pois é inferior a 10% das requisições totais.

Contudo, com uma análise inicial destes resultados, pode-se afirmar que, com pouca saturação, o desempenho do serviço de servidor *web* é aceitável para uso de até quatro máquinas virtuais. Já quando a saturação é maior o limite de uso, para o *hardware* específico testado, é de duas máquinas virtuais. Entretanto, uma afirmação mais concreta deste limite só pode ser dada após a análise de resultados dos testes que provocam uma maior exaustão do servidor, e estes resultados virão dos testes dos cenários com dois e três clientes apresentados nas seções 5.4.2. e 5.4.3..

5.4.2 Cenário com dois clientes

O objetivo de se realizar testes em um cenário com dois clientes é duplicar o número de testes sobre os servidores, propiciando assim, uma maior saturação dos mesmos. Os testes executados aqui são os mesmos que os realizados para o cenário com um cliente, ou seja, as mesmas aplicações são utilizadas e as mesmas métricas são avaliadas.

Um exemplo de que a saturação do servidor ocorre com a execução de dois clientes, pode ser visualizado na Figura 37, que apresenta a captura de tela realizada sobre o servidor no momento em que um dos testes da ferramenta *Autobench* era executada para o cenário de dois clientes e seis máquinas virtuais.



Figura 37: Saturação do servidor para 2 clientes

Ao observar a Figura 37 pode-se perceber que o uso de processador, apesar de oscilar, é saturado em vários instantes. Percebe-se também, que o uso de memória do servidor permanece inalterado, uma vez que o ambiente de

virtualização retira do sistema hospedeiro os recursos de memória utilizados pelas máquinas virtuais.

Confirmado o uso exaustivo dos recursos do sistema principal, é necessário verificar os níveis de desempenho dos diferentes ambientes para a aplicação dos testes replicados em dois clientes. Os resultados obtidos serão apresentados da mesma forma que na seção 5.4.1, com avaliações individuais sobre cada medida de desempenho de cada ferramenta. Primeiramente são apresentados os resultados da ferramenta *ApacheBench*, seguidos dos resultados da ferramenta *Autobench*.

O primeiro item avaliado para os testes com dois clientes é o tempo total utilizado para realizar os testes, os resultados deste podem ser visualizados na Figura 38.

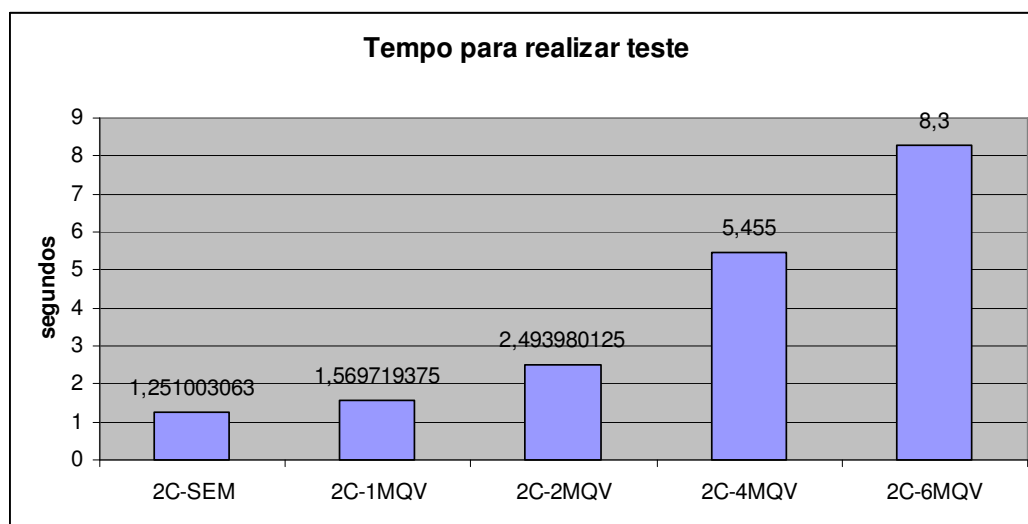


Figura 38: *ApacheBench*: Tempo total dos testes para 2 clientes

A análise da Figura 38 permite constatar que, diferentemente do teste com um cliente, a queda de desempenho ocorre gradativamente até o último ambiente testado. Os dados obtidos indicam que o tempo total dos testes para o ambiente sem virtualização é de 1,25 segundos, já a mesma medida para uma máquina virtual é de 1,57 segundos, um valor levemente superior. Porém, o acréscimo observado para os três demais ambientes é brusco enquanto, o teste para duas máquinas virtuais é realizado em 2,49 segundos, para quatro máquinas o mesmo

é feito em 5,45 segundos e para seis máquinas o valor é de 8,3 segundos. Apesar do aumento acentuado, ele é completamente compreensível, pois os recursos do sistema são divididos igualmente entre as máquinas virtuais e conforme o número de máquinas é duplicado o desempenho tende a cair pela metade.

Outro item também avaliado para testes com dois clientes é o número de requisições por segundo suportadas, os resultados da ferramenta *ApacheBench* podem ser visualizados na Figura 39.

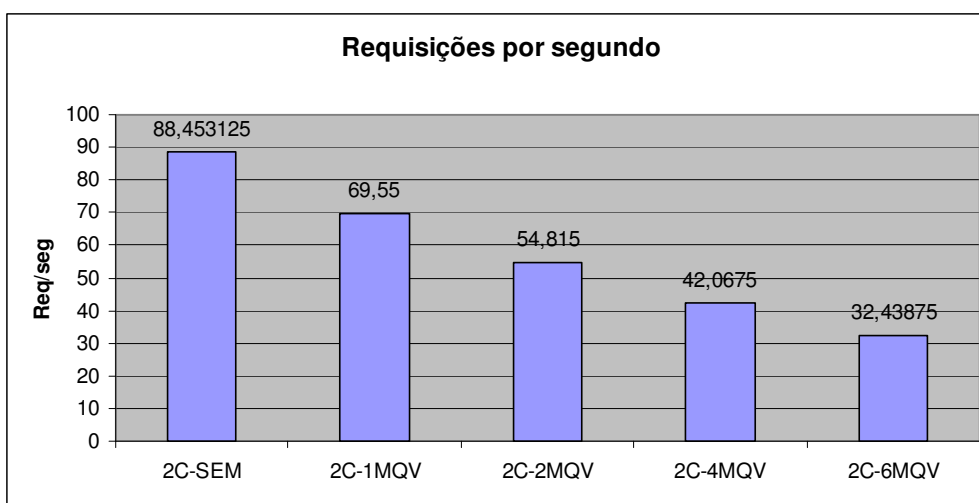


Figura 39: *ApacheBench*: Requisições por segundo para 2 clientes

Os resultados apresentados pela Figura 39 indicam que existe uma grande diferença de desempenho de um servidor *web* instalado em uma máquina nativa (2C-SEM) para uma máquina virtual (2C-1MQV), os valores indicam que a queda entre um ambiente e outro é de 21%, e este mesmo percentual permanece a cada novo ambiente. No entanto, o ponto positivo observado é que a diferença de desempenho dos testes em um ambiente com uma máquina virtual para o ambiente com seis máquinas virtuais é de 53,37% (diferença de 37,12 segundos). Apesar de esta ser uma grande queda, ela não é tão problemática, uma vez que, o valor esperado seria de 83% (diferença de aproximadamente 58 segundos). Esta estimativa se faz, pois teoricamente, em um ambiente com um número de máquinas virtuais multiplicado por seis, o desempenho deveria ser dividido por seis. Portanto, considera-se o ambiente com boa capacidade de equilibrar os recursos por entre as máquinas virtuais.

O ultimo teste realizado pela ferramenta *ApacheBench* para este cenário é o teste referente ao consumo de banda utilizado pelos servidores testados, e o resultado deste teste é apresentado pela Figura 40.

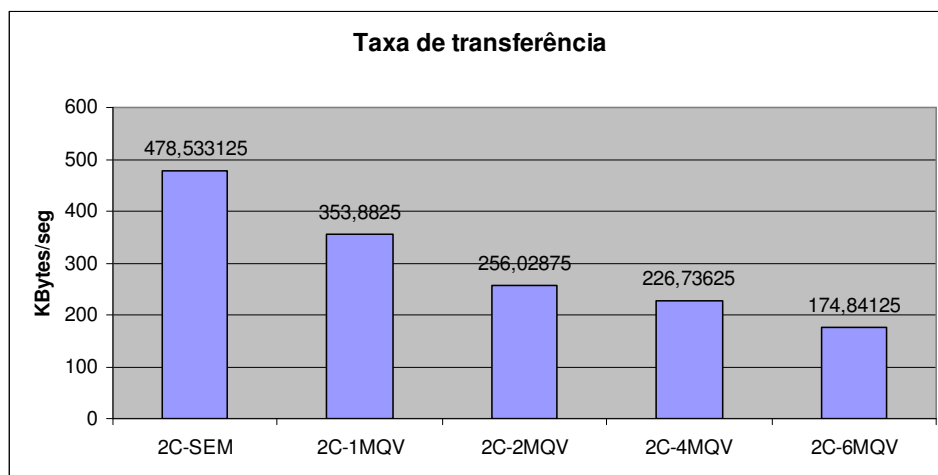


Figura 40: *ApacheBench*: Consumo de banda para 2 clientes.

Os resultados apresentados pela Figura 40 seguem um comportamento semelhante àqueles apresentados pela Figura 39, onde a queda de desempenho de um ambiente nativo para um ambiente com uma máquina virtual é de 26% (diferença de 124,65 Kbytes/seg), sendo que este valor de baixa é confirmado também, para a comparação entre os ambientes com uma e duas máquinas virtuais. No entanto, o teste no ambiente com quatro máquinas virtuais apresentou um resultado apenas 11% inferior ao teste com duas máquinas, já a diferença para seis máquinas virtuais se manteve em 22%. Portanto, a análise deste resultado indica mais uma vez a boa capacidade do ambiente em administrar os recursos entre os servidores, pois, novamente percebe-se que a queda (51%) do ambiente com uma máquina para o ambiente com seis máquinas foi inferior à estimada (83%).

