

# Avaliação dos Hipervisores VMware ESXi e GNU/Linux KVM para Computação de Alto Desempenho

Willian da Silva<sup>1</sup>, Raul Leiria<sup>2</sup>, Claudio Schepke<sup>1</sup>

<sup>1</sup>Universidade Federal do Pampa (UNIPAMPA)  
Alegrete – RS – Brasil

<sup>2</sup>Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)  
Porto Alegre – RS – Brasil

dasilva.williansoares@gmail.com, raul.leiria@acad.pucrs.br

claudioschepke@unipampa.edu.br

**Resumo.** *Este artigo tem como objetivo apresentar uma proposta de desenvolvimento de um script, para realizar a avaliação de desempenho automatizada, com o auxílio da suíte de benchmarks NPB-MPI, entre duas tecnologias de virtualização (VMware ESXi e KVM), em ambientes de computação de alto desempenho. Com base nos resultados obtidos, o hypervisor que teve melhor desempenho pode ser considerado mais adequado para aplicações de HPC.*

## 1. Introdução

O uso da virtualização para Computação de Alto Desempenho - *High Performance Computing* (HPC) - traz vantagens, como por exemplo, o seu uso em nuvens computacionais [Vogel et al. 2016]. Além disso, essa tecnologia também permite que usuários executem diferentes tipos de sistemas operacionais de forma concomitante em um mesmo servidor físico [Buyya et al. 2013].

Apesar das vantagens obtidas com a virtualização, há um custo. O *hypervisor*, primeira camada mais perto do *hardware*, que é responsável pela virtualização, acrescenta uma sobrecarga pelo fato de adicionar uma camada de abstração entre o ambiente virtualizado e os recursos físicos do servidor que ela utiliza [Maliszewski et al. 2018]. Vários aspectos do *software* e sistema operacional podem afetar o desempenho dos virtualizadores e máquinas virtuais como, por exemplo, a maneira que o *hypervisor* escala os recursos como CPU, memória, disco e rede. Contudo, os *hypervisors* apresentam diferentes tipos de desempenho [Hwang et al. 2013].

Neste trabalho, é proposta uma análise comparativa entre dois *hypervisors*: o KVM [KVM 2016] e o VMware ESXi [VMware 2018]. Os testes serão realizados em diferentes cenários, focando no paralelismo das aplicações. O desenvolvimento de um *script* para a automatização das etapas de avaliação terá as seguintes funções: automatização do processo das execuções dos *benchmarks* da suíte *The NAS Parallel Benchmarks* (NPB) - *Message Passing Interface* (MPI), coleta dos resultados das execuções dos *benchmarks* e geração dos gráficos a partir dos resultados obtidos. Por fim, é realizada a comparação dos resultados e a verificação do virtualizador que teve melhor desempenho de acordo com os experimentos.

## 2. Trabalhos relacionados

A comparação de desempenho entre virtualizadores vem sendo amplamente pesquisada, como no trabalho de [Hwang et al. 2013], onde foi realizada a análise de 4 *hypervisors* diferentes: Hyper-V, KVM, vSphere e Xen. Os virtualizadores foram avaliados com o auxílio de *benchmarks*: *Bytemark* para a CPU, *Ramspeed*, ferramenta que avalia o desempenho da *cache* e da largura de banda da memória, *Bonnie++* e *FileBench*. O primeiro é um *benchmark* de vazão de disco, enquanto o segundo é um gerador de *workload*, que simula um servidor de *emails*, de arquivos e de Web. O *Netperf* é utilizado para avaliar o desempenho dos recursos de rede.

Nos resultados obtidos, *vSphere* foi o *hypervisor* que teve melhor desempenho. Porém, de acordo com o autor, é difícil afirmar que um *hypervisor* será a melhor escolha para qualquer tipo de aplicação. O autor defende que diferentes tipos de aplicações podem necessitar de diferentes tipos de *hypervisors*. O autor não utiliza o NPB-MPI e também não comenta sobre uma forma automatizada de avaliação.

Em [Vogel et al. 2016] foi avaliado apenas o desempenho do *hypervisor* KVM em ambientes de nuvens computacionais (*Openstack*, *CloudStack* e *OpenNebula*). Foram utilizadas aplicações científicas para identificar possíveis impactos de desempenho causados por essas ferramentas de gerenciamento de nuvem. Os testes foram realizados utilizando os *benchmarks* da suite NPB com OpenMP e MPI. Os experimentos revelaram que não há diferença de desempenho significativa entre as ferramentas da nuvem. Nesse trabalho, o autor também não comenta detalhes sobre uma possível automatização dos testes.

