A influência das informações de hardware e software nos serviços de IaaS: uma abordagem sobre desempenho de máquinas virtuais em nuvem

Dionísio Machado Leite Filho

SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: _____

Dionisio Machado Leite Filho

A influência das informações de hardware e software nos serviços de IaaS: uma abordagem sobre desempenho de máquinas virtuais em nuvem

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências - Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientadora: Profa. Dra. Regina Helena Carlucci Santana

**USP – São Carlos
Fevereiro de 2016**

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados fornecidos pelo(a) autor(a)

L533i Leite Filho, Dionisio Machado
A influência das informações de hardware e software nos serviços de IaaS: uma abordagem sobre desempenho de máquinas virtuais em nuvem / Dionisio Machado Leite Filho; orientadora Regina Helena Carlucci Santana. -- São Carlos, 2016.
112 p.

Tese (Doutorado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2016.

1. Avaliação de desempenho. 2. Infraestrutura como serviço. 3. Virtualização. I. Santana, Regina Helena Carlucci, orient. II. Título.

Dionisio Machado Leite Filho

**The influence of hardware and software information in the
IaaS services: an approach about performance in virtual
machines in clouds**

Doctoral dissertation submitted to the Instituto de
Ciências Matemáticas e de Computação - ICMC-
USP, in partial fulfillment of the requirements for the
degree of the Doctorate Program in Computer
Science and Computational Mathematics. *FINAL
VERSION*

Concentration Area: Computer Science and
Computational Mathematics

Advisor: Profa. Dra. Regina Helena Carlucci Santana

**USP – São Carlos
February 2016**

*Este trabalho é dedicado às pessoas presentes e ausentes
que contribuem ou contribuíram com suas mais diversas perspectivas
sobre o mundo.*

AGRADECIMENTOS

Agradeço primeiramente aos meus pais que sempre me motivam e incentivam tanto nos momentos felizes como nos momentos tristes. À professora Regina, minha orientadora, por ter aceitado me orientar e ter me guiado nos momentos em que eu necessitava e pela paciência. Agradeço também aos professores que contribuíram com a minha formação, em especial aos professores da graduação.

Aos meus amigos de república que me apoiam tanto nas dificuldades como nas alegrias. Aos amigos de laboratório que me aguentam em momentos inspirados e ainda contribuem para discussões interessantes não só sobre trabalho mas também discussões do cotidiano.

Agradeço ao professor Júlio pelas contribuições sempre bem vindas e pelas contribuições que ainda estão por vir.

Agradeço ao ICMC e principalmente ao LASDPC por proporcionar uma estrutura na qual eu pude crescer tanto como pessoa tanto como profissional.

E por último, mas não menos importante, agradeço à UFMS campus de Ponta Porã pelo apoio para a conclusão dos meus estudos.

*“Todo caminho começa
com um primeiro passo.”
(Lao Tsé)*

RESUMO

LEITE FILHO, D. M.. **A influência das informações de hardware e software nos serviços de IaaS: uma abordagem sobre desempenho de máquinas virtuais em nuvem.** 2015. 112 f. Tese (Doutorado em em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

Esta tese apresenta um conjunto de avaliações experimentais com o objetivo de identificar como as camadas de *hardware* e *software* podem ser determinantes na obtenção de desempenho em máquinas virtuais em nuvem. A computação em nuvem apresenta benefícios no fornecimento de serviços de infraestrutura de forma conveniente e sob demanda. Essa oferta de serviços, sob demanda, pode conduzir os clientes à contratação de serviços virtualizados que atendam às suas necessidades com bom desempenho. Quando os serviços de infraestrutura são vendidos ou alugados com a descrição apenas das características das máquinas virtuais, o desempenho obtido pode não ser o desejado devido às restrições dos recursos físicos. Sendo assim, é proposto um levantamento sobre quais informações são fornecidas pelos provedores de nuvem. De posse das informações coletadas, um conjunto de experimentos demonstra a importância da camada de virtualização para os serviços de infraestrutura. Os resultados apresentados nesta tese demonstram que apesar de os provedores terem tipos de máquinas virtuais semelhantes nos recursos virtuais, os recursos físicos são determinantes na obtenção do desempenho. Além dos recursos físicos, os recursos de *software* também são determinantes na obtenção do desempenho, devido principalmente à marcante influência do virtualizador considerado.

Palavras-chave: Infraestrutura como serviço, Avaliação de Desempenho, Virtualização.

ABSTRACT

LEITE FILHO, D. M.. **A influência das informações de hardware e software nos serviços de IaaS: uma abordagem sobre desempenho de máquinas virtuais em nuvem.** 2015. 112 f. Tese (Doutorado em em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

This thesis presents a set of experimental evaluations in order to identify how the hardware and software layers can be crucial in getting services with good performance levels. Cloud computing offers benefits in providing a convenient way of infrastructure services and on demand. This services' offer, on demand, can drive customers hiring virtualized services that meet their needs with good performance. When the infrastructure services are sold or leased with a description only of the characteristics of virtual machines, the obtained performance may not be desired due to the restrictions of physical resources. Thus, a survey on what information is provided by cloud providers is proposed. Armed with the information collected, a set of experiments demonstrates the importance of the virtualization layer for infrastructure services. The results presented demonstrate that although providers have similar types of virtual machines in virtual resources, physical resources are determinant in obtaining performance. In addition to the physical resources, the resources of software are also critical in getting the performance, mainly due to the strong influence of the considered virtualizer.

Key-words: Infrastructure as a Service, Performance Evaluation, Virtualization.

LISTA DE ILUSTRAÇÕES

Figura 1 – Arquitetura do virtualizador Xen	31
Figura 2 – Arquitetura do virtualizador KVM	32
Figura 3 – Arquitetura do Openvz	33
Figura 4 – Camadas da arquitetura em nuvem	36
Figura 5 – Modelos de classificação de nuvem	37
Figura 6 – Componentes considerados neste projeto de doutorado	61
Figura 7 – Modelo de desenvolvimento utilizado	63
Figura 8 – Tabelas utilizadas para o gerenciamento das informações do protótipo	65
Figura 9 – Tabelas utilizadas no gerenciamento das VMs	66
Figura 10 – Tabelas com as informações de utilização de recursos	67
Figura 11 – Configuração das características dos hospedeiros no protótipo	68
Figura 12 – Definição do limite de recursos disponibilizados por conta	69
Figura 13 – Etapas consideradas na elaboração desta pesquisa	72
Figura 14 – Atividades realizadas no desenvolvimento dos experimentos	74
Figura 15 – Desempenho obtido com a variação de infraestrutura e número de VMs para a carga de trabalho N-queens	77
Figura 16 – Médias de tempos de resposta observadas nos fatores para a carga N-queens	78
Figura 17 – Desempenho obtido com a carga de trabalho Apache	79
Figura 18 – Média de requisições atendidas nas variações entre capacidade de Vm e tipos de provedores	79
Figura 19 – Definição das VMs	81
Figura 20 – Comparação de desempenho para a carga de trabalho N-queens	82
Figura 21 – Desempenho geral obtido pelos fatores para a carga de trabalho N-queens	83
Figura 22 – Desempenho obtido pelas VMs na execução da carga de trabalho Apache	84
Figura 23 – Desempenho geral obtido nas VMs para a carga de trabalho Apache	84
Figura 24 – Desempenho do conjunto de VMs utilizada	86
Figura 25 – Tempo de resposta obtido com o aumento de concorrência no hospedeiro	87
Figura 26 – Requisições por segundo obtidas com o aumento de concorrência no hospedeiro	88
Figura 27 – Carga SmallPt	89
Figura 28 – Carga N-queens	90
Figura 29 – Carga SmallPt	91
Figura 30 – N-queens	91
Figura 31 – Comparação de desempenho entre as VMs.	92

Figura 32 – Influência das cargas de trabalho no desempenho da VM	93
Figura 33 – Desempenho obtido na comparação com cargas de trabalho distintas	94

LISTA DE TABELAS

Tabela 1 – Comparação entre as formas de desenvolvimento de nuvem	42
Tabela 2 – Características a serem consideradas na avaliação de sistemas	47
Tabela 3 – Tipos de avaliação considerando a computação em nuvem	51
Tabela 4 – Trabalhos relacionados à infraestrutura como serviço	56
Tabela 5 – Trabalhos relacionados à avaliação de desempenho utilizando virtualização .	59
Tabela 6 – Características físicas dos hospedeiros	64
Tabela 7 – Informação de infraestrutura fornecida pelos provedores	75
Tabela 8 – Comparação de preço considerando VMs com o mesmo propósito	76
Tabela 9 – Configuração das máquinas virtuais	77
Tabela 10 – Informação da infraestrutura utilizada	80
Tabela 11 – Planejamento de experimentos sobre diferenças de virtualizadores	80
Tabela 12 – Configuração das máquinas virtuais	85
Tabela 13 – Configurações dos experimentos realizados	89
Tabela 14 – Combinação de cargas de trabalho	93
Tabela 15 – Médias e desvios dos resultados obtidos com relação às requisições atendidas	94