Analizados os resultados apresentados pela ferramenta *ApacheBench* é necessário avaliar as informações indicadas pela ferramenta *Autobench*, pois, como dito no inicio desta seção, esta é a que tem a maior capacidade de saturar os ambientes de teste. O primeiro resultado apresentado por esta ferramenta para este cenário é a taxa de requisições suportadas, mensurada em requisições por

segundo. Os valores indicados para este teste podem ser visualizados na Figura 41.

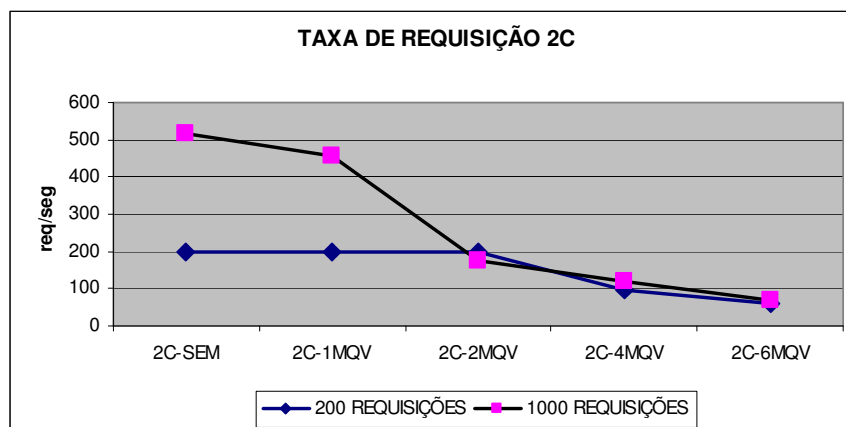


Figura 41: Autobench: Taxa de requisição para 2 clientes

Observando os dados da Figura 41, é possível perceber que o uso de dois clientes nos testes altera o comportamento das curvas de desempenho em relação ao cenário com um cliente. Por exemplo, o valor indicado para o ambiente sem virtualização que era de 1000 requisições por segundo (no teste com 1000 requisições), passou a ser de 500 requisições por segundo neste teste para dois clientes. No entanto a maior diferença na inserção de dois clientes observada é que na curva 200 requisições (linha azul), pode-se observar uma grande queda de desempenho a partir do ambiente com duas máquinas virtuais. Outro aspecto observado, é que a curva de 1000 requisições (linha preta), mantém uma queda superior àquela apresentada para os testes com um cliente, onde os resultados obtidos nos testes com duas máquinas virtuais já se equiparam aos resultados do mesmo ambiente para 200 requisições.

O próximo teste avaliado é o teste para o número de conexões suportadas em um segundo, o resultado deste teste é apresentado pela Figura 42.

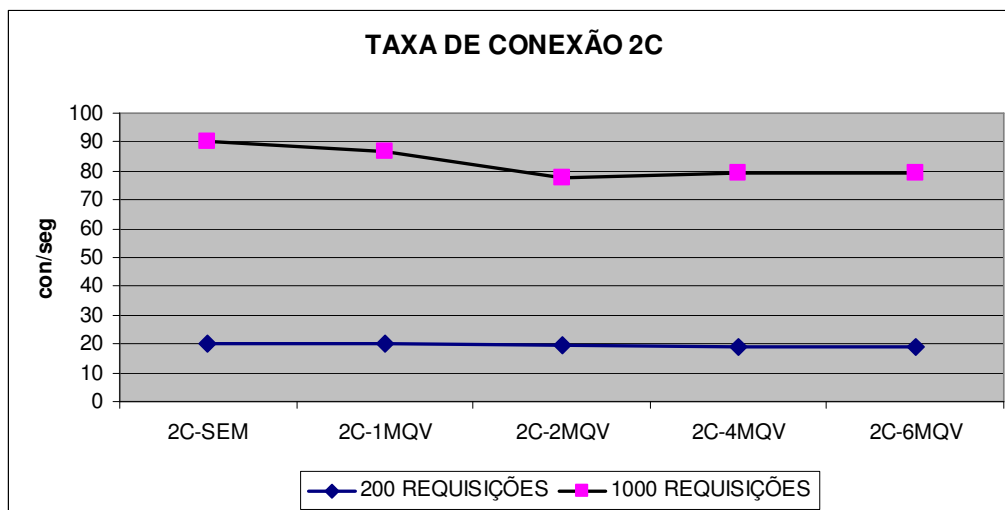


Figura 42: *Autobench*: Taxa de conexão para 2 clientes

Com os resultados apresentados na Figura 42 é possível perceber que as curvas de desempenho do teste para dois clientes mantêm o mesmo comportamento das curvas para o teste com um cliente. Portanto, estes resultados não se relacionam diretamente com a inserção ou não do ambiente de virtualização, mas sim, com o desempenho específico do recurso físico de rede.

O próximo item avaliado da ferramenta *Autobench* são os resultados que apresentam as informações sobre a medida da taxa de resposta. O gráfico com estes valores pode ser visualizado na Figura 43.

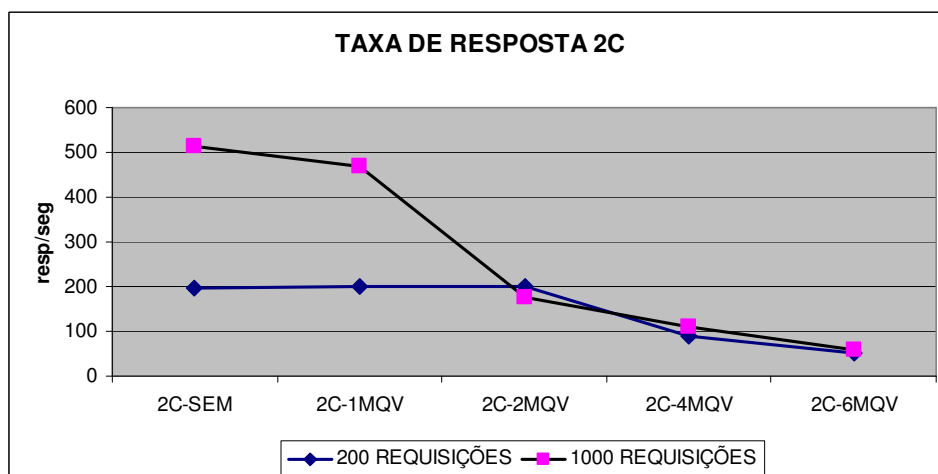


Figura 43: *Autobench*: Taxa de resposta para 2 clientes

Ao analisar os resultados apresentados pela Figura 43, é possível perceber que o comportamento das curvas de desempenho se equivale ao teste que avalia a taxa de requisição, onde a curva de 1000 requisições (linha preta) se equivale à curva de 200 requisições (linha azul) no ambiente com duas máquinas virtuais (2C-2MQV). Portanto, para testes com dois clientes e ambientes com mais de duas máquinas virtuais, o máximo que cada servidor consegue suportar é o resultado apresentado para os testes com 200 requisições, por exemplo, para duas máquinas virtuais o número de respostas por segundo suportadas é de em média 198,9 respostas por segundo. Mais uma vez é válido frisar que os resultados obtidos para 200 requisições e para 1000 requisições são considerados equivalentes em ambientes com duas ou mais máquinas virtuais, pois o desvio padrão obtido para estes testes é de 10%.

A próxima métrica avaliada para o cenário de testes com dois clientes é o tempo de resposta, sendo que os resultados obtidos podem ser visualizados no gráfico da Figura 44.

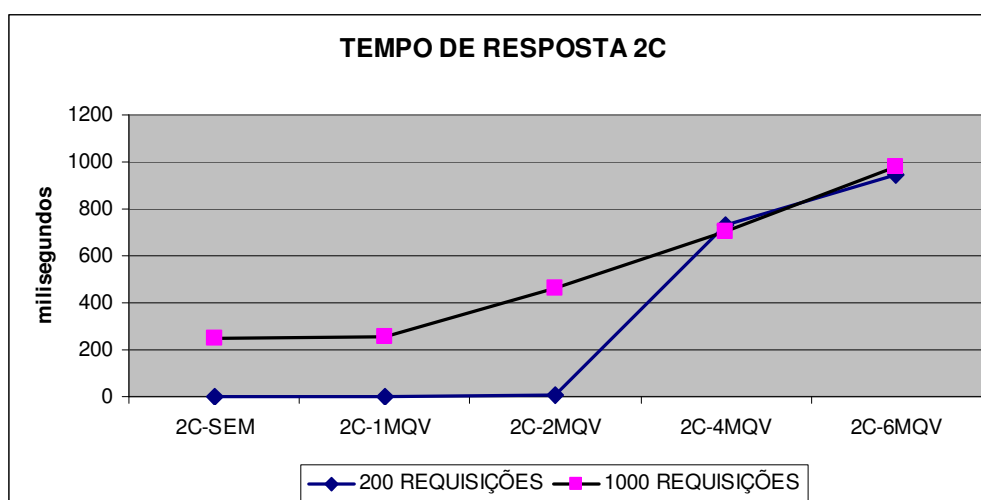


Figura 44: Autobench: Tempo de resposta para 2 clientes

Ao analisar o comportamento das curvas de desempenho apresentadas pela Figura 44, pode-se perceber que o tempo de resposta para 200 requisições (linha azul) se mantém bastante baixo até o ambiente com duas máquinas virtuais, e no ambiente com quatro máquinas virtuais o tempo para executar o mesmo teste

é de 793,3 milissegundos (ms), aumento este que se mantém para o ambiente com seis máquinas virtuais, onde o teste é realizado em 942,72 ms.

No entanto, a curva que demonstra o desempenho para 1000 requisições se mantém equilibrada apenas até o ambiente com uma máquina virtual, e a partir do ambiente com duas máquinas virtuais o tempo para realizar os testes cresce linearmente. Outro aspecto verificado é que no ambiente com quatro máquinas virtuais, as duas curvas obtiveram resultados semelhantes, o que indica que o número de requisições executadas não interfere diretamente no tempo total dos testes nestes ambientes (quatro e seis máquinas virtuais), ou seja, neste teste o uso de 1000 requisições não satura o servidor, que mantém o mesmo desempenho que o obtido para 200 requisições.

A ferramenta *Autobench* também apresenta resultados referentes à medida de desempenho que avalia o consumo de banda, e os mesmos podem ser visualizados no gráfico da Figura 45.