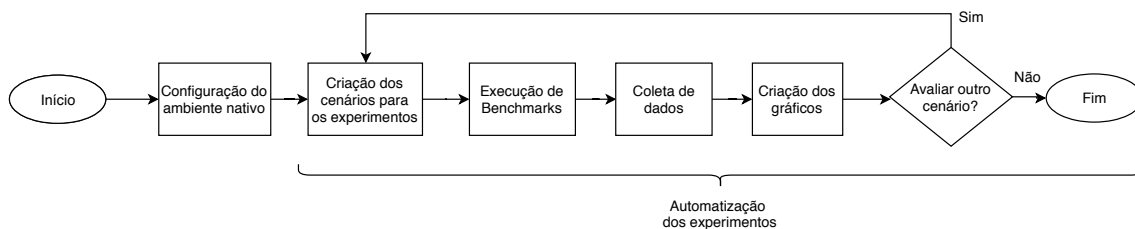
No trabalho de [Maliszewski et al. 2018] foi realizada uma avaliação comparativa de aplicações científicas, com uma instância e múltiplas instâncias em nuvens computacionais, utilizando as tecnologias de virtualização KVM e LXC em uma nuvem privada na plataforma *Cloudstack*. Para a avaliação realizada nesse trabalho, foi utilizado o NPB-OMP para simular diferentes tipos de *workloads*. Nesse experimento, foram utilizados três servidores executando o sistema operacional Ubuntu. A estrutura da nuvem computacional consistiu em um nó *front-end* para administração e outros dois nós para executar os *workloads* e as aplicações. Os autores concluíram que em 93,75% dos resultados LXC obteve melhor desempenho do que KVM.

De acordo com os trabalhos [Hwang et al. 2013] e [Maliszewski et al. 2018], o desempenho dos *hypervisors* pode variar de acordo com o tipo de aplicação que se deseja executar. Porém, não é trivial a maneira com que os experimentos foram conduzidos para chegar a tal conclusão, ou seja, são difíceis de serem reproduzidos. Duas das contribuições deste trabalho são (i) realizar experimentos que sejam reproduzíveis por meio de um *script* de automatização; (ii) verificar a viabilidade de utilização do *hypervisor* VMware ESXi para HPC, devido a sua considerável taxa de utilização na indústria.

## 3. Metodologia

A fim de avaliar o desempenho dos dois virtualizadores, ao executar aplicações científicas, será desenvolvido um *script* que automatizará as etapas de: criação dos cenários, execução dos *benchmarks*, *parsing* dos dados e criação de gráficos. Para um *baseline*, os mesmos experimentos serão realizados no ambiente nativo. Todos os experimentos serão executados em um servidor físico, para que cada *hypervisor* possa ser avaliado de forma isolada em diferentes cenários.

Os cenários serão configurados desta maneira: 1 *Guest* (1 máquina virtual) e *nGuests* (n máquinas virtuais). Para cada cenário, é definido para as VMs o mesmo SO (Ubuntu Server) e as mesmas configurações de *hardware*: o número de vCPUs, a quantidade de memória RAM e o tamanho do HD virtualizado. Com o cenário configurado, o *script* executará os *benchmarks* e ocorrerá o *parsing* dos dados. A última etapa do experimento consiste em obter as informações da etapa anterior para criar os gráficos. A Figura 1 ilustra as etapas automatizadas pelo *script*.



**Figura 1. Etapas do script.**

Serão consideradas às seguintes métricas para medir o desempenho das tecnologias: a **média do tempo de execução** - o tempo total de execução que será obtido através da média de 40 execuções; o **desvio padrão** - para avaliar a qualidade das soluções será considerado o desvio padrão obtido para os tempos de execução; **speed-Up** - métrica que será utilizada para expressar quantas vezes a execução do *benchmark* paralelo ficou mais rápido do que a versão sequencial. O cálculo do *speed-up* consiste na razão entre o tempo de execução da aplicação na versão sequencial e a versão paralela; **eficiência** - medida que mostra como foi a taxa de utilização média das unidades de processamento. O cálculo da eficiência consiste na razão entre o *speed-up* e as unidades de processamento utilizadas.