LISTA DE ABREVIATURAS E SIGLAS

API		<i>Application Programming Interface</i>
AWS		Amazon Web Services
EC2		<i>Amazon Elastic Compute Cloud</i>
HDD		<i>Hard Disk Drive</i>
HPC		<i>High Performance Computing</i>
IaaS		<i>Infrastructure as a Service</i>
KVM		<i>Kernel-based Virtual Machine</i>
NAS		<i>Network-Attached Storage</i>
NFS		<i>Network File System</i>
NIST		Instituto de Padrões e Tecnologia Norte Americano
P2P		<i>Peer to Peer</i>
PaaS		<i>Platform as a Service</i>
QEMU	...	<i>Quick Emulator</i>
QoS		qualidade de serviço
SaaS		<i>Software as a Service</i>
SAS		<i>Serial Attached SCSI</i>
Sata		<i>Serial Advanced Technology Attachment</i>
SLA		<i>Service Level Agreement</i>
SSD		<i>Solid-State Drive</i>
TI		Tecnologia da Informação
vCPUs	...	CPU virtual
VMs		Máquinas Virtuais
VPN		<i>Virtual Private Network</i>
XCP		Xen Cloud Platform

SUMÁRIO

1	INTRODUÇÃO	23
1.1	Contextualização	23
1.2	Motivação	25
1.3	Hipótese	25
1.4	Objetivos	26
1.5	Metodologia	26
1.6	Organização dos capítulos	27
2	COMPUTAÇÃO EM NUVEM	29
2.1	Contextualização	29
2.2	Virtualização	30
2.2.1	<i>Xen</i>	31
2.2.2	<i>KVM</i>	32
2.2.3	<i>OpenVz</i>	33
2.3	Modelo econômico	34
2.4	Serviços ofertados na computação em nuvem	35
2.5	Classificação das nuvens	37
2.6	Desafios da computação em nuvem	38
2.7	Plataformas para o desenvolvimento de nuvem	40
2.8	Considerações finais	43
3	TRABALHOS RELACIONADOS	45
3.1	Contextualização	45
3.2	Metodologias para avaliação de desempenho	45
3.3	Qualidade de serviço	47
3.4	Infraestrutura como serviço	52
3.5	Virtualização	57
3.6	Considerações finais	59
4	PROTÓTIPO PARA AVALIAÇÃO DE COMPUTAÇÃO EM NUVEM	61
4.1	Contextualização	61
4.2	Desenvolvimento da infraestrutura de nuvem	63
4.2.1	<i>Gerenciamento da infraestrutura</i>	65
4.2.2	<i>Gerenciamento de máquinas virtuais</i>	66

4.3	Considerações finais	70
5	RESULTADOS	71
5.1	Considerações iniciais	71
5.2	Planejamento de experimento	71
5.3	Informações sobre as VMs e desempenho entre provedores de IaaS	75
5.3.1	<i>Informações sobre a contratação de VMs</i>	<i>75</i>
5.3.2	<i>Comparação de desempenho entre os provedores Amazon e Microsoft</i>	<i>76</i>
5.4	Comparação entre virtualizadores	80
5.5	Comparação de desempenho com relação ao aumento de concorrência	85
5.6	Planejamento de capacidade	88
5.7	Alocação de máquinas virtuais com base na carga de trabalho	92
5.8	Considerações finais	95
6	CONCLUSÕES	97
6.1	Considerações iniciais	97
6.2	Conclusões	97
6.3	Contribuições	99
6.4	Publicações	99
6.5	Dificuldades encontradas	100
6.6	Trabalhos futuros	101
	REFERÊNCIAS	103

INTRODUÇÃO

1.1 Contextualização

A computação como utilidade é um sonho antigo que, atualmente, virou realidade com o advento da computação em nuvem ([ARMBRUST *et al.*, 2009](#)). Em meados da década de 60, o cientista Joseph Carl Robnett Licklider sugeriu que a computação fosse utilizada como utilidade pública, assim como os serviços de água e luz ([LAURSEN, 1997](#)). No entanto, devido às limitações tecnológicas da época, o projeto de fornecer computação como serviço não teve continuidade.

Em meados de 1997, a empresa Salesforce disponibilizou alguns serviços empresariais *on-line*, fato esse que pode ser compreendido como o primeiro passo para o surgimento da nuvem ([PARTHASARATHY; SOHI, 1997](#)). Além desse fato, a criação da Web por [Berners-Lee, Fischetti e By-Dertouzos \(2000\)](#) e sua popularização podem ser considerados como fatores essenciais no desenvolvimento do que hoje é conhecido como computação em nuvem ([ARMBRUST *et al.*, 2009](#)).

Com a melhoria na infraestrutura de comunicação, a popularização da Web e a possibilidade de aumentar os rendimentos com a venda de serviços, provedores como o Google, Amazon e Microsoft começaram a ampliar e melhorar o fornecimento de serviços pela Internet ([QIAN *et al.*, 2009](#)).

Dessa forma, a computação em nuvem é um modelo de computação que está revolucionando a história da Tecnologia da Informação (TI) ([VAQUERO *et al.*, 2008](#)) ([ARMBRUST *et al.*, 2009](#)) ([STANTCHEV; SCHRÖPFER, 2009](#)) ([GARG; VERSTEEG; BUYYA, 2012](#)) ([DUBEY; AGRAWAL, 2013](#)) ([WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015](#)). Como os serviços da computação em nuvem são oferecidos por meio da Internet, ela se torna atraente tanto para os clientes como para os provedores. Do lado dos provedores, há a possibilidade de maximização dos lucros; do lado dos clientes, há a possibilidade de alugar serviços de forma conveniente e

apenas quando necessário, aumentando a economia com a compra e manutenção de infraestrutura (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015).

Essa forma de computação faz com que os serviços sejam acessados de uma maneira mais prática do que as formas convencionais, uma vez que as aplicações estão disponíveis na Internet e o acesso às informações e aplicações depende, na maior parte das vezes, apenas de um navegador Web (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015).

Os serviços ofertados pela nuvem podem ser divididos em três categorias principais, como sugere o Instituto de Padrões e Tecnologia Norte Americano (NIST), que são: serviços de *software* ou *Software as a Service* (SaaS); serviços de plataforma ou *Platform as a Service* (PaaS); e infraestrutura como serviço ou *Infrastructure as a Service* (IaaS) (MELL; GRANCE, 2011).

Nessa classificação, o serviço de infraestrutura pode ser considerado como um serviço determinante, uma vez que ele está presente nos demais serviços, por ser o mais próximo à camada de *hardware* (MELL; GRANCE, 2011). Os serviços de infraestrutura são apoiados pelo uso de virtualização, possibilitando aumentar a quantidade de recursos disponíveis, uma vez que a virtualização permite particionar um computador físico em vários computadores lógicos (CHIEU *et al.*, 2009).

Como os serviços da computação em nuvem estão cada vez mais populares, é necessário que haja preocupações relacionadas ao desempenho que os clientes irão obter de seus serviços. Neste contexto, surge a necessidade de fornecer qualidade de serviço (QoS) para os serviços de IaaS (ARMBRUST *et al.*, 2009) (GARG; VERSTEEG; BUYYA, 2012) (WARD; BARKER, 2014) (AL-AYYOUB *et al.*, 2015).

A qualidade de um serviço pode ser definida como a percepção do cliente em relação à eficiência de um dado serviço. Prover QoS significa garantir características, como: disponibilidade, desempenho, agilidade, usabilidade, segurança, garantia e responsabilidade (KUMAR *et al.*, 2013) (DUBEY; AGRAWAL, 2013) (ALHAMAZANI *et al.*, 2014).

Segundo Liu *et al.* (2013), uma das formas de manter a QoS é estabelecer acordos, ou contrato de serviço *Service Level Agreement* (SLA) entre clientes e provedores. Além da SLA, podem-se considerar formas de desenvolvimento da nuvem, alocação de Máquinas Virtuais (VMs) e controle de carga ou tarefas na nuvem (CHANG; WALTERS; WILLS, 2013) (ZAMAN; GROSU, 2013) (BABU; KRISHNA, 2013) (WARD; BARKER, 2014) (AL-AYYOUB *et al.*, 2015) (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015).

Independentemente do contexto envolvido, os tipos de gerenciamento de QoS devem incluir requisitos, tais como: mapeamento de atributos de QoS, estabelecimento de SLA e monitoramento dos recursos virtuais (GARG; VERSTEEG; BUYYA, 2012) (DUBEY; AGRAWAL, 2013) (WARD; BARKER, 2014) (AL-AYYOUB *et al.*, 2015). O monitoramento dos recursos virtuais torna-se importante no ciclo de vida da nuvem, uma vez que os serviços da nuvem

dependem da camada de virtualização.

1.2 Motivação

Atualmente, os serviços de infraestrutura estão em evidência no fornecimento de desempenho para os demais serviços que são hospedados na nuvem (WARD; BARKER, 2014) (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015). Esses serviços são considerados como essenciais, uma vez que são os serviços mais básicos de nuvem (MANVI; Krishna Shyam, 2014). Além disso, os serviços de infraestrutura são apoiados pelo uso de virtualização, permitindo aumentar a utilização dos recursos físicos e a rentabilidade dos provedores (CASALICCHIO; SILVESTRI, 2013) (CORRADI; FANELLI; FOSCHINI, 2014) (AL-AYYOUB *et al.*, 2015) (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015). No entanto, ainda há incertezas sobre o desempenho esperado pelo serviços de IaaS, como apontam Ward e Barker (2014) e Manvi e Krishna Shyam (2014).