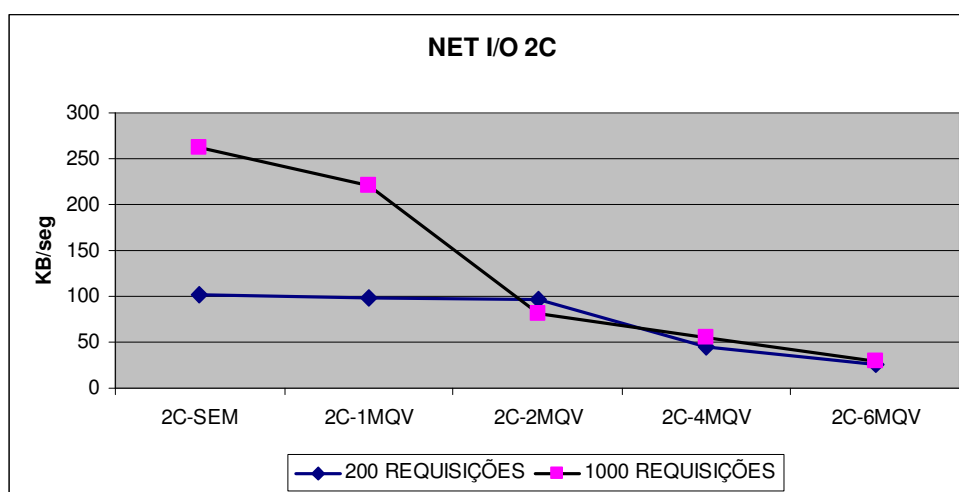


Figura 45: Autobench: Consumo de banda para 2 clientes

Ao analisar os resultados da Figura 45 é possível identificar que o comportamento das curvas de desempenho é semelhante àquele apresentado pelo gráfico que avaliara a taxa de resposta (Figura 43). Neste teste o desempenho da curva de 200 requisições (linha azul) se mantém equilibrado, retornando em média 100 Kbytes por segundo para os ambientes sem virtualização (2C-SEM), com uma máquina virtual (2C-1MQV) e com duas

máquinas virtuais (2C-2MQV). Nos ambientes com quatro máquinas virtuais (2C-4MQV) e seis máquinas virtuais (2C-6MQV), a queda é abrupta, neste último atingindo cerca de 75% em relação ao teste com uma máquina virtual. No entanto, para 1000 requisições o desempenho dos testes se mantém em queda de um ambiente pro outro e no ambiente com duas máquinas virtuais o mesmo se equipara com os resultados obtidos para 200 requisições, o que indica que a partir deste ambiente, tem-se o limite de 200 requisições suportadas.

O último item avaliado para o cenário de dois clientes é número de erros ocorridos durante os testes, sendo que os resultados obtidos para esta métrica podem ser visualizados na Figura 46.

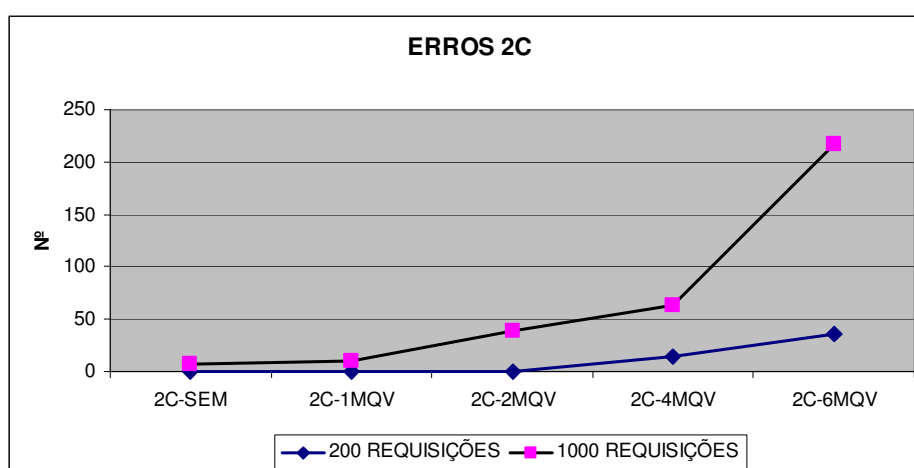


Figura 46: Autobench: número de erros para 2 clientes

Ao comparar os dados apresentados pela Figura 46 com aqueles indicados para o mesmo teste com um cliente (Figura 36), pode-se perceber que o número de erros ocorridos nos ambientes é superior no teste com 2 clientes, enquanto o teste com 1000 requisições (linha preta) para o ambiente com duas máquinas virtuais (2C-2MQV) apresentava um número de erros igual a 10, agora o mesmo ambiente com testes com dois clientes apresenta um número de erros igual a 38, portanto a inserção de novos clientes, por si só, faz aumentar o número de erros ocorridos. Percebe-se que, neste cenário o número de erros no ambiente com seis máquinas virtuais (2C-6MQV) e 200 requisições (linha azul) é de 35,87 ou 17,93% do total de requisições. Portanto, ao se falar em número de erros, pode-se considerar que o uso de até quatro máquinas virtuais é aceitável, pois apresentam

números baixos de erros tanto para 200 requisições, quanto para 1000 requisições, que apresentam respectivamente, 14 (7% do total de requisições) e 64 (0,64% do total de requisições).

Contudo, através uma análise secundária sobre os resultados obtidos no cenário com dois clientes, pode-se afirmar que o uso de duas máquinas virtuais apresenta resultados aceitáveis em testes que exijam pouca exaustão dos recursos do servidor. Porém, conforme a exaustão é maior, o desempenho apresenta grande queda, o que indica que o uso de máquinas virtuais pode interferir sensivelmente na percepção do usuário. No entanto, uma análise sobre a situação crítica de uso do servidor também deve ser simulada, sendo que os resultados desta simulação virão dos testes realizados a partir de três máquinas clientes e estão descritos na seção 5.4.3.

5.4.3 Cenário com três clientes

A intenção de se utilizar um cenário com três clientes executando as aplicações de teste, é justamente simular uma situação de alta carga no servidor, onde os testes são simultaneamente triplicados, ou seja, o uso dos recursos do mesmo é saturado neste cenário.

A Figura 47 demonstra uma captura de tela realizada sobre o monitor do sistema operacional que abriga o ambiente de virtualização, durante a execução da ferramenta *Autobench* em um ambiente com seis máquinas virtuais.



Figura 47: Saturação do servidor para 3 clientes

Ao analisar a Figura 47, é possível perceber que o uso do processador se mantém quase sempre acima de 50%, chegando, em algumas situações, a 100%, o que indica a saturação do servidor, mais uma vez percebe-se que o uso de memória segue inalterado durante a execução dos testes, o que indica que as máquinas virtuais não podem acessar diretamente a memória remanescente do sistema, o que caracteriza o isolamento.

É importante frisar que a metodologia dos testes e explicação dos resultados desta seção, segue o mesmo padrão dos cenários anteriores, ou seja, as mesmas medidas de desempenho serão avaliadas neste cenário.

Assim como nas outras seções, os primeiros resultados avaliados são aqueles obtidos através da ferramenta *ApacheBench*, sendo que o primeiro deles é referente ao tempo total utilizado para realização dos testes. O gráfico com estas informações pode ser visualizado na Figura 48.

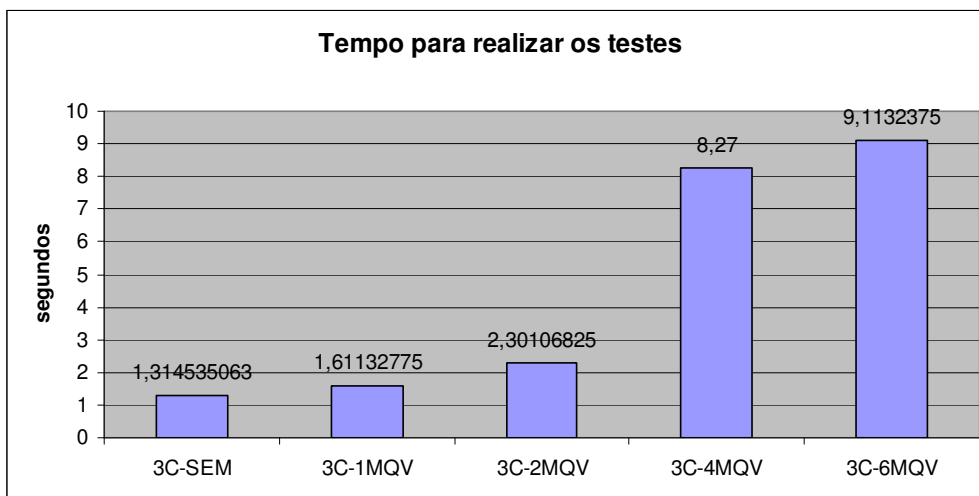


Figura 48: *ApacheBench*: Tempo para realizar testes para 3 clientes

Os dados apresentados pela Figura 48 indicam o tempo total utilizado para realizar os testes da ferramenta *ApacheBench* nos cinco diferentes ambientes, e com esses resultados é possível perceber que o cenário de testes com três clientes promove uma queda de desempenho em relação ao cenário com dois clientes. Percebe-se também que o desempenho mantém um decréscimo equilibrado entre os ambientes: sem virtualização, teste realizado em 1,31 segundos; com uma máquina virtual, realizado em 1,61 segundos; e com duas máquinas virtuais, realizado em 2,30 segundos. Porém, uma queda abrupta de desempenho é percebida no ambiente com quatro máquinas virtuais, onde o teste foi realizado em 8,27 segundos. No entanto o equilíbrio da queda se manteve no ambiente com seis máquinas virtuais, onde o teste foi realizado em 9,11 segundos, este pode ser devido à dificuldade de se manter o sincronismo nos instantes finais dos testes, pois, como os clientes utilizados possuíam configurações de *hardware* diferentes, eles terminavam as execuções em tempos distintos, e em instantes finais dos testes o sincronismo não era mais percebido.

O segundo item avaliado pela ferramenta *ApacheBench* é o número de requisições por segundo que são suportadas pelo servidor *web*, e os resultados obtidos para o cenário de três clientes podem ser visualizados na Figura 49.

