

# Monitoramento e Análise do Impacto no Desempenho em Ambientes Virtualizados

Pedro Freire Popiolek, Odorico Machado Mendizabal  
Centro de Ciências Computacionais  
Universidade Federal do Rio Grande – FURG  
Campus Carreiros: Av. Itália km 8 Bairro Carreiros, Rio Grande, Brasil  
{p.f.popiolek,odoricomendizabal}@furg.br

**Resumo**—Embora não seja uma prática inovadora, a utilização de infraestruturas virtualizadas é amplamente empregada em computação em nuvem. Porém, o impacto no desempenho ocasionado por tais infraestruturas torna-se um dos maiores desafios para que rígidos acordos de níveis de serviço (SLA) sejam respeitados. Este trabalho apresenta técnicas de monitoramento e realiza a análise de desempenho em um ambiente virtualizado. Com o uso de cargas sintéticas sobre a infraestrutura monitorada, é possível identificar as principais causas da degradação de desempenho observada.

## I. INTRODUÇÃO

Sistemas computacionais modernos fazem uso de infraestruturas computacionais amplamente distribuídas, possibilitando a implementação de aplicações colaborativas e o compartilhamento de recursos remotos. Tais sistemas têm como principais objetivos melhorar o desempenho, escalabilidade e disponibilidade dos serviços oferecidos, além de proporcionar uma melhor utilização de recursos.

Nesse contexto, computação em nuvem, associada ao uso de virtualização, permite que um conjunto de servidores físicos hospede dezenas ou mesmo centenas de máquinas virtuais. Dessa forma, a escalabilidade de sistemas é ampliada, garantindo a maximização na utilização de recursos. Um grande desafio associado aos serviços oferecidos por esse tipo de infraestrutura é garantir que as máquinas hospedeiras não sejam sobrecarregadas. Uma vez que a carga de trabalho aplicada em uma máquina física excede seu poder computacional, todas as máquinas virtuais executando nesse hardware sofrerão degradação de desempenho, impactando as aplicações em uso.

Este trabalho descreve os primeiros passos no desenvolvimento de mecanismos de monitoramento e detecção proativa de gargalos de desempenho para computação em nuvem. Ao implementar tais mecanismos, será possível prever instabilidade e degradação em máquinas virtuais com razoável antecedência, permitindo a adoção de políticas eficientes para gerenciamento dinâmico de recursos.

Como resultados preliminares desta pesquisa, é apresentado um estudo sobre monitoramento de recursos em sistemas operacionais Windows e Linux. Sem uma análise exaustiva, este artigo identifica um conjunto mínimo de contadores de desempenho a serem observados e, a realização de experimentos indica como as métricas coletadas auxiliam na previsão de degradação de desempenho.

Este artigo está organizado da seguinte maneira. A próxima seção introduz o conceito de máquinas virtuais. A Seção III apresenta ferramentas de monitoramento para plataformas Windows e Linux. Os experimentos realizados são descritos na Seção IV e as conclusões deste trabalho aparecem na Seção V.

## II. MÁQUINAS VIRTUAIS

O uso massivo de máquinas virtuais (VM - *Virtual Machines*) é uma das principais características das infraestruturas de computação em nuvem. A técnica de virtualização permite o particionamento de recursos de servidores físicos entre vários servidores virtuais que executam concomitantemente em uma mesma máquina física. Dentre as vantagens dessa abordagem estão o aumento de escalabilidade da própria infraestrutura e a otimização no uso dos recursos computacionais disponíveis.

Em ambientes virtualizados, além do SO (Sistema Operacional) hospedeiro, que executa sobre o hardware físico, instâncias de VMs e SOs hóspedes, que executam sobre hardwares virtuais, também existe um monitor (VMM - Virtual Machine Monitor). O VMM, também conhecido por *hypervisor*, é responsável pelo gerenciamento e controle dos recursos compartilhados pela máquina física (por exemplo, CPU, memória e dispositivos de E/S). Dentre outras tarefas, o VMM é responsável pelo escalonamento e o tratamento de instruções das VMs, fornecendo transparência aos SOs visitantes sobre o hardware onde esses executam [1]. A Figura 1 representa uma possível disposição dos elementos descritos.