Essas incertezas são provocadas devido à natureza de serviços compartilhados da nuvem e ao fornecimento de apenas algumas informações da infraestrutura física por parte dos provedores (WARD; BARKER, 2014). Sendo assim, esta tese tem como foco de pesquisa a avaliação de como os recursos físicos e suas configurações podem influenciar na qualidade dos serviços que são hospedados em máquinas virtuais na nuvem, tendo como objetivo a manutenção ou melhoria do desempenho dessas máquinas.

1.3 Hipótese

Atualmente, há estudos que consideram o monitoramento e a manutenção dos recursos de infraestrutura de nuvem, integração de QoS com as características de virtualização em nuvem, mecanismos de manutenção de SLA e alocação de recursos virtualizados em nuvem (CASALICCHIO; SILVESTRI, 2013) (CORRADI; FANELLI; FOSCHINI, 2014) (AL-AYYOUB *et al.*, 2015) (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015).

Uma característica que deve ser levada em consideração é a relação entre os recursos físicos e os recursos virtuais que são disponibilizados aos clientes. Considerando essas características e considerando que o cliente dos serviços de IaaS tem uma relação mais íntima com questões relacionadas à infraestrutura, pode-se definir que as informações de *hardware* e *software* podem determinar a relação de custo-benefício aos clientes. Essas informações podem acompanhar as definições das VMs contratadas.

Além disso, como a virtualização é a base dos serviços de IaaS, a escolha de mais de um virtualizador ou a presença de uma lista de virtualizadores suportados pode conduzir a uma melhoria de desempenho das aplicações hospedadas nas VMs.

As considerações sobre a relação de recursos físicos e virtuais e o *software* utilizado são por parte do cliente e a contratação de serviços. Por parte dos provedores, a alocação dos serviços pode ser um diferencial no fornecimento de serviços com altos níveis de desempenho. Para esse caso, é possível utilizar duas abordagens: fornecimento de descontos aos clientes caso haja diminuição do desempenho contratado ou a alocação de VMs que consideram os serviços que serão utilizados nessas VMs em relação a outras VMs já alocadas. Em ambas as situações, é necessário considerar o desempenho na elaboração das SLAs.

1.4 Objetivos

O objetivo principal desta tese é demonstrar como as características de *hardware* e *software* podem ser decisivas na seleção de serviços de IaaS na nuvem.

Dentre os objetivos específicos, pode-se destacar:

- Levantamento das metodologias utilizadas na avaliação de sistemas de nuvem e a relação dessa metodologia com o desempenho.
- Levantamento sobre os serviços de infraestrutura de nuvem com a finalidade de identificar quais características são apresentadas e como as mesmas podem ser utilizadas na aquisição de VMs.
- Desenvolvimento de um protótipo com uma ferramenta de nuvem para avaliar características levantadas no estudo entre os provedores de nuvem.

O cumprimento dos objetivos elencados visa validar as hipóteses levantadas e, como consequência, demonstrar que apesar da nuvem ser um ambiente distribuído e com vários recursos disponíveis, esses recursos podem variar de provedor para provedor e fornecer apenas as características das máquinas virtuais pode não conduzir a uma contratação com base em custo-benefício.

1.5 Metodologia

Para a elaboração desta tese foram considerados estudos teóricos, através de trabalhos relacionados à área de nuvem e infraestrutura como serviço. Além de trabalhos relacionados à área, foram necessários estudos experimentais sobre os provedores de nuvem.

Esses dois estudos visaram acompanhar o estado da arte, tanto por parte dos trabalhos acadêmicos como por parte dos trabalhos comerciais.

Além do estudo dos trabalhos relacionados e da realização de estudos experimentais, foram realizados experimentos utilizando um protótipo de nuvem. A utilização do protótipo

permite variar configurações de *software* que não são possíveis utilizando nuvens públicas, além disso, as configurações de *hardware* sempre serão constantes, o que não ocorre entre provedores de nuvem.

1.6 Organização dos capítulos

Os Capítulos presentes nesta tese estão dispostos da seguinte forma:

- No **Capítulo 2** são apresentados os conceitos relacionados à computação em nuvem, bem como conceitos relacionados aos modelos de pagamento de serviços em nuvem e conceitos que envolvem virtualização. Além dos conceitos, são apresentados os desafios e formas de desenvolvimento de nuvem.
- No **Capítulo 3** são apresentados os trabalhos relacionados a esta tese. Os trabalhos relacionados abordam metodologias para avaliação de desempenho, qualidade de serviço, infraestrutura como serviço e virtualização, relacionando o estado da arte ao que foi desenvolvido.
- No **Capítulo 4** é apresentada a metodologia utilizada. Essa metodologia vai desde a construção do protótipo utilizado, até as políticas que foram desenvolvidas.
- No **Capítulo 5** são apresentadas a metodologia para a obtenção dos resultados e os resultados obtidos em análises empíricas e experimentais.
- No **Capítulo 6** são apresentadas as conclusões referentes a esta tese, as publicações, dificuldades encontradas e os trabalhos futuros.

COMPUTAÇÃO EM NUVEM

2.1 Contextualização

A computação em nuvem é assim intitulada por se tratar de um modelo computacional em que os recursos utilizados não estão disponíveis localmente e são acessados via Internet. A nuvem facilita o uso de infraestrutura e plataforma computacional que pode ser dinamicamente configurada e reconfigurada, de acordo com as necessidades dos clientes (MELL; GRANCE, 2011) (ZHANG; YAN; CHEN, 2012) (LIANG *et al.*, 2014).

De acordo com o NIST, a computação em nuvem é um modelo que permite o acesso conveniente a conjuntos de recursos compartilhados (tais como: redes, servidores, armazenamento, aplicações e serviços) que podem ser disponibilizados e liberados com um pequeno esforço de gestão ou de interação com o provedor de serviços (MELL; GRANCE, 2011). Nesse modelo, os clientes acessam serviços (*software*, plataforma e infraestrutura) utilizando, na maior parte das vezes, um navegador Web (ABDELMABOUD *et al.*, 2015).

Dessa forma, pode-se compreender a nuvem como sendo um conjunto de componentes que executam serviços para os clientes sob demanda, onde os clientes possuem acesso apenas aos seus serviços, independente do compartilhamento dos demais recursos disponibilizados na nuvem (WARD; BARKER, 2014) (ABDELMABOUD *et al.*, 2015).

O funcionamento da nuvem se assemelha aos sistemas operacionais de rede, sistemas de tempo real e grades computacionais (KIM; BELOGLAZOV; BUYYA, 2009) (WU *et al.*, 2010), onde os recursos computacionais são fornecidos como um serviço regular. A diferença entre a computação em nuvem e os demais sistemas que a nuvem incorpora não é o objetivo, mas sim as tecnologias que se uniram para realizar a formação da nuvem (LIANG *et al.*, 2014). Os serviços de e-mail gratuito, serviços de busca na Internet, portais de Internet, serviços de hospedagem Web e alguns serviços de infraestrutura computacional são exemplos de serviços oferecidos pela nuvem e que já eram utilizados antes da concepção do termo (KIM; BELOGLAZOV; BUYYA,

2009).

Uma das principais diferenças entre a computação em nuvem e os outros sistemas que entregam serviços é a utilização da virtualização, que se propõe a utilizar de forma maximizada os recursos computacionais disponíveis (CAROLAN *et al.*, 2009) (LI *et al.*, 2011) (LIN; CHEN; LIN, 2014). A virtualização pode ser definida como a execução de vários computadores lógicos em um único computador real. Essa característica faz com que os usuários tenham a impressão de usarem um recurso físico dedicado, quando na verdade podem estar utilizando recursos compartilhados (ALHAMAZANI *et al.*, 2014). Na próxima seção são apresentados os principais conceitos e exemplos de virtualização.

2.2 Virtualização

De uma maneira geral, a computação em nuvem é a migração dos servidores de dentro de uma empresa para a nuvem. O cliente define os requisitos para os recursos a serem alocados, tais como: processamento, armazenamento e largura de banda. O provedor de nuvem, virtualmente, instancia esses recursos dentro da infraestrutura de nuvem (LI *et al.*, 2014) (REHMAN; HUSSAIN; HUSSAIN, 2015).

O modelo utilizado para acesso aos serviços da nuvem permite a contratação de serviços sob demanda, conduzindo a um paradigma para a computação de serviços dinâmicos que são geralmente apoiados por *data centers* (BUYYA; RANJAN; CALHEIROS, 2010). Os *data centers* podem ser considerados como um conjunto de servidores contendo, cada um deles, um conjunto de máquinas virtuais em rede (MANVI; Krishna Shyam, 2014) (LU; CHANG; LI, 2015).

O conjunto de máquinas virtuais em rede, por meio da virtualização, é um diferencial significativo entre a computação em nuvem e os demais sistemas computacionais, viabilizando a melhor utilização de recursos físicos pelos provedores de serviço (MCEVOY; MURY; SCHULZE, 2014) (WEI; BLAKE, 2014).

A virtualização permite ainda que os servidores, dispositivos de armazenamento e *hardware* sejam considerados como um único sistema, ao invés de sistemas separados, permitindo a alocação dinâmica dos recursos pelos servidores (ZHOU; JIANG, 2014). Assim, a virtualização é adotada pela nuvem, pois, oferece vantagens no compartilhamento, gerenciamento e isolamento de recursos físicos. Cada máquina virtual pode ter seu próprio sistema operacional, aplicações e serviços de armazenamento e rede (CHE *et al.*, 2010) (ZHANG; BOUTABA, 2014).