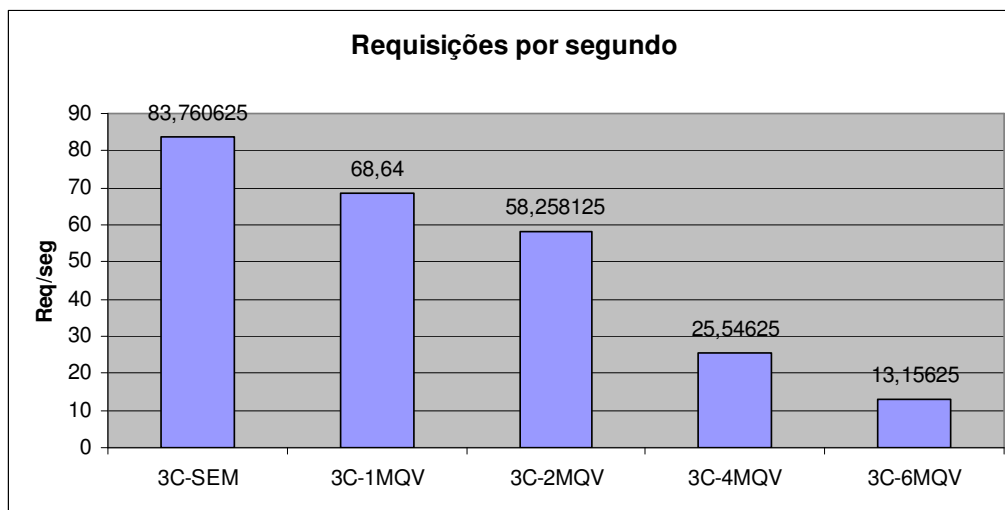


Figura 49: ApacheBench: Requisições por segundo para 3 clientes

Analisando os resultados apresentados pela Figura 49 é possível perceber que o número de requisições simultâneas suportadas por cada ambiente cai equilibradamente até os testes com duas máquinas virtuais (3C-2MQV), onde o resultado obtido foi de 58,26 requisições por segundo, valor este que é 15% inferior ao obtido para uma máquina virtual (3C-1MQV) que obteve um resultado igual 68,64 segundos. No entanto, os testes realizados para quatro e seis máquinas virtuais indicam uma queda de desempenho mais acentuada: a diferença entre o ambiente com duas máquinas para o ambiente com quatro é de 56%, e a diferença entre o ambiente com quatro máquinas para o ambiente com seis é de 48%. Estes resultados podem indicar o limite de uso de apenas duas máquinas virtuais com desempenho aceitável.

A última métrica mensurada pela ferramenta é o consumo de banda. Os resultados obtidos para esta medida de desempenho podem ser visualizados na Figura 50.

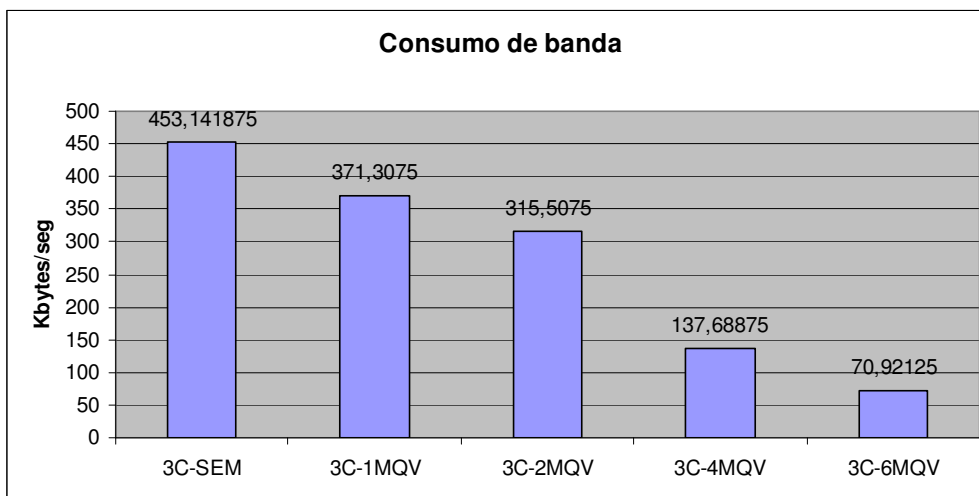


Figura 50: *ApacheBench*: Consumo de banda para 3 clientes

Ao observar o gráfico da Figura 50 é possível perceber que o comportamento do consumo de banda nos cinco ambientes é semelhante ao comportamento obtido para a medição das requisições por segundo, onde o desempenho mantém uma queda equilibrada até o ambiente com duas máquinas virtuais, e cai mais de 50% em testes realizados sobre o ambiente com quatro máquinas virtuais. A mesma queda se mantém quando se compara os ambientes com quatro e seis máquinas virtuais.

Portanto, com uma análise preliminar dos resultados, obtidos com testes da ferramenta *ApacheBench* sobre o cenário com três clientes, é possível afirmar que para este *hardware* específico pode se utilizar a virtualização para consolidar dois servidores, pois o ambiente de virtualização é capaz de equilibrar a disponibilidade dos recursos para os servidores virtualizados. Porém, ainda é necessário verificar qual o comportamento dos testes realizados com a ferramenta *Autobench*, pois com estes foi possível simular a saturação do servidor. O primeiro resultado avaliado é a taxa de requisição; os dados sobre esta métrica podem ser visualizados no gráfico da Figura 51.

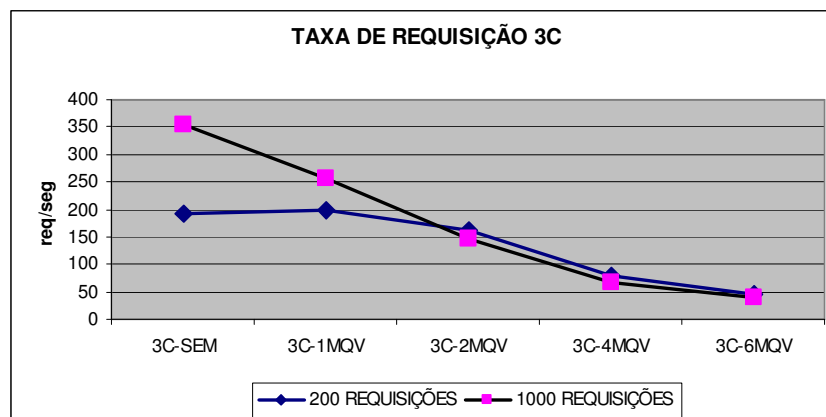


Figura 51: *Autobench*: Taxa de requisição para 3 clientes

Ao analisar o resultado mostrado na Figura 51 é possível perceber que a saturação do servidor interfere diretamente na taxa de requisição. A queda de desempenho para os testes com 200 requisições se inicia no ambiente com uma máquina virtual, e decai 25% para o ambiente com duas máquinas virtuais, exatamente onde a curva de desempenho dos testes de 1000 requisições se equivale à curva de 200 requisições, o que indica que, a partir de duas máquinas virtuais, o máximo de requisições por segundo que cada servidor consegue suportar é aquele indicado para os testes com 200 requisições, ou seja, para o ambiente com duas máquinas virtuais o valor suportado é de 160,65, para quatro máquinas o valor é de 78,2 e para seis máquinas o valor é de 44,9 requisições por segundo.

O próximo item a ser avaliado para os testes da ferramenta *Autobench* é a taxa de conexão, sendo que os resultados obtidos podem ser visualizados na Figura 52.

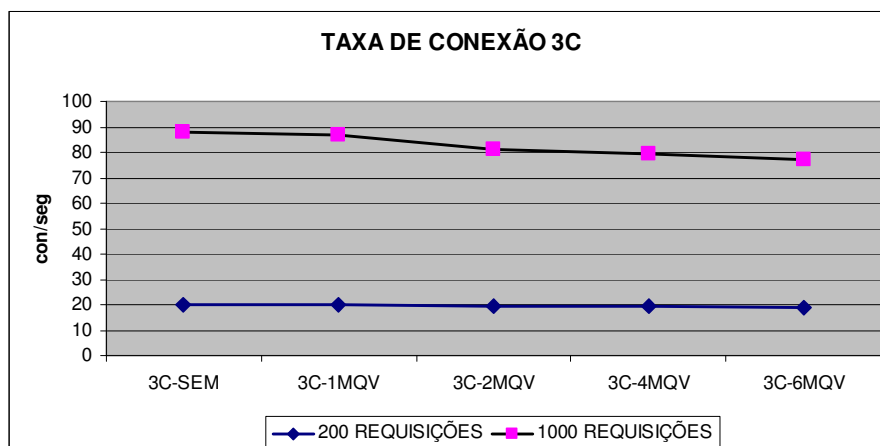


Figura 52: Autobench: Taxa de conexão para 3 clientes

Ao analisar o gráfico incluso da Figura 52 é possível perceber que o resultado obtido para os testes com três clientes é idêntico àqueles obtidos para os cenários com um e dois clientes. Portanto, a saturação do servidor e o uso de virtualização não interferem significativamente nos dados para a taxa de conexão.

O próximo resultado a ser avaliado para este cenário é a taxa de resposta, sendo que as medidas obtidas podem ser visualizadas na Figura 53.