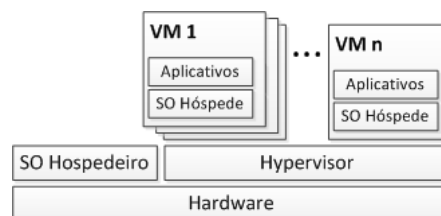


Figura 1. Virtualização híbrida.

No entanto, é comum observar sobrecarga na utilização de recursos em sistemas virtualizados. Supondo uma mesma arquitetura configurada com máquinas virtuais ou apenas com uma máquina física, as VMs não

mantêm o mesmo desempenho que a máquina física ao tratar uma carga de trabalho semelhante. O impacto desse comportamento varia de acordo com o tipo de carga de trabalho demandada pelas aplicações. Em suma, esse custo é ocasionado pelo acréscimo de camadas de software que realizam emulações, virtualização de dispositivos, e ao compartilhamento competitivo de recursos entre VMs [2].

Outro aspecto importante no desempenho desses ambientes é o escalonamento feito pelo VMM. Tradicionalmente o escalonamento prioriza a partilha de recursos do processador entre as VMs existentes, deixando em segundo plano o agendamento de operações de E/S. Por esse motivo aplicações *I/O-bound* apresentam perda de desempenho, ou seja, o atraso na execução das operações torna-se um gargalo [3].

Algumas empresas que oferecem serviços em nuvem computacional possuem *hypervisors* próprios, visando melhor aproveitamento do tipo específico de suas arquiteturas computacionais. A empresa Amazon utiliza o Xen Hypervisor com alterações personalizadas no código fonte. O Windows Azure utiliza o Wazure Hypervisor, solução não comercial, baseada em uma solução fornecida pela própria empresa, o Hyper-V.

Com este trabalho e futuros estudos, pretende-se criar um mecanismo de detecção preventiva de degradação de desempenho em ambientes virtualizados. Porém, a atual pesquisa preocupa-se somente em entender e avaliar aspectos de desempenho desses ambientes decorrentes do escalonamento realizado pelo VMM.

### III. TÉCNICAS DE MONITORAMENTO

Esta seção descreve como um sistema operacional permite observar um histórico sobre o seu desempenho. A obtenção desse histórico dá-se através de ferramentas de monitoramento, que coletam dados representando níveis de utilização de recursos de hardware e sistema operacional. A análise desses dados de monitoramento, coletados em intervalos regulares de tempo, permite identificar características do sistema, detectar gargalos de desempenho ou ainda correlacionar dados na busca de causa para um comportamento observado.

Monitorar recursos do sistema para detecção de perfis de carga requer conhecimento sobre as métricas de monitoramento disponíveis. Embora as métricas variem de acordo com o sistema operacional utilizado, é possível estabelecer um conjunto de métricas comuns a qualquer sistema. As principais métricas identificadas são de memória, disco, rede e processador. A HP apresenta um estudo detalhado sobre métricas de monitoramento para o sistema operacional Windows [4]. Cada métrica possui um conjunto de contadores e cada contador fornece um valor numérico que representa uma informação específica de utilização do recurso (por exemplo, Interrupções/s é um contador do processador).

Sistemas operacionais Windows possuem uma ferramenta nativa, chamada *Performance Monitor*, responsável pelo monitoramento de recursos do sistema. As métricas de monitoramento são representadas por objetos e cada

objeto disponível pode possuir uma ou mais instâncias. As instâncias servem para diferenciar elementos não unitários do sistema.

Em sistemas operacionais Linux, é possível extrair dados para monitoramento de recursos do sistema através de arquivos virtuais existentes no diretório */proc/*. Porém, um conjunto de ferramentas presentes no pacote *sysstat* automatizam essa tarefa, sendo as principais: *Sar*, *Iostat* e *Pidstat*.

A Tabela I contém alguns exemplos de contadores de desempenho presentes tanto em Windows quanto em Linux. Embora alguns contadores sejam específicos de um determinado sistema operacional, é possível, em alguns casos, correlacionar métricas observadas em outros contadores para a obtenção uma métrica equivalente.

Tabela I  
EXEMPLOS DE CONTADORES DE DESEMPENHO

Windows	Linux	Descrição
Tamanho da Fila de CPU	runq-sz	Quantidade de processos em estado pronto e enfileirados.
% tempo de interrupção	% irq + % soft	Média percentual de uso de CPU atendendo interrupções.
Transferências de disco/s	r/s + w/s	Requisições de disco completadas por segundo.