A utilização de virtualização permite que um único servidor físico execute vários outros servidores lógicos que podem ter objetivos diferentes e sistemas diferenciados uns dos outros (KANG; WANG, 2013). Ao permitir essa execução, o uso dos recursos na nuvem se torna mais eficiente, reduzindo os custos operacionais e de gestão da infraestrutura (ZHANG; CHENG; BOUTABA, 2010) (LIN; CHEN; LIN, 2014) (WEINGÄRTNER; BRÄSCHER; WESTPHALL,

2015).

Um dos pontos que deve ser observado quando se deseja utilizar virtualização é qual o propósito do sistema e, conseqüentemente, qual virtualizador deverá ser utilizado. Os virtualizadores são os responsáveis por proverem uma interface para administração e configuração das VMs.

Atualmente, há uma grande quantidade de virtualizadores, como, por exemplo: KVM, Xen, e OpenVz (NADEEM; QAISER, 2015). A computação em nuvem não oferece nenhuma restrição na utilização de virtualizadores (ZHOU; JIANG, 2014).

Quando o objetivo é avaliar sobrecarga de virtualização, os virtualizadores Xen e KVM são os mais utilizados (CHIERICI; VERALDI, 2010). Apesar de esses virtualizadores possuírem propósitos semelhantes, foco no desempenho das máquinas virtuais, eles possuem implementações diferentes (NADEEM; QAISER, 2015).

2.2.1 Xen

O Xen é um virtualizador que suporta a execução de diversos sistemas operacionais convidados, máquinas virtuais, com isolamento de recursos entre as VMs. O Xen é um *software Open Source*, lançado sob os termos da GNU. Qualquer distribuição Linux (Red Hat, SUSE, Debian, Mandrake) pode ser utilizada como hospedeiro ou máquina física hospedeira (YOUNGE *et al.*, 2014) (CARVALHO; BELLEZI, 2014). A Figura 1 apresenta a arquitetura do virtualizador Xen.

Figura 1 – Arquitetura do virtualizador Xen

Fonte: Adaptada de Pascal (2011).

No topo do monitor de máquinas virtuais há dois tipos de máquinas virtuais denominadas “Domínios”, como pode ser observado na Figura 1. O Domínio 0 ou Dom0 é uma máquina virtual que possui privilégios para controlar o monitor de máquinas virtuais, ou seja, todo acesso que

necessite de recursos específicos ou recursos que tenham proteção com relação à concorrência devem passar pelo Dom0. Diferente do Dom0, o Domínio U ou DomU é uma máquina virtual normal, ou seja, é a máquina do cliente que está em funcionamento no hospedeiro (YOUNGE *et al.*, 2014).

Tanto o Dom0 quanto DomU precisam de sistemas operacionais especiais, isso se deve ao fato do Xen ser paravirtualizado. A paravirtualização é uma técnica onde a VM não acessa as funcionalidades do *hardware* em baixo nível (acesso ao kernel), mas possui uma interface de programação (*Application Programming Interface* (API)) padrão para comunicação com o monitor de máquinas virtuais (XENBUS e dispositivos). Já o Dom0 é uma máquina virtual especial, uma vez que ela é quem faz as ligações entre os drivers dos dispositivos e o acesso direto ao kernel. Assim, o Dom0 tem acesso a todos os dispositivos e as DomUs não têm nenhum acesso privilegiado aos dispositivos (CROSBY; BROWN, 2006).

Esse modelo de funcionamento é devido ao tipo de implementação do Xen. Outros modelos adotam a virtualização completa, como é o caso do KVM.

2.2.2 KVM

O *Kernel-based Virtual Machine* (KVM) é um virtualizador do tipo virtualização completa com extensões de virtualização (Intel VT ou AMD-V). Assim, o KVM consiste em um módulo de *kernel* (kvm.ko), que fornece a infraestrutura de virtualização e um módulo de processador específico, intel.ko kvm ou amd.ko kvm. O KVM requer o *Quick Emulator* (QEMU) para a virtualização de sistemas operacionais específicos, como é o caso da Microsoft Windows (KVM, 2011) (CARVALHO; BELLEZI, 2014) (LIU *et al.*, 2014a).

Figura 2 – Arquitetura do virtualizador KVM

Fonte: Adaptada de Gienger (2011).

Com o KVM é possível executar múltiplas máquinas virtuais, em paralelo, sem a ne-

cessidade de modificar as máquinas convidadas. Cada máquina virtual possui um conjunto de *hardware* virtual privado: uma placa de rede, disco, placa gráfica, etc. A visão geral do KVM é apresentada na Figura 2.

Os módulos de virtualização executam em modo kernel e quando a máquina virtual necessita realizar um acesso ao disco, por exemplo, o virtualizador trata essa requisição como uma solicitação de leitura de um programa comum, evitando assim passar por mais uma camada de *software* como ocorre no Xen (KVM, 2011) (LIU *et al.*, 2014a). Além das virtualizações completa e paravirtualizada, também há o modelo de virtualização por contêiner que visa maximizar o desempenho das máquinas virtuais.

2.2.3 OpenVz

Diferente dos virtualizadores Xen e KVM, o virtualizador OpenVz apresenta a virtualização em nível de sistema operacional (CHE *et al.*, 2010) (BABU *et al.*, 2014). Em outras palavras, o virtualizador OpenVz adota o tipo de virtualização em contêiner, ou seja, para realizar a virtualização é necessário que o *kernel* da máquina virtual seja o mesmo do hospedeiro (CHE *et al.*, 2010) (ZHENG; NICOL, 2011) (BABU *et al.*, 2014). A Figura 3 apresenta a arquitetura desse virtualizador.

Figura 3 – Arquitetura do Openvz

Fonte: Elaborada pelo autor.

Como apresentado na Figura 3, as máquinas virtuais são tratadas como ambientes virtuais e compartilham o mesmo ambiente operacional do hospedeiro. Esse isolamento de ambientes

virtuais dentro de um único *Kernel* visa aumentar o desempenho e a escalabilidade das máquinas virtuais (ZHENG; NICOL, 2011).

Para a máquina virtual é como se houvesse toda a árvore de processos, seu próprio sistema de arquivos, usuários, grupos e interfaces de entrada e saída. Os principais desenvolvedores desse virtualizador são a Intel, IBM e Parallels e, diferente dos outros virtualizadores, não há a necessidade de *hardware* específico para realizar a virtualização, ou seja, é possível utilizar máquinas consideradas ultrapassadas para proceder com a virtualização (ZHENG; NICOL, 2011) (BABU *et al.*, 2014).

2.3 Modelo econômico

O pagamento pelos recursos utilizados na computação em nuvem possui uma natureza sob demanda, delimitando os recursos que serão disponibilizados aos clientes (ARMSTRONG; DJEMAME, 2011). O modelo econômico da computação em nuvem permite que as organizações criem ambientes dinâmicos onde recursos podem ser incorporados com base na carga de trabalho e no desempenho necessário para a execução de serviços (MELL; GRANCE, 2011) (KANG; WANG, 2013) (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015).

O cliente que utiliza o modelo de computação em nuvem paga apenas por recursos que são utilizados, o que caracteriza o modelo pague o que utilizar (*Pay-As-You-Go*), onde as garantias são oferecidas pelo provedor de infraestrutura por meio de acordos de nível de serviço personalizados ou SLA (LAMPE *et al.*, 2014). O SLA estipula um contrato de serviços, entre o provedor e o cliente, que especifica as características de como o serviço será prestado (MACÍAS; GUITART, 2014). Ao invés de negociar com uma organização de TI, os recursos de infraestrutura ou plataforma necessários para a implantação de um serviço, na nuvem é proposto o modelo de autoatendimento. Nesse modelo, o cliente compra ciclos de computação e recebe uma interface Web ou uma API para a criação de máquinas virtuais (LIU *et al.*, 2013).

Dessa forma, não é necessário um contrato de longo prazo entre o cliente e os provedores de serviços, as nuvens são pagas por uso e um aplicativo pode ser criado para executar uma tarefa por poucos minutos ou poucas horas. Nas nuvens os aplicativos ou serviços são considerados temporários e o pagamento é baseado no consumo de recursos como: CPU, volumes de dados movidos ou quantidade de dados armazenados (FURHT, 2010) (CASALICCHIO; SILVESTRI, 2013) (LAMPE *et al.*, 2014).

A regra de negócios utilizada pela nuvem transfere os custos de manutenção da infraestrutura que eram bancados pelos clientes para o provedor de serviços em nuvem (CAROLAN *et al.*, 2009). Esta inversão da responsabilidade tem consequências significativas. No passado, os clientes determinavam como os diversos componentes de uma aplicação seriam definidos em um conjunto de servidores. Agora, um cliente pode usar a API de um provedor de nuvem para criar não apenas a composição inicial de uma aplicação, mas também como ela irá evoluir para

acomodar as mudanças de carga de trabalho (LI *et al.*, 2012) (LIU *et al.*, 2014b).