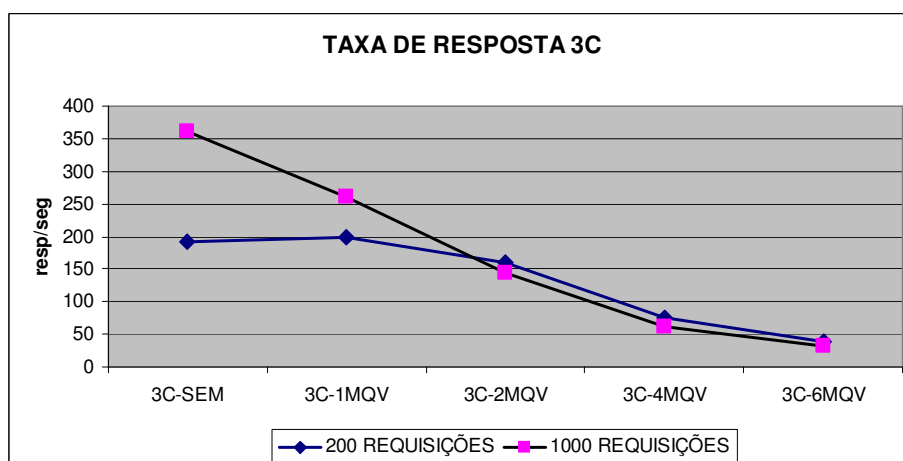


Figura 53: Autobench: Taxa de resposta para 3 clientes

Ao observar os dados presentes na Figura 53 pode-se perceber que o comportamento das curvas de desempenho, para a taxa de resposta, seguem o mesmo padrão daquelas que mensuravam a taxa de requisição, ou seja, para 200 requisições o desempenho se mantém com pouca queda até o ambiente de duas

máquinas virtuais, e a partir deste ambiente uma queda linear pode ser verificada. Percebe-se também que a equivalência de desempenho entre as curvas de 1000 e 200 requisições novamente ocorre no ambiente com dois servidores virtuais. Portanto, a partir de duas máquinas virtuais, o melhor resultado apresentado pelos servidores, quanto à taxa de resposta, é aquele obtido nos testes com 200 requisições.

O próximo gráfico avaliado é aquele que indica valores referentes ao tempo de resposta, e pode ser visualizado na Figura 54.

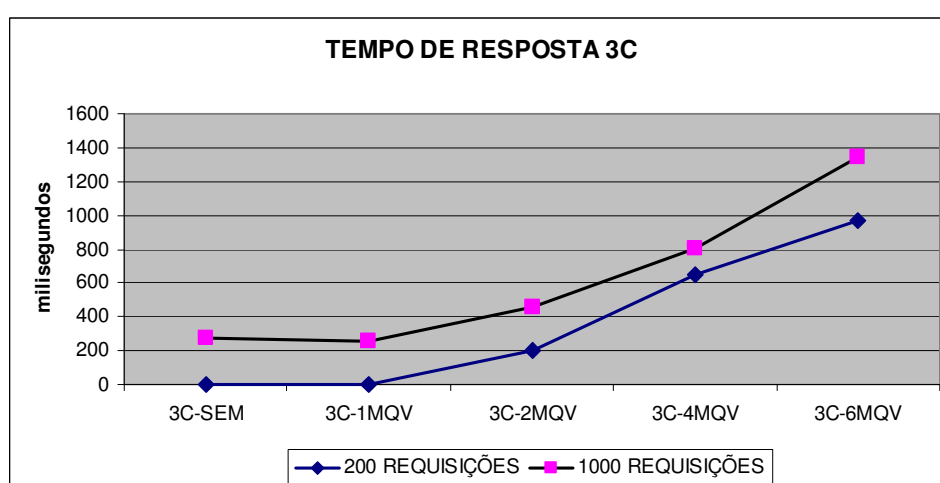


Figura 54: Autobench: Tempo de resposta para 3 clientes

Ao analisar os resultados referentes ao tempo de resposta, demonstrados pela Figura 54, pode-se perceber que os valores obtidos para ambientes sem virtualização e com uma máquina virtual são estatisticamente equivalentes, ou seja, o desempenho de uma máquina virtual é semelhante ao desempenho de uma máquina nativa. Percebe-se também que, nos testes com 1000 requisições, ao duplicar o número de máquinas virtuais de uma para duas e de duas para quatro, o tempo de resposta também é duplicado. Outra característica importante percebida é que as duas curvas não se encontram, o que indica que mesmo com saturação do servidor, o limite de desempenho ainda é aceitável para um número de requisições maior que 200, resultado este diferente daquele obtido em outras métricas.

A próxima métrica a ser avaliada para os resultados da ferramenta *Autobench* é o consumo de banda, sendo que o mesmo também foi mensurado para testes com 200 e 1000 requisições. Os dados obtidos com os testes podem ser visualizados na Figura 55.

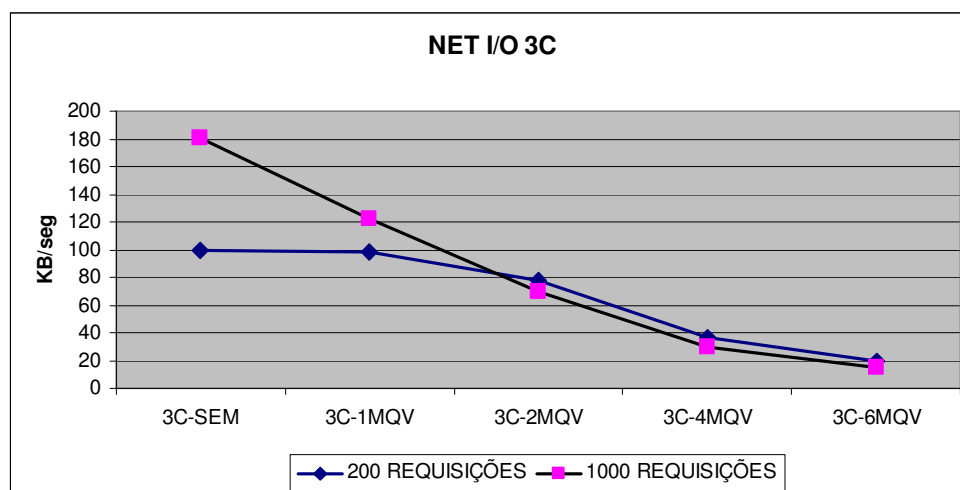


Figura 55: Autobench: Consumo de banda para 3 clientes

Ao analisar as curvas de desempenho apresentadas pela Figura 55, é possível perceber que elas seguem o mesmo comportamento já observado nos testes sobre a taxa de resposta e a taxa de requisição, onde, para testes de 1000 requisições, o ambiente sem virtualização possui um desempenho superior ao ambiente com uma máquina virtual, sendo que a diferença observada é de 57,8 Kbytes por segundo (uma queda de 32%). Percebe-se também que esta degradação se mantém linearmente para os três demais ambientes. No entanto, os testes com 200 requisições indicam um equilíbrio de desempenho para os ambientes 3C-SEM e 3C-1MQV, e é a partir deste último que se inicia a diminuição do valor do consumo de banda, que decai linearmente para os ambientes 3C-2MQV, 3C-4MQV e 3C-6MQV, mantendo o mesmo nível de desempenho obtido para os testes com 1000 requisições.

Por fim, o último item a ser avaliado é o número de erros ocorridos durante os testes, e o gráfico com estes resultados pode ser visualizado na Figura 56.

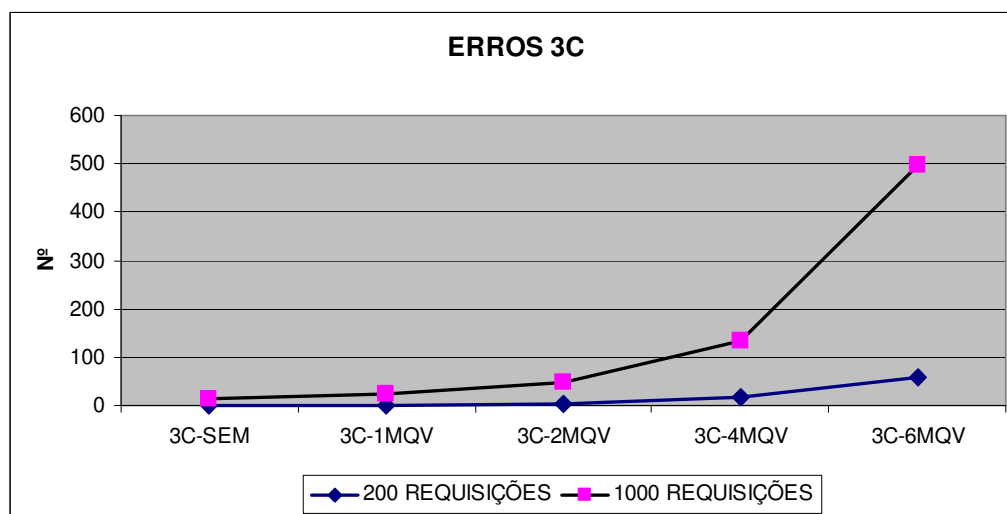


Figura 56: Autobench: Número de erros para 3 clientes

Com a análise dos dados apresentados pela Figura 56 é possível perceber que, apesar de as curvas terem comportamento semelhante aos outros cenários, o número de erros ocorridos para o cenário com três clientes é superior àquele apresentado para o cenário com dois clientes. Percebe-se que, para testes com 1000 requisições, os valores obtidos para ambientes com até duas máquinas virtuais é inferior a 10%, porém o valor obtido para o ambiente com quatro máquinas é igual a 132,52 (13,25% das requisições totais), sendo que, este número é quadruplicado para o ambiente com seis máquinas virtuais, onde atinge 498 erros (quase 50% do total de requisições). No entanto, o comportamento da curva onde o teste foi realizado para 200 requisições é diferente daquele apresentado para 1000 requisições, pois, no ambiente com duas máquinas virtuais, tem-se um número de erros igual a 2,85 (1,4% do total de requisições), com quatro máquinas virtuais este valor é igual a 17,5 (8,75% do total de requisições), e com seis máquinas este valor é igual a 58,54 (29,27% do total).

Ao analisar estes valores, pode-se afirmar que, com pouca exaustão do servidor o percentual de erros em relação ao total é inferior àquele apresentado para testes com muita exaustão, porém ainda é cabível recomendar o uso de apenas duas máquinas virtuais, pois o número de erros se mantém aceitável tanto para 200 quanto para 1000 requisições.

Obtidos os resultados dos testes para os três cenários distintos, é necessário realizar uma análise final, sobre o uso de virtualização para consolidação de servidores, sendo que esta é realizada na seção 5.5.

5.5 Análise de custo-benefício da virtualização

Depois de obtidos os resultados dos testes com diferentes cenários e diferentes ambientes, é necessário avaliar os mesmos para que se possa realizar a análise do custo-benefício da virtualização para consolidação de servidores, sendo que esta será realizada nesta seção. Para tanto, primeiramente deve-se analisar os resultados obtidos da comparação entre o sistema nativo e o sistema com o ambiente de virtualização, depois se deve avaliar os dados obtidos da comparação do desempenho dos recursos de rede nos diferentes cenários, para então finalmente avaliar as informações obtidas quanto ao desempenho do servidor *web*.

