

Comparação de desempenho de Máquinas Virtuais na arquitetura XenServer

Darlon H. Fracari¹, Sérgio L. Dill¹, Edson L. Padoin¹

Universidade Regional do Noroeste do Estado do Rio Grande do Sul (UNIJUI)

¹Departamento de Tecnologia (DETEC) – Santa Rosa – RS - Brasil

{darlon.fracari, dill, padoin}@unijui.edu.br

1. Introdução

A virtualização tem se tornado cada dia mais presente e utilizada pelas equipes de TI. Contudo, nos últimos anos esta tecnologia tem apresentado um crescimento sólido e robusto, além das reduções de custos, do aumento da confiabilidade, da escalabilidade e do excelente balanceamento de carga de algumas soluções. Contudo, trata-se de uma tecnologia desenvolvida na década de 1960 pela IBM, usada em Mainframes no OS/370 que hoje passa a ser novamente empregada [Candido 2007].

Com o ressurgimento da virtualização na década de 1990, alguns dos grandes mitos da área de TI foram superados, ou ao menos mudaram de posicionamento: a eterna disputa entre “preferência de investimento em *hardware*” e “preferência em *software*” deixa de fazer sentido quando o que importa é ter um bom *hardware* centralizado e uma maneira produtiva de criar e administrar máquinas virtuais (MV) [Carmona 2008].

Atualmente existem muitos monitores de máquinas virtuais (MMV) no mercado, alguns que empregam a Virtualização Completa e outros a Paravirtualização. Na Virtualização Completa, o sistema *Host* compartilha recursos de *hardware* com a máquina *Guest*. Para tal recurso ser utilizado, é necessário utilizar *hardware* diferenciado, como a tecnologia Intel-VT e AMD-V [Gonçalves 2009]. Já na Paravirtualização, o Sistema *Host* sofre modificações para que a interação com o MMV seja mais eficiente, acessando assim diretamente os recursos de *hardware*, porém limitando a sua compatibilidade e suporte [Laureano 2006].

O objetivo do presente trabalho é avaliar o ganho/perda no uso da virtualização utilizando ambientes de trabalhos *Windows* e *Linux*. Para tal foi necessário utilizar uma estrutura adequada para que fossem realizados os testes de desempenho de CPU, Alocação de Memória e ES.

Este trabalho está assim organizado. Na seção 2 são abordadas a arquitetura e as características do processador utilizado para os estudos de caso. Na seção 3 são apresentados os problemas utilizados para validação de desempenho. Na seção 4 aborda-se o ambiente de execução utilizado para criar as MV e a realização dos testes. Por fim são apresentados os resultados e as considerações finais.

2. Arquitetura do Processador

O processador AMD utilizado neste trabalho foi lançado no ano de 2007 e utiliza a tecnologia AMD-V, esta que possui uma área de TLB (*Translation lookaside Buffer*) rotulada. Os hospedes são mapeados para um espaço de endereços que pode ser completamente diferente do que a MV utiliza. O motivo de ser chamada de área rotulada é que a TLB possui informações adicionais conhecidas como Identificadores de Espaço de Endereço (ASID), estes que são responsáveis por garantir que a TLB não

tenha de ser completamente renovada a cada vez que um chaveamento de contexto ocorre. A AMD desenvolveu uma tecnologia para controlar o acesso a chamada Unidade de Gerenciamento de Memória de E/S (IOMMU) que é responsável pela E/S das MV, incluindo limitar o DMA para o espaço válido e atribuindo diretamente dispositivos reais para elas [Mathews 2009].

Além disso, existe o, chamado Mapa de Permissão de Entrada e Saída (IOPM), que determina para cada uma das portas de entrada e saída, se o acesso a uma exceção deve ser interceptado, sendo possível alocar cada um dos dispositivos a um sistema hóspede exclusivo. Com o Dispositivo de Exclusão de Vetor (DEV), a *northbridge* do processador verifica, em cada acesso direto de memória de um dispositivo (PCI), em um vetor de bit, se o acesso é permitido à respectiva página da memória. Um acesso indevido a memória ativaria uma Resposta de Aborto Mestre (MARSP), no dispositivo e interromperia a transação como uma tentativa de escape que depende de manipular o acesso indevido diretamente [Carmona 2008].