2.4 Serviços ofertados na computação em nuvem

Os serviços na nuvem podem ser considerados como uma arquitetura de camadas onde vários recursos estão disponíveis, formando a nuvem. Essas camadas podem ser consideradas como um conjunto escalável de recursos virtualizados que são capazes de hospedar aplicações e fornecer serviços aos clientes (ZHANG; CHENG; BOUTABA, 2010).

Segundo a definição do NIST (MELL; GRANCE, 2011), para que um serviço seja considerado de nuvem, ele precisa possuir cinco características essenciais, que são:

1. **Autoatendimento sob demanda:** um consumidor pode unilateralmente dispor de capacidades computacionais, como processamento, armazenamento e tempo de rede, conforme necessário, sem a necessidade de interação humana.
2. **Amplo acesso à rede:** recursos são disponibilizados por meio da Internet e podem ser acessados por clientes heterogêneos com qualquer capacidade de processamento (celulares, tablets, notebooks, etc.).
3. **Agrupamento de recursos:** os recursos de computação do provedor são agrupados para atender múltiplos consumidores, com recursos físicos e virtuais (armazenamento, processamento, largura de banda) atribuídos dinamicamente e designados de acordo com a demanda do consumidor.
4. **Elasticidade rápida:** capacidades computacionais podem ser provisionadas e liberadas elasticamente, sendo que, em alguns casos, esta elasticidade é automática, para se ajustar à demanda de cada consumidor. Para o consumidor, as capacidades provisionadas são feitas de forma transparente e podem ser atribuídas a qualquer momento.
5. **Medição de serviço:** recursos da nuvem são controlados e otimizados automaticamente, sendo que o seu uso é medido a partir de um nível de abstração apropriado para cada tipo de serviço (contas de armazenamento, processamento, largura de banda e clientes ativos). A utilização dos recursos pode ser monitorada e descrita em relatórios, o que proporciona transparência tanto para o provedor quanto para o cliente.

No âmbito da computação em nuvem pode-se considerar que diferentes tipos de serviços podem ser oferecidos, tais como: *software*, *hardware*, plataforma, dados, infraestrutura, dentre outros (RIMAL; CHOI; LUMB, 2009). Os serviços podem ser classificados em várias camadas, que podem ser agrupadas em três principais categorias (MELL; GRANCE, 2011), conforme apresentado na Figura 4.

Figura 4 – Camadas da arquitetura em nuvem

Fonte: Adaptada de [Rimal, Choi e Lumb \(2009\)](#).

A primeira camada (**SaaS**) incorpora os serviços que estão hospedados na nuvem. Esses serviços podem variar desde serviços de e-mail até editores de texto. O cliente utiliza o serviço que pode ser pago ou gratuito ([ARMBRUST *et al.*, 2009](#)) ([LIU *et al.*, 2013](#)).

Na segunda camada (**PaaS**), também conhecida como plataforma de desenvolvimento, o cliente interage na criação de aplicações e na publicação das mesmas. A estrutura e as tecnologias a serem utilizadas são definidas pelo provedor, de forma fixa. Cabe ao cliente escolher dentre as opções de plataforma a que melhor se enquadra as suas necessidades ([ARMBRUST *et al.*, 2009](#)) ([ZHANG; CHENG; BOUTABA, 2010](#)) ([ATAS; GUNGOR, 2014](#)).

Por ultimo é apresentada a camada de infraestrutura (**IaaS**), onde o serviço oferecido aos clientes é a infraestrutura virtualizada ([ZHANG; CHENG; BOUTABA, 2010](#)). Nessa camada os clientes contratam a infraestrutura (máquinas virtuais) e armazenamento e a partir deste ponto o cliente tem liberdade para criar sua plataforma bem como para interagir com os serviços ([ZHANG; CHENG; BOUTABA, 2010](#)) ([MANVI; Krishna Shyam, 2014](#)).

Segundo [Han *et al.* \(2014\)](#) e [Pascual *et al.* \(2014\)](#), há três envolvidos nas atividades da nuvem. Esses três envolvidos são: os clientes de serviços, os provedores de serviços e os provedores de infraestrutura.

Os clientes de serviços são os clientes finais, ou seja, os seus serviços podem ser SaaS ou PaaS. Para esses clientes a infraestrutura é transparente ([PASCUAL *et al.*, 2014](#)).

Os provedores de serviço atuam tanto como provedores, fornecendo serviços para os clientes finais, como clientes de infraestrutura ([HAN *et al.*, 2014](#)). Esses clientes possuem acesso aos recursos virtualizados e têm uma visão mais específica dos recursos utilizados. São esses

clientes que definem qual o tipo de máquina virtual será utilizada para o desenvolvimento de um determinado serviço, que pode ser SaaS ou PaaS (HAN *et al.*, 2014).

Os provedores de infraestrutura são, em sua maioria, os donos da infraestrutura física e determinam como os recursos serão disponibilizados aos clientes, quais tecnologias serão utilizadas e qual o preço será cobrado pelo serviço (VM) (HAN *et al.*, 2014) e (PASCUAL *et al.*, 2014). Os serviços de IaaS podem afetar a qualidade geral dos demais serviços por serem a base dos serviços hospedados na nuvem, independentemente do tipo de nuvem que fornece esse serviço.

2.5 Classificação das nuvens

As organizações podem optar por implantar aplicativos em quatro tipos de nuvens: públicas, privadas, híbridas e comunitárias, cada qual com características diferentes (MELL; GRANCE, 2011). As organizações podem fazer uma série de considerações a respeito do modelo ideal de nuvem que atenda aos seus interesses e pode ser adotado mais de um modelo de nuvem para resolver problemas diferentes. A Figura 5 apresenta os tipos de nuvem.

Figura 5 – Modelos de classificação de nuvem

Fonte: Adaptada de Mell e Grance (2011).

As **Nuvens Públicas** são gerenciadas por terceiros. Geralmente quem mantém as nuvens públicas fornece infraestrutura e serviços aos clientes. Este modelo é interessante para clientes que precisam de flexibilidade e serviços temporários para executar suas tarefas, ou programas, reduzindo custos com compra e manutenção de *hardware* (FURHT, 2010). Os serviços prestados pelas nuvens públicas podem ser gratuitos como editores de texto, por exemplo, ou pagos como os serviços oferecidos pela Amazon EC2.

As **Nuvens Privadas** são constituídas para prover recursos específicos para clientes particulares, isto é, a infraestrutura pode ser alugada ou mesmo pertencer ao cliente (FURHT, 2010) (ZHANG; CHENG; BOUTABA, 2010). Neste modelo de nuvem a infraestrutura pode ser modificada de acordo com as necessidades do cliente.

A **Nuvem Híbrida** é uma combinação entre as duas abordagens citadas anteriormente. Nesse modelo pode-se compartilhar temporariamente o uso de recursos das nuvens públicas para garantir o desempenho de todos os serviços, nos momentos em que se apresenta uma alta carga de trabalho na nuvem privada (FURHT, 2010) (MELL; GRANCE, 2011).

A **Nuvem Comunitária** é semelhante à nuvem privada, sendo que nesse caso os recursos da nuvem são compartilhados entre os membros de uma comunidade fechada, que possuem interesses semelhantes. Uma nuvem comunitária pode ser operada por terceiros ou pode ser controlada de forma colaborativa, como no paradigma de computação de grade ou sistemas *Peer to Peer* (P2P) (MARINOS; BRISCOE, 2009).

Os tipos de nuvens não representam necessariamente a localização física da nuvem, por exemplo, as nuvens privadas apesar de representarem um modelo de organização privado, onde os recursos são exclusivos para uma determinada organização, podem estar localizadas fora do espaço físico da organização. Dessa forma, os clientes não precisam saber onde os seus dados estão sendo processados ou armazenados, a ideia é que esses recursos estão na nuvem e podem ser acessados através da Internet (CAROLAN *et al.*, 2009).

2.6 Desafios da computação em nuvem

A computação em nuvem oferece uma série de vantagens em relação a outros paradigmas computacionais, tais como: acesso conveniente, acesso à rede facilitado e pagamento apenas por recursos utilizados. No entanto, ainda há uma série de desafios, que são objetos de pesquisa. Esses desafios são descritos a seguir (FURHT, 2010)(CASALICCHIO; SILVESTRI, 2013)(FATEMA *et al.*, 2014):

- **Desempenho** – O desempenho deve ser considerado de forma geral, ou seja, os serviços, independentemente do tipo (IaaS, PaaS e SaaS), devem ser acompanhados de características que garantam um desempenho adequado. Há muitas soluções oferecidas por provedores que abordam as necessidades de TI de diferentes organizações. Cada solução

oferece diferentes opções em termos de funcionalidade, tempo de resposta e precisão. As organizações devem verificar como as suas aplicações irão se comportar em diferentes provedores e definir qual deles melhor satisfaz suas necessidades;