Ao analisar o comparativo realizado sobre a inserção da ferramenta XEN, pode-se perceber que esta instalação por si só promove alguma queda de desempenho em alguns recursos, principalmente no caso de criação de processos e na comunicação entre processos. No entanto, os resultados obtidos para os demais recursos indicam um equilíbrio de desempenho, por exemplo, a avaliação do sistema de arquivos e execução de processos. Portanto, em análise inicial, o uso de virtualização é bem vindo em aplicações que necessitem utilizar os recursos do sistema de arquivos. Porém, a análise complexa de um destes serviços (servidor *web*) ainda deve ser realizada, para que se possa afirmar qual o limite do uso de virtualização para que se tenha um melhor custo-benefício.

O próximo item a ser avaliado, é o desempenho dos recursos de rede, sendo que os resultados foram obtidos dos testes nos cinco ambientes, e com três

diferentes níveis de saturação. Admite-se como nível baixo de saturação o uso de um cliente sobre cada servidor, nível médio de saturação o uso de dois clientes sobre cada servidor, e alto de saturação o uso de três clientes.

Com os resultados obtidos para testes sobre os recursos de rede, foi possível perceber que, em testes com baixa saturação, o ambiente de virtualização distribui de forma equilibrada os recursos para todos os servidores. Porém, nos testes com média saturação, o uso de virtualização somente é equilibrado até quatro máquinas virtuais, e, em testes com alta saturação o ambiente de virtualização sofre grande queda de desempenho, onde os resultados aceitáveis para virtualização são aqueles que indicam em no máximo o uso de duas máquinas virtuais. Portanto em uso médio do servidor, pode-se utilizar, para este *hardware*, um nível de quatro servidores virtualizados, pois desta forma não se tem grande perda de desempenho.

Entretanto, ainda é necessário avaliar o desempenho do servidor com relação ao comportamento do uso do serviço de servidor *web*, e da mesma forma como os recursos de rede, esta avaliação é feita com base nos três níveis de saturação do servidor.

Percebe-se que, na avaliação do serviço, o desempenho obtido segue um comportamento diferenciado para os diferentes níveis de saturação. Para um nível baixo de saturação pode-se perceber que o uso de até quatro máquinas virtuais pode ser realizado com baixas perdas de desempenho. No entanto, em níveis de média e alta saturação o limite proposto é de duas máquinas virtuais, pois mesmo com grandes quedas de desempenho para 1000 requisições simultâneas, em 200 requisições esta não é tão flagrante. Sabe-se também, que em uso corriqueiro, os servidores não trabalham em alta saturação, estas situações ocorrem esporadicamente e, portanto admite-se o uso de duas máquinas virtuais.

Contudo, com a análise sobre os diferentes testes e resultados é possível afirmar que a faixa de uso de dois a quatro servidores *web* virtualizados é recomendado, pois assim não haverá grandes perdas de desempenho em situações extremas de uso e pode se manter um baixo nível de ociosidade. Sabe-se também que o desempenho do *hardware* é variável limitante para esta

definição, uma vez que, em *hardware* com capacidade maior de recursos, o número de máquinas virtuais utilizadas poderia ser maior. Porém, é necessário avaliar qual o custo-benefício da utilização de duas máquinas virtuais em um servidor.

O primeiro ponto positivo analisado para a formulação do custo-benefício é a economia que pode ser realizada, estima-se que o custo anual de um servidor está por volta de US\$ 10.000,00 [CROSBY, 2006], e com os testes realizados foi possível afirmar que o uso de duas a quatro máquinas virtuais pode ser feito sem grandes perdas de desempenho. Portanto, em uma situação onde fossem utilizados dez servidores físicos, estes poderiam ser substituídos por cinco máquinas físicas com dois servidores virtuais cada e a economia obtida seria de US\$ 50.000,00 anuais ou então, na mesma situação de dez máquinas físicas pode-se reduzir este número para três, duas com três máquinas virtuais e uma com quatro, o que representaria uma economia de US\$ 70.000,00 anuais.

O segundo item avaliado está relacionado ao uso eficaz dos recursos tecnológicos. Sabe-se que em um servidor trabalha normalmente com 10% dos seus recursos utilizados, tendo 90% deles em ociosidade [VOGELS, 2008]. Com a análise realizada foi possível perceber que, ao se utilizar máquinas virtuais, este uso de recursos sobe a cada nova inserção das máquinas, e varia de acordo com o serviço utilizado por cada uma. Entretanto, como o nível de desempenho aceitável foi obtido em um cenário para média saturação, o uso indicado seria de duas máquinas virtuais. Porém, se o cenário nos quais os servidores a serem virtualizados estão inseridos não sofre grande saturação, pode-se até limitar o uso de quatro máquinas virtuais para que se tenha melhor aproveitamento dos recursos, sem grandes perdas de desempenho.

Contudo, a afirmação de que duas máquinas virtuais provêm um bom custo-benefício é aceitável, uma vez que, se percebe uma boa economia com relação aos custos de *hardware*, e mantém-se um bom desempenho em situações com pouca saturação, e um desempenho sem grandes perdas em cenários esporádicos, como é o caso da saturação elevada, no entanto, sabe-se que estes serviços que demandam alta carga de recursos, são exceções ao uso normal dos

servidores. Além de que, ao se utilizar duas a quatro máquinas virtuais pode-se diminuir a ociosidade do servidor que as mantém. Entretanto, é válido frisar que esta definição é dada para o *hardware* específico utilizado neste trabalho, para se ter o número real de máquinas virtuais aceitáveis em um *hardware* com diferentes recursos, é necessário realizar um comparativo de desempenho entre o mesmo e este que foi utilizado nestes testes. Possivelmente, em um *hardware* que possua o dobro de capacidade de recursos, o número de máquinas virtuais admitidas também pode ser duplicado.

6. CONCLUSÃO

Antes de apresentar as conclusões obtidas com este trabalho é necessário relembrar o objetivo principal da realização do mesmo. A principal motivação da execução deste estudo é a obtenção de uma análise de custo-benefício da virtualização para consolidação de servidores. A consolidação de servidores, que consiste em substituir diversos servidores físicos por uma única máquina que concentra todos os serviços, é uma tendência que tem sido cada vez mais utilizada no mercado. Entre os motivos por trás da consolidação, pode-se citar a simplificação da administração de *data centers* e principalmente, reduções de custos, incluindo gastos com hardware, espaço físico e energia. A virtualização é uma tecnologia chave para a consolidação de servidores, pois permite que os serviços sejam alocados sem máquinas virtuais, mantendo assim, o isolamento entre eles. Uma questão importante no uso da virtualização para consolidação de servidores é entender até que ponto existe benefício no uso da consolidação, isto é, quando é que as desvantagens dessa abordagem (como a perda de desempenho) superam as suas vantagens.

A virtualização pode ser aplicada em diferentes camadas de um sistema computacional: hardware, sistema operacional e aplicação. Percebeu-se que dentre as camadas de virtualização estudadas, as que podem ser utilizadas para a consolidação de servidores são as inseridas em níveis de sistema operacional e de *hardware*, sendo que a camada de *hardware* foi a utilizada durante o andamento deste trabalho.

Ao se comparar as ferramentas de virtualização, percebeu-se que o ambiente de virtualização Xen é o que melhor se relacionou com os objetivos de consolidação de servidores. Dentre os fatores relevantes para esta definição pode-se citar a capacidade de virtualização de *hardware* e sistema operacional distintos

daqueles onde o ambiente é inserido, a boa documentação e de não necessitar de pagamento de licença para seu uso.

Depois de realizados os testes em diferentes ambientes, e com os resultados obtidos, foi possível realizar uma análise completa do custo-benefício da virtualização para consolidação de servidores. Conclui-se que, para este *hardware* e serviço específicos, o uso de duas a quatro máquinas virtuais pode ser feito sem grandes perdas de desempenho, e ao utilizar este número uma boa redução de custos pode ser percebida.

Um possível trabalho a ser realizado seria um estudo de caso referente a virtualização para consolidação de servidores que executem serviços diferentes. Por exemplo, pode-se realizar uma simulação, onde, em um mesmo servidor se tivesse uma máquina virtual com servidor *web*, uma máquina virtual como servidor de arquivos e uma máquina virtual como servidor de *proxy*, e aí então analisar o desempenho obtido para testes sobre cada um destes servidores. Seria importante que o cenário de testes fosse inalterável, onde os clientes utilizados para os testes fossem idênticos, ou seja, que tivessem desempenho idêntico, porém, para isso seria necessário utilizar de um maior desembolso financeiro.

Outro trabalho que pode ser realizado na área de virtualização para consolidação de servidores, seria conduzir estudos sobre a viabilidade da virtualização em um cenário específico, usando serviços “reais” onde a mesma carga aplicada a eles, deveriam ser feita em um cenário sem virtualização. Ou seja, pode-se realizar um estudo de caso, onde se deve avaliar o desempenho de um serviço real que esteja executando normalmente em um ambiente de uso normal, como por exemplo, um servidor de uma empresa, e depois deve-se migrar este serviço para um ambiente virtualizado, para então avaliar os resultados obtidos com o uso normal neste cenário. Por fim, com os resultados obtidos pode-se ter dados concisos referentes ao desempenho nos dois ambientes.

7. ANEXOS

Neste capítulo estão descritas as etapas do processo de instalação e configuração da ferramenta XEN, as etapas de criação das máquinas virtuais, os arquivos de configuração utilizados para as mesmas e o plano de TCC utilizado para este trabalho.

7.1 Configurando ambiente de virtualização

Nesta seção está descrito as etapas seguidas para a instalação da ferramenta de virtualização XEN.