### IV. EXPERIMENTOS

Serão apresentados nesta seção experimentos que visam expor características de gerenciamento de recursos realizado por *hypervisor*, com enfoque no desempenho de sistemas virtualizados. Foram utilizadas cargas de trabalho sintéticas aplicadas em máquinas virtuais, reproduzindo o comportamento de processos *CPU-bound* e *I/O-bound*.

Para realização dos experimentos, foi utilizada uma máquina hospedeira com processador AMD FX-6100 (seis núcleos), 8GB de memória RAM, 500GB de disco rígido, sistema operacional Ubuntu 12.04 64 bit e como *hypervisor* o KVM. As máquinas virtuais hospedadas foram configuradas cada uma com processador QEMU 64, 1GB de memória RAM, 20 GB de disco virtual pré-alocado e sistema operacional Windows XP Professional SP3 32 bit.

Foram criados seis cenários diferenciando o número de VMs ativas, sendo seis a quantidade máxima. Ao total foram realizados doze experimentos, seis para cada tipo de carga de trabalho nos diferentes cenários existentes. Durante o ensaio, todas VMs em execução recebem o mesmo tipo e volume de carga de trabalho.

Para execução de processo *CPU-bound* foi utilizado o aplicativo *WPrime Benchmark v2.09* [5]. A opção de processamento “1024M” foi selecionada nos testes, ou seja, é calculada a raiz quadrada dos primeiros 1024 milhões números inteiros. Como ponto de partida para análise da utilização de recursos por ambientes virtualizados, observe os tempos de execução desse *benchmark* na Tabela II.

A partir de 3 VMs executando concorrentemente, conforme aumenta o número de VMs em execução, mais tempo leva para a aplicação finalizar. Dentre os fatores desse atraso na execução das tarefas estão o escalonamento

Tabela II  
DURAÇÃO MÉDIA DE EXPERIMENTOS CPU-bound

VMs em Execução	Média do Tempo de Processamento
1	34.98 min
2	34.04 min
3	34.82 min
4	39.70 min
5	43.58 min
6	48.54 min

entre VMs e a competição por recursos da máquina hospedeira. Com base nos monitores de desempenho do sistema observa-se que com um número elevado de VMs o percentual de tempo destinado a cada VM reduz, enquanto o tempo de execução do hospedeiro em modo privilegiado aumenta. A Figura 2 exibe as métricas coletadas para porcentagem de uso de CPU para VMs (modo convidado), porcentagem de uso de CPU no modo privilegiado e porcentagem de CPU ociosa.

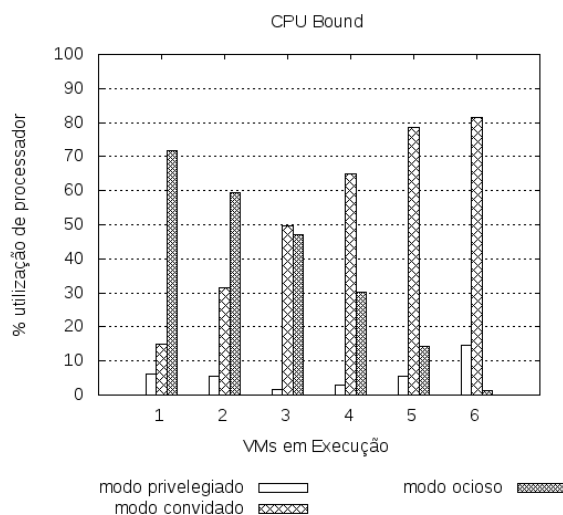


Figura 2. Utilização de CPU da máquina hospedeira durante experimentos CPU-bound

Observe que a carga de utilização da máquina hospedeira cresce linearmente (melhor visualizado pelo decréscimo linear de % CPU ociosa). No entanto, com o gerenciamento mais custoso do VMM, o aproveitamento de CPU por parte das VMs não cresce linearmente (veja o crescimento no % de CPU executando no modo convidado).

Para processos predominantemente I/O-bound, foi utilizado o benchmark *CrystalDiskMark 3.0.1* [6]. Como opção de carga de trabalho foi escolhida 2000MB de leitura/escrita sequencial. A Tabela III apresenta as taxas de leitura/escrita de cada experimento<sup>1</sup>.