- **Segurança** – Os provedores de serviços devem fornecer acesso apenas a clientes autorizados e autenticados, garantindo que os dados não serão manipulados por pessoas não autorizadas. A segurança deve ser integrada em todos os aspectos da aplicação e sua implantação na arquitetura das nuvens deve ocorrer em todos os processos;
- **Escalabilidade** – Os serviços projetados para a computação em nuvem precisam ser escaláveis, seguindo as cargas de trabalho que chegam a esses serviços. Para esse fim, os serviços devem permitir que recursos sejam adicionados ou retirados de forma transparente às aplicações que estão em uso, sem afetar a execução dos mesmos;
- **Disponibilidade** – Como o modelo de computação em nuvem segue o modelo de pague o que utilizar, o cliente espera que os serviços contratados estejam permanentemente em funcionamento;
- **Confiabilidade** – Os componentes do sistema devem apresentar uma taxa de falhas reduzida e devem ser substituídos, sem necessidade de interrupções, a qualquer momento. Um ponto relacionado à confiabilidade trata da perda de dados. As aplicações devem operar os dados, mantendo-os intactos, mesmo se houver falhas de um ou mais servidores ou máquinas virtuais no qual estes dados são armazenados ou processados;
- **Flexibilidade e agilidade** – A velocidade na qual as aplicações são entregues é um requisito cada vez mais exigido. A computação em nuvem pode melhorar o processo de criação de serviços, criando servidores de plataforma com serviços mais flexíveis, que suportam um grande número de frameworks de desenvolvimento e várias linguagens de programação;
- **Manutenção** – Na computação em nuvem os componentes de um aplicativo de infraestrutura devem ser atualizados ou mesmo substituídos sem interromper o fornecimento do serviço. Isto é, quando um serviço tem que ser reparado o tempo de inatividade tem que ser mínimo de forma a se tornar transparente para o cliente.
- **Interoperabilidade** – As plataformas normalmente oferecem APIs de acordo com sua infraestrutura. É necessário que haja padrões entre os diversos provedores de serviço tanto com relação à infraestrutura como com relação às plataformas e serviços.

É importante observar que vários pontos ainda não estão bem definidos ou formalizados para a nuvem, podendo ter abordagens diferentes de desempenho, segurança, interoperabilidade, entre outros. A análise mais detalhada desses pontos, de forma conjunta ou separada, pode conduzir a uma melhor definição de como prover qualidade de serviço para a computação em nuvem.

2.7 Plataformas para o desenvolvimento de nuvem

Os principais provedores de serviço no domínio de nuvem incluem: *Amazon Web Services*, *Microsoft Azure* e *Google AppEngine*. Esses provedores oferecem uma variedade de pacotes de serviços de monitoramento, gerenciamento e provisionamento de recursos e serviços (FURHT, 2010) (SAKELLARI; LOUKAS, 2013).

O Amazon Web Services (AWS) oferece três tipos de serviços: *Elastic Load Balancer*, *Auto Scaling* e *CloudWatch*. Esses serviços possuem funcionalidades que são necessárias para a alocação de serviços e recursos do *Amazon Elastic Compute Cloud* (EC2) (TANG *et al.*, 2014).

O serviço *Elastic Load Balancer* dispõe automaticamente a carga dos aplicativos de entrada em várias instâncias do EC2 disponibilizados pela Amazon. Já o serviço de *Auto Scaling* é utilizado para aplicar dinamicamente ou ampliar o número de instâncias do Amazon EC2, para tratar alterações na demanda dos serviços (RANJAN *et al.*, 2010). Finalmente, o serviço *CloudWatch* é integrado com os outros *Webservices* para tomada de decisões estratégicas com base em informações em tempo real dos recursos agregados e informações sobre o desempenho dos serviços (KOKKINOS *et al.*, 2014).

O Windows Azure tem uma estrutura composta de nós formados por servidores com balanceamento de carga. O *Fabric Controller* gerencia o nó por meio de uma base de serviços denominado agente controlador Azure, que monitora o estado do servidor. Se uma falha é encontrada, o gerenciador pode administrar uma reinicialização do servidor ou a migração dos serviços do atual servidor para outros servidores que estejam funcionando normalmente (ZHANG; CHENG; BOUTABA, 2010) (GANNON *et al.*, 2014).

Ao contrário de outras plataformas em nuvem, o *Google App Engine* oferece aos programadores uma plataforma para o desenvolvimento e publicação de serviços. Nesse tipo de plataforma, os clientes ficam limitados ao uso de APIs disponibilizadas pelo Google App Engine. As estratégias de balanceamento de carga, serviço de configuração e dimensionamento são gerenciados automaticamente, sem a interação dos clientes (KRISHNAN; GONZALEZ, 2015).

Segundo Sakellari e Loukas (2013), alguns experimentos podem ser realizados nos provedores citados anteriormente, porém, há outras formas de realizar experimentações com o uso de *testbeds*, como: OpenCirrus, Open Cloud testbed, Science Cloud, Virtual Computing Lab, Future Grid, Grid 5000, Okeanos, entre outros proprietários.

O OpenCirrus é um *testbed* de larga escala distribuído em 10 locais do mundo, ele permite o acesso a recursos virtualizados por meio da Internet (BAUN; KUNZE, 2010).

Grossman *et al.* (2009) apresentam o *Open Cloud testbed* como uma solução para o desenvolvimento de serviços de nuvem. Esse *testbed* está espalhado por quatro regiões dos Estados Unidos e tem suporte a mais de um *data center*.

Keahey *et al.* (2008) definem o *Science Cloud* como sendo um *testbed* distribuído pelos

Estados Unidos e com suporte a serviços da Amazon EC2, permitindo desenvolver serviços de nuvem híbrida provendo a integração entre serviços privados e públicos.

[Shen et al. \(2011\)](#) apresentam o *Virtual Cloud Lab*, uma nuvem privada que permite utilização livre para institutos acadêmicos porém, com limite de tempo.

Além das abordagens de nuvens propriamente ditas, é possível utilizar o *Future Grid*. O *Future Grid* é um *testbed* em grade espalhado pelos Estados Unidos e pode ser utilizado para o desenvolvimento de algumas infraestruturas em nuvem.

[Moussa \(2012\)](#) utilizou o Grid5000 como *testbed* para desenvolver sistemas em nuvem com aplicação em grandes volumes de análise de dados.

Já o Okeanos é uma nuvem apenas de infraestrutura ainda em fase de testes e limitada a universidades e institutos de pesquisa localizados na Grécia ([SAKELLARI; LOUKAS, 2013](#)).

A vantagem dos *testbeds* é a possibilidade de utilização de um grande volume de processamento. No entanto, melhorias no provisionamento da infraestrutura não são possíveis devido à manutenção e a forma de desenvolvimento dos *testbeds*. Um exemplo são os *testbeds* baseados em grades computacionais, que foram adaptados para prover serviços virtualizados.

Para melhorar a experiência no desenvolvimento de aplicações e serviços para nuvem, é possível criar nuvens privadas. Para esses casos, pode não haver uma grande quantidade de computadores envolvidos, no entanto, é possível desenvolver soluções de escalonamento e provisionamento de acordo com as necessidades específicas dos utilizadores ([SAKELLARI; LOUKAS, 2013](#)).

Para esse desenvolvimento é possível utilizar: Eucalyptus, iVIC, OpenNebula, OpenStack, Nimbus, Xen Cloud Platform (XCP) e Cloudstack.

O Eucalyptus é uma ferramenta para o desenvolvimento de infraestrutura de grades que foi adaptado para implementar infraestruturas de nuvem. É uma ferramenta *opensource* que monta a infraestrutura de nuvem de uma forma genérica ([KUMAR; GUPTA, 2014](#)).

O iVIC é uma ferramenta *opensource* voltada para o desenvolvimento de infraestrutura (IaaS) e serviço (SaaS). Trata-se de um ambiente virtual com ênfase em eficiência, energia e segurança ([ZHOU; LI, 2014](#)).

O OpenNebula é uma ferramenta *opensource* para desenvolvimento de infraestrutura idealizada primeiramente para manutenção de *clusters*. Ele gerencia o ambiente de máquinas virtuais e permite a integração de algoritmos personalizados ([KESSACI; MELAB; TALBI, 2014](#)).

[Corradi, Fanelli e Foschini \(2014\)](#), apresentam o Openstack como uma forma de desenvolvimento de infraestrutura para computação em nuvem. É uma ferramenta com a maior comunidade ativa e com a maior aprovação por parte da indústria, sendo um dos seus utilizadores a Canonical (Ubuntu).

O Nimbus é uma ferramenta para desenvolvimento de infraestrutura em nuvem. Essa ferramenta é utilizada como suporte para o desenvolvimento da infraestrutura no Grid5000 e no FutureGrid, facilitando assim a integração dos serviços (TUDORAN *et al.*, 2012).

Nadeem e Kaiser (2015) apresentam o XCP, uma ferramenta opensource baseada no virtualizador Xen para o desenvolvimento de infraestrutura virtualizada. O XCP é uma alternativa ao Xen puro que não possui interfaces de gerenciamento, ou seja, o XCP melhora a forma de interação entre os usuários e o virtualizador Xen.

O Cloudstack é uma ferramenta *opensource* utilizada no desenvolvimento de infraestruturas de nuvem. Essa ferramenta possui suporte a diversos tipos de virtualizados e possui várias funcionalidades para gerência de identidades e gerenciamento de rede (KUMAR *et al.*, 2014).

Analisando as ferramentas citadas anteriormente, é possível notar que a base da computação em nuvem está na infraestrutura. É a partir dela que os demais serviços são gerados e entregues aos clientes (ZHANG; YAN; CHEN, 2012).