#### 4. Avaliação de desempenho

O *script* será desenvolvido através das linguagens de programação *Bash* e *Python*. A avaliação dos sistemas de virtualização será realizada com o auxílio da suíte de *benchmarks* NPB-MPI. Para a criação dos cenários, onde os experimentos serão realizados, será utilizada a ferramenta *Vagrant*. O código será *open source* e disponibilizado na plataforma de hospedagem de código fonte *GitHub*.

A Figura 2 ilustra a arquitetura do trabalho proposto. A primeira camada representa os componentes de *hardware*, que incluem, CPU, RAM e HD, do servidor físico onde serão realizados os experimentos. Na camada acima é localizado o *kernel*, que será responsável pelo escalonamento dos recursos físicos para os componentes de virtualização. A próxima camada, acima do *kernel*, é onde serão alocados os sistemas de virtualização para a realização dos experimentos. Os resultados obtidos da avaliação do desempenho serão utilizados para gerar os gráficos.

O *script* receberá como entrada um arquivo de configuração. A partir da leitura desse arquivo, será realizada a alocação dos cenários, execução dos *benchmarks* e *parsing* dos resultados. A saída do *script* será o desempenho dos *hypervisors*. Os gráficos esperados para este trabalho são gráficos de desempenho do mesmo *hypervisor* em cenários diferentes, de diferentes *hypervisors* em cenários iguais e de *hypervisors* diferentes em cenários diferentes. Com isso, espera-se identificar discrepâncias entre essas duas tecnologias no ambiente de HPC.

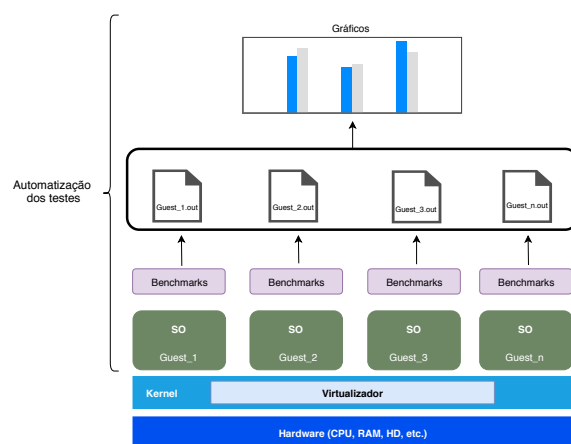


Figura 2. Arquitetura do trabalho proposto.

## 5. Considerações finais

Este trabalho teve como objetivo apresentar uma proposta de avaliação de desempenho automatizada dos virtualizadores VMware ESXi e Linux KVM. Para isso, foi proposto o desenvolvimento de um *script* que será responsável pela automatização dos experimentos. O *script* contribuirá para a reprodutibilidade das etapas de criação e execução de ambientes, cenários, experimentos, coleta de dados e geração de gráficos. E finalmente, identificar qual tecnologia é mais adequada para aplicações científicas.

Outro desafio que considera-se, é a adaptação do *script* desenvolvido para auxiliar trabalhos em outras áreas da computação. A ideia consiste em modularizar o programa, para que estudantes e pesquisadores utilizem o *script* de maneira que contribua para o desenvolvimento de trabalhos. Em síntese, poderia ser considerada a utilização desta ferramenta como um método para auxiliar e facilitar algumas etapas durante o processo de realização de experimentos para trabalhos acadêmicos.

## Referências

- Buyya, R., Vecchiola, C., and Selvi, S. T. (2013). *Mastering cloud computing: foundations and applications programming*. Newnes.
- Hwang, J., Zeng, S., Wu, F., and Wood, T. (2013). A component-based performance comparison of four hypervisors. In *IM*, pages 269–276.
- KVM (2016). Main Page — KVM. [Online; accessed 5-December-2018].
- Maliszewski, A. M., Griebler, D., Schepke, C., Ditter, A., Fey, D., and Fernandes, L. G. (2018). The NAS Benchmark Kernels for Single and Multi-Tenant Cloud Instances with LXC/KVM. In *2018 International Conference on High Performance Computing Simulation (HPCS)*, pages 359–366.
- VMware (2018). VMware vSphere Hypervisor gratuito, Virtualização gratuita (ESXi) . [Online; accessed 5-December-2018].
- Vogel, A., Maron, C. A., Griebler, D., and Schepke, C. (2016). Medindo o desempenho de implantações de openstack, cloudstack e opennebula em aplicações científicas. *16th Escola Regional de Alto Desempenho do Estado do Rio Grande do Sul (ERAD/RS)*, pages 279–282.