Análogo ao experimento anterior, percebe-se que com

<sup>1</sup>Devido aos resultados obtidos em cada VM apresentarem variações consideráveis, optou-se por exibir cada valor obtido, ao invés da média.

Tabela III  
TAXA LEITURA/ESCRITA DE EXPERIMENTOS I/O-bound

VMs em Execução	Taxa de Leitura/Escrita de cada VM (MB/s)
1	65.02/14.40
2	69.3/5.205 - 72.97/5.265
3	65.57/3.296 - 69.03/3.320 - 63.65/6.770
4	47.61/2.460 - 62.75/2.358 - 58.86/2.251 - 14.72/3.165
5	9.619/1.793 - 10.61/1.755 - 10.89/1.741 - 30.38/1.707 - 59.21/2.809
6	6.506/1.411 - 6.954/1.423 - 6.574/1.396 - 6.799/1.442 - 11.08/1.260 - 45.29/2.284

um número elevado de VMs há um decréscimo considerável na taxa de operações de leitura e escrita. É possível observar na Figura 3 que a oferta de processamento em modo convidado não cresce proporcionalmente ao número de VMs em execução. Esse fato pode ser explicado com base na percentagem crescente de espera de CPU por operações de E/S.

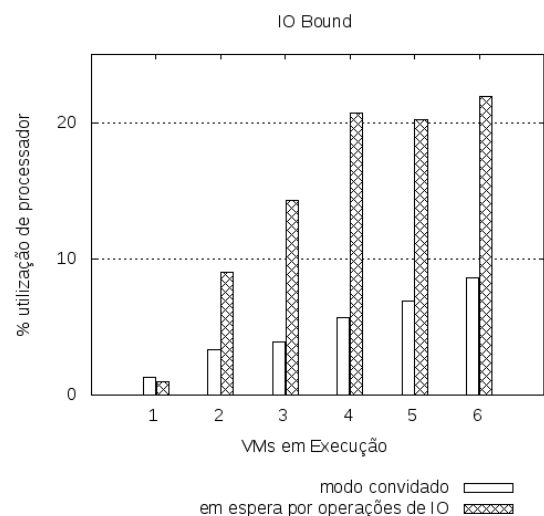


Figura 3. Utilização de CPU da máquina hospedeira durante experimentos I/O-bound

Ainda que cada VM possua o seu próprio disco virtual, elas compartilham um mesmo disco rígido físico. O uso intensivo de operações de leitura/escrita nessa unidade de armazenamento acarreta a degradação no desempenho de cada VM, podendo ser observada através da fila de requisições ao disco rígido do hospedeiro (vide Figura 4). Essa competição por recursos da máquina física explica o crescimento não linear na utilização de CPU em espera por operações de E/S, assim como a queda de utilização de processamento destinada às VMs.

Os resultados deste trabalho reafirmam que a virtualização de servidores adiciona um custo ao desempenho do sistema. No entanto, com a observação de métricas obtidas

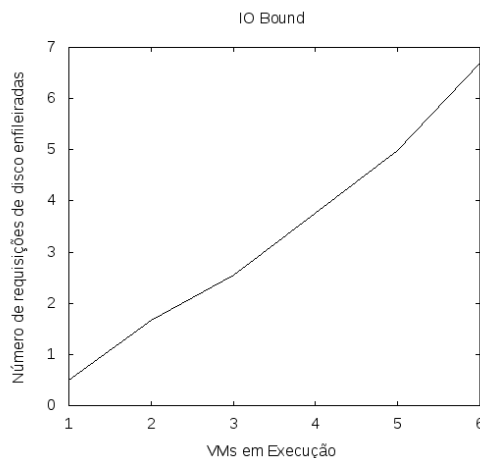


Figura 4. Formação de fila de requisições de disco durante experimentos *IO Bound*

por contadores de desempenho, é possível compreender as causas e relacionar os efeitos que levam à degradação do sistema. Apesar do conjunto de contadores descrito neste artigo ser reduzido, outras métricas também foram observadas e há indícios de que haja correlação entre vários contadores.

Futuramente pretende-se fazer uma análise estatística para comprovar a dependência entre métricas e estabelecer valores limites aos quais o sistema pode operar sem ocasionar perda de desempenho significativa. Além disso, outras métricas serão estudadas, incluindo métricas obtidas pelo monitoramento de interfaces de rede ou considerando infraestruturas com múltiplos discos rígidos.