Outra forma de desenvolvimento de experimentos para a computação em nuvem é por meio de simulação. Simuladores como o CloudSim (CALHEIROS *et al.*, 2009) (PARIKH *et al.*, 2015), CDOSim (FITTKAU; FREY; HASSELBRING, 2012) (AHMED; SABYASACHI, 2014), TeachCloud (JARARWEH *et al.*, 2012), NetworkCloudSim (GARG; BUYYA, 2011) (AHMED; SABYASACHI, 2014), MdcSim (LIM *et al.*, 2009), entre outros podem ser utilizados para desenvolver soluções de infraestrutura de nuvem e simular ambientes em grandes escala.

Todas as ferramentas descritas nesta seção apresentam vantagens e desvantagens que vão desde a impossibilidade de modificar o ambiente (*testbeds*), limitação na quantidade de recursos (ferramentas para desenvolvimento de nuvens privadas) e abstração de um ambiente com várias camadas (simulação). A Tabela 1 explora esses pontos.

Tabela 1 – Comparação entre as formas de desenvolvimento de nuvem

Soluções	Ambiente	Vantagem	Desvantagem
Amazon Microsoft Azure Google App Engine	De produção (Real)	Resultados que expressam o real funcionamento do ambiente	Ausência de informações técnicas
OpenCirrus, Open Cloud, Science Cloud, Virtual Computing Lab, Future Grid, Grid 5000, Okeanos	<i>Testbed</i>	Maior flexibilidade e informações sobre a infraestrutura	Não permite a manipulação de configurações ou troca de virtualizadores
Eucalyptus, iVIC, OpenNebula, OpenStack, Nimbus, XCP	Protótipo/produção	Amplo acesso aos recursos disponíveis e maior compreensão sobre o funcionamento geral do ambiente	Nem todas as ferramentas possuem integração com virtualizadores e ferramentas de nuvem.
CloudSim, CDOSim, TeachCloud, NetworkCloudSim, MdcSim	Simulado	Praticidade no desenvolvimento de soluções e facilidade na adaptação às mudanças de escopo	Nem sempre a modelagem acompanha o real funcionamento e algumas atividades são complexas de simular.

Fonte: Dados da pesquisa.

Independente das limitações impostas, é necessário definir qual a melhor ferramenta para

o propósito desejado e, como apresentado na Tabela 1, várias ferramentas podem ser utilizadas com o propósito de avaliação de qualidade de serviço na nuvem.

De acordo com a Tabela 1, nota-se a presença de quatro tipos de ambientes, sendo eles: Produção, *Testbed*, Protótipo e Simulado. O ambiente real também pode ser considerado como o de nuvens públicas, onde os clientes podem contratar serviços e desenvolver suas aplicações. Esse ambiente proporciona o desenvolvimento de novos serviços aos clientes finais. Por outro lado, ele insere desvantagens com relação à manipulação de recursos físicos e realocações de recursos.

Já o ambiente de *testbed* proporciona maiores informações sobre a infraestrutura que está sendo disponibilizada. Nesse caso, é possível ter um domínio maior sobre as limitações dos serviços que serão disponibilizados aos clientes. No entanto, algumas configurações não podem ser alteradas e há políticas específicas de uso dos recursos.

O ambiente de protótipo permite a criação não só de protótipos de nuvem como também o desenvolvimento de nuvens privadas. Esse ambiente permite a total configuração e definição dos parâmetros a serem utilizados pela nuvem. No entanto, ele gera um ciclo maior de desenvolvimento, uma vez que todas as políticas de acesso devem ser definidas, políticas de armazenamento, políticas de alocação de máquinas e outras características pertinentes à nuvem.

No ambiente simulado há uma liberdade maior e agilidade no desenvolvimento de soluções para a nuvem. Esse tipo de ferramenta permite desenvolver modelos e testar a hipótese por traz desses modelos, contribuindo de forma direta para o desenvolvimento de novas tecnologias para a nuvem. Entretanto, o modelo deve ser validado de forma correta para não provocar inconsistências no produto final.

De forma geral, o estudo sobre o ambiente real e o ambiente de protótipo se torna viável. Por um lado é possível verificar alguns parâmetros de desempenho por parte dos provedores de nuvem e, a partir do desenvolvimento de um protótipo, contribuir com o desenvolvimento de técnicas voltadas ao gerenciamento dos recursos.

2.8 Considerações finais

A computação em nuvem fornece diversos benefícios como escalabilidade, flexibilidade e a minimização dos custos. Por outro lado, apresenta alguns desafios presentes em todos os tipos de nuvens e camadas de serviços implementados por elas.

Em sua base a computação em nuvem utiliza a virtualização, que permite a criação e gerência de máquinas virtuais de uma forma mais prática para os clientes (sem a necessidade de manipulação da máquina física). Considerações a respeito do uso de virtualização, como a sobrecarga que é gerada sobre a máquina física e se de fato a máquina virtual cumpre com o que foi contratado devem ser observadas.

Como apresentado neste capítulo, são necessárias cinco características para que um serviço seja considerado de nuvem, dentre essas características está a medição de serviço. Como o foco é no desempenho, os aspectos relacionados à medição de serviços serão explorados com relação, principalmente, à forma como as máquinas virtuais são alocadas e mantidas.

A nuvem ainda apresenta vários desafios para a sua adoção definitiva, dentre esses desafios, está o desempenho, que é o principal ponto de interesse desta tese. As questões de desempenho são essenciais no fornecimento de serviços com qualidade, além disso, influenciam diretamente no preço final dos serviços.

Como já descrito, há várias possibilidades de desenvolvimento para explorar as questões relacionadas à medição de serviços e desempenho de nuvem. Considerando o objetivo deste projeto de doutorado, no Capítulo 3 serão discutidos os trabalhos relacionados que apresentam os fundamentos para o desenvolvimento deste projeto. Para validar as hipóteses apresentadas no Capítulo 1, avaliações utilizando sistemas em produção foram realizadas (considerando o Amazon EC2 e o Microsoft Azure) e a utilização de um protótipo de nuvem com a ferramenta Cloudstack foi utilizada para avaliar características relacionadas à virtualização. As características do protótipo serão apresentadas no Capítulo 4 e os resultados obtidos serão apresentados no Capítulo 5.

TRABALHOS RELACIONADOS

3.1 Contextualização

Neste capítulo serão apresentados os trabalhos utilizados como base para o desenvolvimento deste projeto de doutorado. Dentre os conceitos que serão apresentados, pode-se destacar: as metodologias para avaliação de desempenho, a qualidade de serviço, a infraestrutura como serviço e a virtualização.

Os conceitos apresentados neste capítulo serão utilizados na definição da metodologia e na elaboração dos estudos referentes aos serviços de nuvens.

3.2 Metodologias para avaliação de desempenho

Os sistemas computacionais apresentam várias camadas, o que dificulta uma visão analítica do problema. [Skadron *et al.* \(2003\)](#) argumentam que a forma mais utilizada para realizar experimentações é a simulação, por permitir mudanças radicais no escopo da pesquisa. No entanto, como os sistemas estão cada vez mais complexos, a construção de simuladores se torna cada vez mais difícil.

Aferições podem ser utilizadas para contornar o problema da modelagem com solução analítica ou simulação ([SKADRON *et al.*, 2003](#)). No entanto, a aferição em sistemas computacionais não é uma tarefa trivial, uma vez que um elevado número de características pode influenciar nas variáveis de resposta, tornando difícil garantir que o experimento ofereça resultados legítimos e que represente adequadamente o sistema em teste ([GUSTEDT; JEANNOT; QUINSON, 2009](#)). Ainda assim, a aferição se faz necessária nos casos em que se deseja uma análise do sistema como um todo, o que é difícil de ser obtida com técnicas de modelagem.

Segundo [Patil e Lilja \(2012\)](#), no passado a preocupação era em quais tarefas poderiam ser realizadas e em que velocidade. No entanto, com a evolução da computação, o desempenho

envolve problemas multidimensionais que incluem quantas tarefas simultâneas poderão ser executadas e quão eficientes elas serão.

De acordo com [Patil e Lilja \(2012\)](#), quatro tarefas são necessárias para executar de forma coerente a avaliação de um sistema: identificar os objetivos da avaliação, especificar os experimentos, coletar os dados de desempenho obtidos e analisar os dados.

[Li et al. \(2012\)](#) argumentam que a maior parte das avaliações para a nuvem é realizada de forma imprecisa e confusa, como experimentos para avaliar a escalabilidade de aplicações sem especificar como será essa escalabilidade vertical (quando os recursos como CPU e memória são adicionados) ou horizontal (quando a quantidade de máquinas é aumentada), por exemplo. Esse fato pode interferir severamente e dificultar a compreensão dos resultados obtidos.

A avaliação de desempenho dos serviços de nuvem pode seguir diferentes enfoques, como por exemplo, a análise da relação entre custo e benefício na adoção da nuvem, a seleção do prestador de serviços em nuvem, dentre outros. Além disso, o que pode ocorrer são análises imprecisas e análises incorretas, com a utilização de parâmetros não específicos e a falta de definição dos parâmetros que compõe os experimentos ([LI et al., 2012](#)).

Segundo [Chang, Walters e Wills \(2013\)](#), os desafios relacionados à avaliação de desempenho da nuvem, que são:

- Não há um método preciso para analisar o desempenho dos serviços na nuvem. Há o foco em classificações e justificativas do modelo de negócios.
- Portabilidade entre os provedores de nuvem. Esse fato dificulta a utilização dos serviços uma vez que o cliente pode ficar preso a um determinado provedor de serviços de nuvem, impossibilitando avaliar os provedores utilizando os mesmos serviços em todos os provedores avaliados.