3. Definição do Problema

No intuito de explorar os recursos oferecidos pelo processador *multi-core* com virtualização do processador AMD, foram executados 3 algoritmos: I) Solução do modelo matemático que descreve o movimento da água no solo e da absorção da água pelas raízes de plantas [Padoin 2008], II) Aplicativo de simulação de E/S e III) Aplicativo de Alocação de Memória. Todos os cálculos foram realizados através em MV em rede no ambiente computacional do XenServer.

Para realizar os estudos de caso, foram utilizados 2 microcomputadores interconectados através de um *switch fast ethernet 100baseT Full Duplex*, que serão abordados na seção 4.

4. Ambiente de Trabalho

Os estudos de caso foram realizados em um microcomputador com Processador *AMD Turion™ 64 X2 Dual-Core Mobile Technology TL-60* de 2.0 GHz, núcleo Tyler, 2 x 512 KB de *Cache L2* dedicado, *FSB HyperTransport 800 MHz*, 4GB *DDR2* de memória *RAM*, *XenServer Express Edition 5.5.0-15119p 64 Bits (kernel 2.6.18-128)*.

Como terminal de acesso foi utilizado um microcomputador com Processador *Intel Core™ 2 Duo E7500* de 2.93 GHz, núcleo *Wolfdale*, 3MB de *Cache L2*, *FSB 1066 MHz*, 3GB *DDR2* de memória *RAM*, *Windows XP Professional SP2 32 Bits* e fazendo acesso ao servidor através do aplicativo *Citrix XenCenter 5.5.0*.

Para interconexão foi utilizado *switch central Fast Ethernet 100baseT Full Duplex*. Os SOs utilizados para análise de desempenho nas MV foram: *Ubuntu 9.04 32 e 64 Bits* (ambos com *Kernel 2.6.28-16-default* e com o compilador *GCC Versão 4.3.3-4*), *Windows Server 2003 32 Bits* com o compilador *Dev-C Versão 4.9.9.2*, *Windows XP Professional SP2 32 Bits* com o compilador *Dev-C Versão 4.9.9.2*.

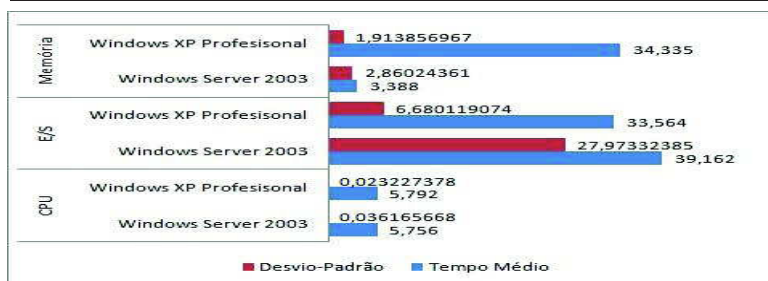
5. Estudos de Caso e Resultados Obtidos

O primeiro estudo de caso tem como objetivo analisar o desempenho dos aplicativos com característica *CPU-Bound*, *E/S* e Alocação de Memória nas MV. Para tanto foram criadas 2 MV, cada uma com 8 GB alocados em disco e 512 MB de memória *RAM*, contendo uma o Sistema Operacional Microsoft *Windows XP Professional SP2 32 bits* e na outra Microsoft *Windows 2003 Server*.

Na Tabela 1 é apresentado o tempo médio das 10 execuções, bem como o desvio padrão dos testes realizados com os SOs *Windows XP* e *Windows 2003 Server*.

Tabela 1 – Tempos de execução do primeiro estudo de caso (em segundos)

Sistema Operacional	Teste	Tempo Médio	Desvio-Padrão
Windows XP Profissional	CPU	5,792	$\pm 0,0232227378$
	E/S	33,564	$\pm 6,680119074$
	Memória	34,335	$\pm 1,913856967$
Windows 2003 Server	CPU	5,756	$\pm 0,036165668$
	E/S	39,162	$\pm 27,97332385$
	Memória	3,114	$\pm 0,036884806$



No segundo estudo de caso o objetivo, semelhante ao primeiro, é analisar o desempenho de processamento com característica *CPU-Bound*, E/S e Alocação de Memória, obtido em MV com o Sistema Operacional Linux Ubuntu 9.04 – 32 *bits* e 64 *bits* com a seguinte configuração: 8 GB alocados em disco e 512 MB de memória RAM.