Estudo similar ao apresentado neste trabalho foi realizado por Menon et al. em [7]. No entanto diferencia-se na utilização de ferramenta de monitoramento própria e trás problemas de desempenho de rede detectados. Gupta et al. consegue resolver alguns problemas de desempenho acrescentando isolamento extra entre as VMs [8]. Como resultado final desta pesquisa que se inicia, pretende-se desenvolver uma solução com abordagem diferente da apresentada no último trabalho citado. Almeja-se detectar problemas de desempenho em tempo de execução e com antecedência, de forma que seja possível realocar VMs para outro servidor presente na infraestrutura com o mínimo de impacto no desempenho dos sistemas.

## V. CONCLUSÃO

Embora o gerenciamento de recursos em computação em nuvem se beneficie da dinamicidade de infraestruturas virtuais, os impactos no desempenho decorrentes da virtualização não são detectados facilmente. Portanto, garantir acordos de níveis de serviço (SLA) nesses ambientes ainda é um grande desafio.

Este trabalho apresenta recursos para monitoramento em plataformas Windows e Linux, a serem utilizados na análise de desempenho de infraestruturas virtuais. Através da realização de experimentos, foi possível destacar efeitos da

degradação de desempenho decorrentes da virtualização, além de identificar a causa raiz com base em métricas coletadas por contadores de desempenho.

Os experimentos utilizaram *benchmarks* para estimular o sistema com cargas de trabalho simulando aplicações *CPU-bound* e *I/O-bound*. Essa abordagem evidencia padrões de uso de CPU, memória e disco em situações extremas. Embora essas cargas de trabalho não representem a grande maioria das aplicações, elas auxiliam na compreensão da degradação de desempenho e possibilitarão a identificação de correlação entre métricas coletadas por diferentes contadores de desempenhos. Futuramente, pretende-se usar essas correlações para desenvolver algoritmos capazes de identificar níveis de desempenho associados às VMs em execução e prever com antecedência a sobrecarga em máquinas hospedeiras. Em posse dessas informações, medidas preventivas permitirão um gerenciamento de recursos mais equilibrado, possibilitando que infraestruturas de computação em nuvem possam atender SLA mais rigorosos.

Em sequência a este trabalho, pretende-se realizar novos experimentos com utilização de padrões de carga mais realistas, como por exemplo padrões estabelecidos pelo consórcio TPC [9]. Além disso, é necessário identificar um maior número de correlações entre métricas de contadores de desempenho.

## REFERÊNCIAS

- [1] J. E. Smith and R. Nair, "An overview of virtual machine architectures," *Elsevier Science*, vol. 26, pp. 1–21, 2003.
- [2] R. McDougall and J. Anderson, "Virtualization performance: perspectives and challenges ahead," *SIGOPS Oper. Syst. Rev.*, vol. 44, 2010.
- [3] Y. Hu, X. Long, J. Zhang, J. He, and L. Xia, "I/O scheduling model of virtual machine based on multi-core dynamic partitioning," in *Proceedings of the 19th ACM HPDC*, 2010.
- [4] *HP Performance Engineering Best Practices*, HP, (Acesso em agosto de 2012). <http://h30499.www3.hp.com/t5/HP-LoadRunner-and-Performance/HP-Performance-Engineering-Best-Practices-Series/ba-p/2407627>.
- [5] wPrime. (Acesso em agosto de 2012). <http://www.wprime.net>.
- [6] CrystalMark, (Acesso em agosto de 2012). <http://crystalmark.info/software/CrystalDiskMark/index-e.html>.
- [7] A. Menon, J. R. Santos, Y. Turner, G. J. Janakiraman, and W. Zwaenepoel, "Diagnosing performance overheads in the xen virtual machine environment," in *Proceedings of the 1st ACM/USENIX international conference on Virtual execution environments*, ser. VEE '05. New York, NY, USA: ACM, 2005, pp. 13–23.
- [8] D. Gupta, L. Cherkasova, R. Gardner, and A. Vahdat, "Enforcing performance isolation across virtual machines in xen," in *Proceedings of the ACM/IFIP/USENIX 2006 International Conference on Middleware*. Springer-Verlag New York, Inc., 2006. <http://dl.acm.org/citation.cfm?id=1515984.1516011>.
- [9] TPC. (Acesso em agosto de 2012). <http://www.tpc.org>.