1)Buscando pacotes:

```
apt-get install iproute bridge-utils python-twisted gcc binutils make zlib1g-dev python-dev transfig bzip2 screen ssh debootstrap libcurl3-dev libncurses5-dev x-dev libsdl1.2-dev bin86 bcc python-pam patch latex-make qemu qemu-launcher qemu-utl graphviz graphviz-dev lvm2 libvncserver-dev libjpeg62-dev libjpeg62-gettext python-xml gawk tetex-extra tetex-base openssl libssl-dev mercurial
```

2)Baixando o XEN:

```
cd /usr/src
wget http://bits.xensource.com/oss-xen/release/3.2.0/xen-3.2.0.tar.gz
hg clone http://xenbits.xensource.com/linux-2.6.18-xen.hg
```

3)Compilando

```
tar zxvf xen-3.2.0.tar.gz
```

```
cd /usr/src/xen-3.2.0
make prep-kernels
make dist && make install
depmod 2.6.18.8-xen
mkinitramfs -o /boot/initrd.img-2.6.18.8-xen 2.6.18.8-xen
```

4) Atualizando GRUB

```
update-grub
```

5) Ativar os Serviços

```
update-rc.d xend defaults 20 21
update-rc.d xendomains defaults 21 20
```

6) Instalando LVM

```
# aptitude install lvm2
```

7.2 Criando as máquinas virtuais

1) Criar o disco virtual na LVM

```
apt-get install debootstrap
lvcreate -L 10G -n debian.disk servidores
```

2) Particionar o volume logico em duas partições

```
fdisk -b 512 -C 4096 -H 16 -S 32 /dev/servidores/debian.disk
partição 1: tipo swap (82) : 512M (ou coloque o tamanho que desejar)
partição 2: tipo linux (83) : o restante.
```

3) Para liberar acesso as partições é necessário instalar o aplicativo:

```
apt-get install multipath-tools
```

4)Disponibilizando acesso:

```
kpartx -a /dev/mapper/servidores-debian.disk
```

5)Formatando as partições:

```
mkswap /dev/mapper/servidores-debian.disk1
mkfs.ext3 /dev/mapper/servidores-debian.disk2
```

6)Montar a partição /dev/mapper/servidores-debian.disk2 para que se possa instalar o sistema Debian básico com o debootstrap:

```
mkdir /mnt/debian
mount /dev/mapper/VGxen-debian.disk2 /mnt/debian
```

7)Instalando o debian:

```
debootstrap etch /mnt/debian http://ftp.br.debian.org/debian
```

7.3 Arquivo de configuração das máquinas virtuais

```
import commands
kern_vers = commands.getoutput('uname -r')
builder = 'linux'
name ='debian10'
disk = [ 'phy:/dev/servidores/debian10.disk,xvda,w' ]
memory = 128
vif = [ 'bridge=xenbr0' ]
```



```
kernel = '/boot/vmlinuz-' + krn_vers
ramdisk = '/boot/initrd.img-' + krn_vers
root = '/dev/xvda2 ro'
on_poweroff = 'destroy'

on_reboot = 'restart'
on_crash = 'restart'
extra = 'xencons=xvc console=xvc0 video=tty'
vfb = [ 'type=vnc,vncdisplay=10,vnclisten=0.0.0.0,vncpasswd=ricardo' ]
```

7.4 Plano de TCC aprovado

Plano de Trabalho de Conclusão de Curso Análise de Custo-Benefício de Virtualização Para Consolidação de Servidores

UDESC - Centro de Ciências Tecnológicas
Departamento de Ciência da Computação
Bacharelado em Ciência da Computação - Integral
Turma 2008/1 - Joinville – Santa Catarina

Ricardo Jose Pfitscher – joinvas@hotmail.com

Orientador: Rafael Rodrigues Obelheiro – rro@joinville.udesc.br

Resumo – Este Trabalho de Conclusão de Curso tem por objetivo promover um estudo referente ao uso de múltiplos servidores em um mesmo *hardware* com auxílio de máquinas virtuais. A intenção é verificar os aspectos como a perda de desempenho, e definir o custo/benefício de uma implantação dessa natureza. Para desenvolvimento deste trabalho também será necessária a utilização de um ambiente de virtualização específico para serem realizados os testes. Este comparativo será realizado com auxílio de ferramentas que irão fornecer dados e informações referentes aos experimentos a ser proposto no decorrer do trabalho.

Palavras-chave: *Máquina Virtual, Fornecimento de Serviços, Múltiplos Servidores, redução de custos, Virtualização.*

1. Introdução e Justificativa

A utilização de estruturas informatizadas para armazenamento de informações teve sua explosão a partir da década de 90, e cada vez mais tem

aumentado o número de serviços necessários à sociedade, como por exemplo, o armazenamento de arquivos, o acesso à *web* e o compartilhamento de arquivos. E para suprir essa demanda de serviços, diversos artefatos informatizados são apresentados como soluções, entre estas podem-se encontrar diversos tipos de servidores [STALLINGS, 2005].

Servidores são máquinas físicas que tem como função principal prover serviços específicos a usuários, ou até centralizar serviços, a fim de atingir diversos fatores, como segurança e redução de custos. Um exemplo seria o uso de um servidor para prover um serviço de arquivos centralizado, mantendo em apenas uma máquina física dados que poderiam estar repetidos em diversas estações. Outro serviço que pode ser considerado é um *proxy* de acesso à *web*, que permite que um administrador tenha maior controle sobre os acessos externos realizados pelos usuários da sua rede interna.

Com o incremento na adoção desse tipo de solução, alguns desafios devem ser levados em consideração. Quanto maior a quantidade de serviços e servidores disponibilizados, maior é a complexidade de gerenciamento e maiores são as necessidades de utilização de plataformas e sistemas operacionais distintos [FIGUEIREDO, 2005].

Além disso, é comum que serviços distintos sejam implementados fisicamente em servidores separados. As razões pelas quais isso acontece incluem preocupações de segurança e confiabilidade (isolamento de serviços) e a diversidade de sistemas operacionais requeridos por diferentes serviços. Entretanto, esta forma tem se tornado cada vez mais cara. Estima-se que o custo de cada servidor no *data center* de uma empresa é de US\$ 10.000,00 por ano, incluindo gastos com administração, manutenção, energia elétrica, treinamentos, atualizações, entre outros custos. Uma possível solução para tais custos seria a utilização de virtualização a fim de diminuir a quantidade de *hardware* utilizada [CROSBY, 2006].

A virtualização consiste na inserção de uma camada de *software* entre o sistema operacional e o *hardware* que fornece ao usuário uma abstração dos recursos físicos da máquina, chamada de máquina virtual. Entre outras coisas,

isto permite que sejam utilizadas diversas imagens do sistema operacional (SO) em um único sistema, sendo que cada SO possui a impressão de interagir diretamente com o *hardware*. O mecanismo responsável por propiciar estes acessos é o ambiente de virtualização, que distribui os recursos necessários de acordo com a necessidade das máquinas virtuais. Entre os recursos gerenciados pelo ambiente de virtualização encontram-se a unidade de gerenciamento de memória, os dispositivos de entrada e saída, e os controladores de acesso direto à memória [CROSBY, 2006].

Na consolidação de servidores, cada servidor lógico pode executar em uma máquina virtual separada como se tivesse todos os recursos de *hardware* à sua disposição. Isto facilita a consolidação, uma vez que múltiplas unidades de máquinas virtuais podem ser executadas, disponibilizando múltiplos serviços, ficando o ambiente de virtualização responsável por gerenciar os recursos compartilhados pelas máquinas virtuais. A figura 1 apresenta um exemplo desta aplicação [FRASER, 2004]:

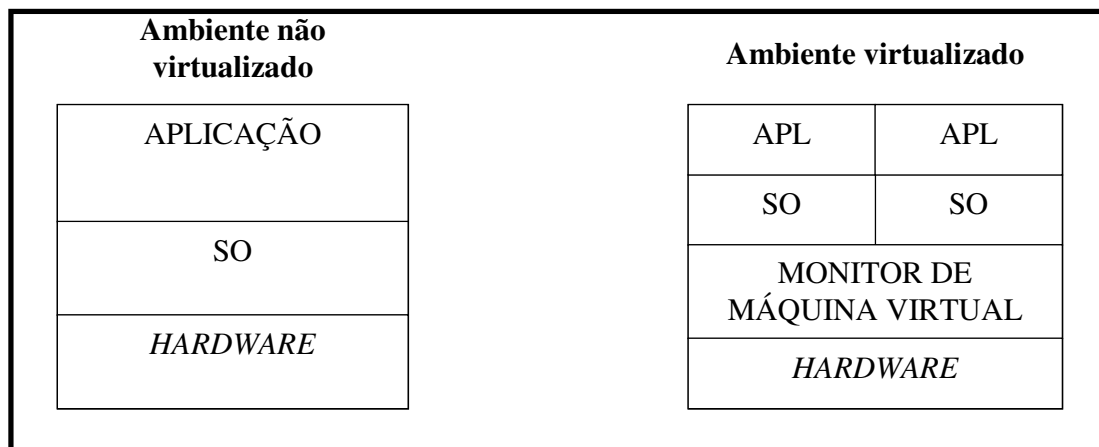


Figura 1 Comparativo de ambiente virtualizado com ambiente não virtualizado.

Basicamente, em um ambiente onde não é utilizada a virtualização tem-se a utilização de um *hardware*, um sistema operacional e uma ou mais aplicações. Nesse tipo de ambiente, quando se deseja usar diferentes sistemas operacionais é preciso usar servidores físicos distintos. Quando se tem consolidação de servidores através de um ambiente virtualizado, pode se

utilizar diversos sistemas operacionais e diversas aplicações em um único servidor físico, e assim não seria necessária a utilização de um novo *hardware* para cada serviço [VOGELS, 2008].

Tendo em vista que a utilização de servidores virtuais sugere uma diminuição do custo relacionado à manutenção de múltiplos servidores, é necessário avaliar o real custo/benefício desta utilização: até que ponto a redução de custos irá igualar a balança com relação à perda de desempenho? Este é o principal problema a ser analisado neste trabalho, descobrir até que ponto pode-se utilizar a virtualização para que se tenha uma relação custo/benefício aceitável.

Como instrumento de análise, serão feitos testes comparativos entre serviços executando diretamente em máquina física (não virtualizada) e serviços executando em máquinas virtuais (virtualizadas). Esta virtualização dar-se-á de diferentes formas, utilizando um serviço unicamente virtualizado e também a utilização de várias réplicas do mesmo serviço virtualizado. É importante garantir também que os testes realizados com a máquina física e os testes realizados com as máquinas virtuais sejam feitos precisamente no mesmo ambiente, ou seja, utilizem o mesmo *hardware* e o mesmo ambiente de rede, mantendo a consistência dos dados recolhidos.