Na Tabela 2 é apresentado o tempo médio das 10 execuções do segundo estudo de caso, bem como o desvio padrão dos testes realizados com o SO Linux Ubuntu 9.04 32 e 64 bits kernel 2.6.28-16-*default*.

Tabela 2 – Tempos de execução do segundo estudo de caso (em segundos)

Sistema Operacional	Teste	Tempo Médio	Desvio-Padrão
Ubuntu 9.04 32 Bits	CPU	7,401	$\pm 0,271232352$
	E/S	63,892	$\pm 11,94337947$
	Memória	3,388	$\pm 2,86024361$
Ubuntu 9.04 64 Bits	CPU	7,185	$\pm 0,605918073$
	E/S	65,438	$\pm 8,521050249$
	Memória	9,672	$\pm 0,83590999$



6. Considerações Finais

Analisando os tempos obtidos nas execuções dos testes em cada uma das MV criadas no XenServer, pode-se observar que o uso da virtualização apresentam variações de desempenho para diferentes SO. Se compararmos os tempos para execução do modelo matemático que descreve o movimento da água no solo, uma aplicação do tipo *CPU-bound*, observa-se que foram muito semelhantes variando de 5,7 a 7,4 segundos. Porém, observa-se uma grande variação de tempo para aplicações do tipo E/S que gera um arquivo de 956 Mb na MV que esta alocada e sendo executada pelo XenServer. Tal arquivo que quando criado gera um *delay* por estar sendo criado através de uma rede 100baseT e Alocação de Memória. Estas que podem ser superadas através da utilização de *hardware* específico e uma estrutura de rede adequada com o intuito de obter-se um melhor desempenho. Por exemplo, utilizar tecnologia *Gigabit Ethernet* e HDs *SCSI* ou *SAS* para reduzir o tempo de acesso quando executado o aplicativo do tipo E/S.

Já quanto a utilização de MV com sistemas operacionais Linux de 32 bits e 64 bits, ambos apresentaram um desempenho muito semelhantes. Já em ambiente windows é possível observar uma diferença maior de tempos de processamento.

Muitas pesquisas vem sendo realizadas superando tais desafios. Assim produtos novos incorporam estas inovações caracterizando cada vez mais esta tecnologia como solução das atuais demandas.

Referências

- CANDIDO, Caio (2007) *História da Virtualização*, Disponível em: <http://caiocandido.wordpress.com/2007/05/19/conceitos-de-virtualizacao>.
- GONÇALVES, Danilo Brandão; JUNIOR, José Claudio Vahl (2009). *White Paper - Virtualização*. Disponível em http://www.sensedia.com/br/anexos/wp_virtualizacao.pdf.
- LAUREANO, Marcos. (2006). *Máquinas Virtuais e Emuladores: Conceitos, Técnicas e Aplicações*. Novatec. p. 184.
- MATHEWS, Jeanna N.; DOW, Eli M; DESHANE, Todd; HU, Wenjin; BONGIO, Jeremy; WILBUR, Patrick F.; JOHNSON, Brendan. (2009). *Executando o Xen Um Guia Prático para a Arte da Virtualização*: Alta Books. p. 616.
- PADOIN, E. L.; BORGES, Pedro A.P.; DILL, Sérgio L; PADILHA, Nelson R. (2008). Resolução do problema de absorção da água do solo através da paralelização do problema inverso. In: *VIII Escola Regional de Alto Desempenho*, 2008, Santa Cruz do Sul: Evangraf. v. 1. p. 268-271
- CARMONA, Tadeu (2008). In: *Virtualização – Coleção Linux Technical Review*: Linux New Media do Brasil Editora Ltda, São Paulo SP. p. 14-17.