Com relação ao ambiente de testes, inicialmente tem-se a idéia de utilizar uma máquina física como provedora do serviço (que será definido durante o desenvolvimento do trabalho) e duas ou três máquinas clientes realizando acessos a este serviço em uma rede local. Posteriormente, a mesma máquina física (*hardware*) servidora irá propiciar o uso do ambiente virtualizado.

A definição do ambiente de virtualização a ser explorado está direcionada para o uso de *software* livre com o intuito de eliminar restrições impostas por licenças proprietárias. Dentre as possibilidades levantadas inicialmente estão Xen, Linux Virtual Server e User-Mode Linux [FIGUEIREDO, 2005].

Os dados colhidos nos testes serão avaliados a fim de verificar qual a real usabilidade do conceito de virtualização, levando em consideração

principalmente o custo e o desempenho das diferentes abordagens, viabilizando ou não o uso de máquinas virtuais para múltiplos servidores.

Algumas dificuldades devem ser encontradas para desenvolvimento deste trabalho, uma delas será a implantação dos serviços sendo que deve-se buscar um conhecimento específico sobre a utilização do ambiente de virtualização. Outra dificuldade que deve ser encontrada é a definição das métricas e experimentos a serem utilizados para realizar os comparativos referentes a custo e desempenho, pois existem diversas métricas disponíveis mas será necessário selecionar as mais representativas para o problema em estudo.

2. Objetivos

O objetivo principal deste trabalho é realizar um estudo referente à implantação de múltiplos servidores através de uma máquina virtual, visando determinar a relação custo/benefício desse tipo de solução.

2.1 Objetivos Específicos

- Realizar um estudo sobre tecnologias de virtualização.
- Realizar um estudo sobre a utilização de máquina virtual específica.
- Analisar e implantar algum tipo de servidor, a fim de ser testado com medidas de desempenho.
- Analisar e implantar o mesmo tipo de servidor através de máquina virtual.
- Definir as métricas a serem utilizadas nos testes.
- Definir as ferramentas de avaliação.
- Efetuar testes com múltiplos servidores em um mesmo *hardware* através de máquinas virtuais.
- Avaliar resultados e realizar comparações.
- Apresentar resultado final baseado na análise custo/benefício.

3. Metodologia

Para que os objetivos propostos sejam alcançados será necessário o cumprimento de algumas etapas de desenvolvimento. A primeira etapa consiste em avaliar as tecnologias atuais referentes à utilização de máquinas virtuais e realizar uma pesquisa bibliográfica com o intuito de encontrar fontes e trabalhos realizados nessa área, bem como na área de servidores. Outro item importante a ser pesquisado é relacionado às medidas de desempenho a serem utilizadas no trabalho, Juntamente com buscar ferramentas que possam auxiliar neste processo.

Depois de realizados os estudos referentes à utilização de máquina virtual junto a servidores, um novo estudo deve ser realizado para levantar informações referentes ao uso de um ambiente de virtualização específico, bem como definir aspectos importantes como capacidade de uso e requisitos necessários.

Após o levantamento dos métodos de análise e os pontos a serem analisados, deve-se passar para a definição dos aspectos a serem avaliados. Primeiramente serão definidas as métricas a serem utilizadas como medida de desempenho. A seguir, será definida qual ferramenta será utilizada para coletar tais métricas. Esta definição virá após uma comparação das ferramentas disponíveis, sendo que pontos como facilidade de uso e diversificação de testes serão de alta relevância para seleção da ferramenta.

Terminadas as definições referentes a métricas e ferramentas utilizadas, deve se pensar nos experimentos a serem realizados, e o objetivo desta etapa é justamente definir estes experimentos.

A última etapa consiste em analisar os resultados obtidos nos experimentos, e com estes dados, definir as perdas ocorridas, ou não, sobre a utilização de máquina virtual para a aplicação de múltiplos servidores. Esta informação será necessária para apresentar uma perspectiva completa sobre a viabilidade do propósito apresentado no trabalho.

Depois de definidas as etapas principais, citadas acima, outras etapas devem ser consideradas para a construção do cronograma:

1. Pesquisa bibliográfica;
2. Estudo e definição do escopo do trabalho;
3. Desenvolvimento do Plano do Trabalho de Conclusão de Curso;
4. Correção e entrega definitiva do Plano do Trabalho de Conclusão de Curso;
5. Estudo sobre uso de máquina virtual e servidores;
6. Estudo sobre uso de um ambiente de virtualização específico;
7. Levantamento de medidas de desempenho e ferramentas;
8. Análise e definição de métricas e ferramentas a serem utilizadas;
9. Definição dos experimentos;
10. Escrita do texto (TCC1);
11. Realizar as implantações;
12. Realizar comparações entre os experimentos;
13. Analisar resultados obtidos;
14. Escrita do texto (TCC2).

4. Cronograma Proposto

Etapas	2008											
	Janeiro	Fevereiro	Março	Abril	Maió	Junho	Julho	Agosto	Setembro	Outubro	Novembro	Dezembro
1		X X X										
2		X X										
3		X X X										
4			X X									
5			X X X									
6			X X X X X									
7				X X								
8				X X X X								
9				X X X X								
10				X X X X X X								
11					X X X X X X X X							
12						X X X X X						
13							X X X X X X X					
14							X X X X X X X X X X X X X X X X					

5. Linha e Grupo de pesquisa

O grupo de pesquisa a ser utilizado para apoio ao desenvolvimento do trabalho é o GRADIS – Grupo de Redes e Aplicações Distribuídas da UDESC – Universidade do Estado de Santa Catarina. O trabalho se enquadra na linha de pesquisa de sistemas operacionais do grupo.

6. Forma de Acompanhamento/Orientação

O acompanhamento sobre o desenvolvimento do trabalho será realizado semanalmente, com reuniões realizadas entre orientador e aluno, sendo que a cada semana o aluno irá apresentar um relatório indicando situação na qual a elaboração da pesquisa se encontra. Estas reuniões estão marcadas para todas as segundas-feiras às 15h20min.

7. Referências Bibliográficas

CROSBY, Simon. ; BROWN, David. **The Virtualization Reality**, ACM QUEUE, Volume 4, Nº 10, P: 34 – 41. December 2006.

FIGUEIREDO, Renato. ; DINDA, Peter A. ; FORTES, José. **Resource Virtualization Renaissance**, IEEE Computer, Volume 85, Nº 5, P: 28 – 31. May 2005.

FRASER, Keir. ; HAND, Steven. ; NEUGEBAUER, Rolf. ; PRATT, Ian. ; WARFIELD, Andrew.; WILLIAMSON, Mark. **Safe Hardware Access With the Xen Virtual Machine Monitor**, University of Cambridge Computer Laboratory. 2004.

STALLINGS, W. **Redes e sistemas de comunicação de dados**. Rio de Janeiro, 2005.

VOGELS, Werner. **Beyond Server Consolidation**, ACM QUEUE, P: 21 – 26. January 2008.

Rafael Rodrigues Obelheiro

Ricardo Jose Pfitscher

Mauricio Aronne Pillon

REFERÊNCIAS BIBLIOGRÁFICAS

BARHAM, Paul. ; DRAGOVIC, Boris. ; FRASER, Keir. ; HAND, Steven.; HARRIS, Tim. ; HO, Alex. ; NEUGEBAUER, Rolf.; PRATT, Ian. ; WARFIELD, Andrew. **Xen and the Art of Virtualization**, University of Cambridge Computer Laboratory. October 2003.

CROSBY, Simon. ; BROWN, David. **The Virtualization Reality**, ACM QUEUE, Volume 4, Nº 10, P: 34 – 41. December 2006.

FIGUEIREDO, Renato. ; DINDA, Peter A. ; FORTES, José. **Resource Virtualization Renaissance**, Proceedings of the IEEE, Volume 85, Nº 5, P: 28 – 31. May 2005.

FRASER, Keir. ; HAND, Steven. ; NEUGEBAUER, Rolf. ; PRATT, Ian. ; WARFIELD, Andrew.; WILLIAMSON, Mark. **Safe Hardware Access With the Xen Virtual Machine Monitor**, University of Cambridge Computer Laboratory. 2004.

MAZIERO, Carlos Alberto. **Sistemas Operacionais**, cap. 8. Máquinas Virtuais. Disponível em: <<http://www.ppgia.pucpr.br/~maziero/ensino/livro/so-cap08.pdf>> Acessado em 01/04/2008.

MIDGLEY, Julian T. J. **The Linux HTTP Benchmarking HOWTO**. July, 2001. Disponível em: < <http://www.xenoclast.org/doc/benchmark/> > Acessado em 01/06/2008.

NANDA, Suzana; CHIUEH, Tzi-cker. **A Survey on Virtualization Technologies**. Research Proficiency Report. February 2005.

PAI, Vivek S.; Druschel, Peter; ZWAENEPOEL, Willy. **Flash: An Efficient and Portable Web Server**. Proceedings of the 1999 USENIX Annual Technical Conference. Monterey, California, USA, June 1999.

POPEK, Gerald; GOLDBERG, Robert. **Formal requirements for virtualizable third generation architectures**. Communications of the ACM, P: 412 – 421. July 1974.

SMITH, Jim; NAIR, Ravi. **Virtual Machines**. Morgan Kaufmann, 2005.

STALLINGS, W. **Redes e sistemas de comunicação de dados**. Rio de Janeiro, 2005.

TANENBAUM, Andrew S.; WOODHULL, Albert S. **Operating System: Design and Implementation**. Prentice Hall. 2006.

VIRTUALBOX. **VirtualBox, Professional, Flexible, Open**. Acessado em: 04/04/2008. Disponível em: <<http://www.virtualbox.org/wiki/VirtualBox>>

VMWARE, **Understanding Full Virtualization, Paravirtualization, and Hardware Assist**. 2007.

VOGELS, Werner. **Beyond Server Consolidation**, ACM QUEUE, P: 21 – 26. January 2008.

WHITAKER, Andrew; SHAW, Marianne; GRIBBLE, Steven D. **Denali: Lightweight Virtual Machines for Distributed and Networked Applications**. The University of Washington, 2002.

XEN. **Xen.org.** Acessado em: 04/04/2008. Disponível em:
<<http://www.xen.org/about/>>