Na Tabela 2 é apresentada uma visão geral a respeito dos trabalhos citados anteriormente.

Tabela 2 – Características a serem consideradas na avaliação de sistemas

Trabalho	Avaliação	Técnica de Avaliação	Sistema
Skadron <i>et al.</i> (2003)	Apresentam uma revisão de métodos para avaliação de sistemas computacionais	Simulação e aferição	Qualquer sistema
Gustedt, Jeannot e Quinson (2009)	Apresentam quais são as características para uma boa avaliação	Simulação e aferição	Computação em grade
Patil e Lilja (2012)	Apresentam características para se obter uma boa avaliação	Simulação	Qualquer sistema
Li <i>et al.</i> (2012)	Apresentam as dificuldades na avaliação de sistemas em nuvem	Aferição	Computação em nuvem
Chang, Walters e Wills (2013)	Apresentam os desafios na avaliação qualitativa e quantitativa em nuvem	Não há	Computação em nuvem

Fonte: Dados da pesquisa.

Como observado na Tabela 2, é possível utilizar tanto a simulação quanto a aferição como metodologias para a avaliação e desenvolvimento de novos serviços de nuvem. É importante considerar que a escolha da técnica de avaliação depende de cada caso e da disponibilidade de cada sistema. Para o uso de aferição é necessário ter-se um sistema disponível, independente se de produção (real), *testbed* ou protótipo. Nesse caso, cada tipo de sistema apresenta suas vantagens e desvantagens com relação à manipulação dos fatores a serem considerados.

No caso de não haver o sistema (real, *testbed* ou protótipo) é possível utilizar simulação, inclusive, pela facilidade de modificação do escopo do projeto. Também é possível trabalhar com as duas técnicas de forma conjunta, como é o caso de geração de *logs* nos sistemas, por meio de aferição, e a utilização desses logs nos simuladores para o desenvolvimento de soluções mais próximas do comportamento real do sistema.

Considerando a proposta deste doutorado, a metodologia a ser utilizada para a avaliação do sistema de nuvem será a aferição por meio de protótipo. Além disso, é necessário definir o que é QoS, quais os atributos podem ser levados em consideração e como o conceito pode ser aplicado na nuvem.

3.3 Qualidade de serviço

Prover Qualidade de Serviço (QoS) não é uma tarefa trivial, uma vez que o ambiente de nuvem é afetado pela dinamicidade e desafios da Internet. Em geral, os atributos de QoS da nuvem podem ser verificados considerando-se duas abordagens complementares e que possuem características diferentes. A primeira abordagem é a QoS em nível de serviços relacionados à rede. A QoS de rede pode ser entendida como um conjunto de requisitos a serem cumpridos pela rede para a entrega de um serviço com qualidade a seus clientes (CRAWLEY *et al.*, 1998)(HAN *et al.*, 2013). Os atributos a serem considerados nesse caso incluem: largura de banda, *delay*, *jitter* e perda de pacotes (LINNOLAHTI, 2004) (HAN *et al.*, 2013). Dentre as preocupações relacionadas à QoS de rede estão: como a conexão é abordada, qual métrica é utilizada (quantidade de saltos

ou atraso da rede) e qual forma de roteamento é utilizada (endereçamento por origem e destino ou por fluxo) (LINNOLAHTI, 2004).

A segunda abordagem para se verificar a QoS da nuvem é em nível de aplicação. Para esse caso, o problema é como determinar se o provedor de serviço cumpre com o nível de QoS esperado pelo cliente. Para resolver esse problema, é proposto um estabelecimento de acordos, como um SLA entre os clientes e os provedores (CHARD, 2011).

Garg, Versteeg e Buyya (2012) definem uma série de serviços e como obter os níveis de QoS apropriados. Os autores avaliam os serviços oferecidos, do ponto de vista do cliente, e se os mesmos cumprem com os requisitos de QoS.

Segundo Garg, Versteeg e Buyya (2012), não há métricas ou métodos padronizados que permitam comparar os provedores de nuvem de forma coerente. Os principais atributos a serem considerados podem ser:

- **Responsabilidade** - Um fator importante para que um cliente implante seus aplicativos e armazene seus dados em um lugar é a existência de prestação de contas sobre segurança e conformidade. Atributos essenciais à responsabilidade são: capacidade de auditoria, observância, propriedade dos dados e sustentabilidade.
- **Agilidade** - Um cliente pode expandir ou mudar rapidamente seus objetivos de negócio. Ao considerar a agilidade de um serviço na nuvem, as organizações devem promover: elasticidade, portabilidade, adaptabilidade e flexibilidade.
- **Custo** - Principal atributo na seleção de serviços de nuvem. É importante considerar as características essenciais para a realização de um serviço e sua relação com as expectativas do cliente.
- **Desempenho** - Os clientes precisam entender como as suas aplicações são executadas nos diferentes tipos de nuvens e se essas implantações irão satisfazer as suas expectativas.
- **Garantia** - Cada cliente pode expandir seus negócios, oferecendo serviços melhores. Portanto, a confiabilidade, resiliência e estabilidade do serviço são fatores importantes na escolha de serviços em nuvem.
- **Segurança e Privacidade** - Estes são atributos multidimensionais em sua natureza e incluem muitos atributos como proteger a confidencialidade e a privacidade, integridade e disponibilidade dos dados.
- **Usabilidade** - A usabilidade de um serviço em nuvem pode depender de vários fatores, tais como a acessibilidade, facilidade de implementação dos serviços, operabilidade entre outros relacionados.

O levantamento dos atributos de QoS, considerando especificamente a nuvem, é necessário quando o objetivo é a avaliação e a determinação de como esses atributos devem ser fornecidos aos clientes. Cada atributo pode ser acompanhado de um conjunto de métricas, como: tempo de resposta e utilização, para o desempenho; integridade e privacidade, para a segurança; entre outras métricas relacionadas à atributos específicos.

O objetivo é determinar quais são os atributos que podem ser utilizados para melhorar a forma como a QoS é fornecida aos clientes. Sendo assim, os trabalhos apresentados abaixo exploram os atributos ou métricas de QoS utilizados em serviços de nuvem, com foco nos serviços de infraestrutura.

Segundo [Armstrong e Djemame \(2011\)](#), como a virtualização é parte fundamental no provisionamento de serviços IaaS, manter bons níveis de desempenho nas VMs pode aumentar a rentabilidade da nuvem. Para validar seu trabalho, dois middlewares de nuvem, um com suporte ao KVM e o outro com suporte ao Xen, foram avaliados utilizando benchmarks como cargas de trabalho sintéticas. O atributo de QoS analisado foi o desempenho obtido na propagação de imagens entre os *middlewares* considerados.

[Ramezani, Lu e Hussain \(2013\)](#) propõe um sistema de apoio à decisão para a gestão de recursos em ambientes de nuvem. Para os autores, quanto mais autônomo e inteligente forem os sistemas de manutenção, melhor será a manutenção de recursos. Os atributos de QoS utilizados pelos autores foram: desempenho, custo e garantia.

Uma abordagem de balanceamento de carga para a gestão de QoS em VMs de nuvem é apresentada por [Deye, Slimani e Sene \(2013\)](#). Os autores discutem sobre a não garantia na manutenção da QoS em tempo de execução dos recursos. Segundo [Deye, Slimani e Sene \(2013\)](#), o conceito de nuvem é muito útil, no entanto, sua adoção ainda encontra algumas dificuldades, entre elas está a manutenção da QoS. No modelo atual de nuvem os clientes são cobrados com base nos recursos utilizados ou reservados e nenhuma garantia de QoS é dada em tempo de execução ([DEYE; SLIMANI; SENE, 2013](#)). Os autores utilizam como atributo de QoS o desempenho observado nas VMs.

[Kumar et al. \(2013\)](#) apresentam uma abordagem para obtenção de QoS baseada em alocação de recursos e escalonamento adaptativo. Segundo os autores, com a alocação eficiente de recursos foi possível melhorar os tempos de resposta observados na nuvem. De acordo com [Kumar et al. \(2013\)](#), a métrica mais comum, que influencia nos SLAs, é o tempo de resposta de um trabalho em execução. O atributo de QoS observado foi o desempenho.

Há duas tendências para computação em grandes volumes: computação de alto desempenho e computação com alta disponibilidade ([DONG; NAN; XU, 2013](#)). A computação em nuvem pode ser utilizada para ambas as tendências. No entanto, a QoS em nuvem nem sempre considera a capacidade de completar tarefas ([DONG; NAN; XU, 2013](#)). Segundo os autores, é necessário considerar a probabilidade que um sistema tem de completar uma tarefa sobre

uma predeterminada condição. Esse fato se deve à natureza heterogênea de algumas nuvens que possuem diferentes tipos de componentes e atributos. Segundo [Dong, Nan e Xu \(2013\)](#), os atributos que devem ser melhor explorados são o desempenho, custo e confiança.

[Macías e Guitart \(2014\)](#) apresentam uma negociação de SLA, execução de políticas de maximização de receitas e classificação de clientes em provedores de nuvem. Segundo os autores, o principal objetivo dos provedores de nuvem é maximizar seus lucros. A classificação realizada pelos autores considera o tipo de pagamento realizado pelo cliente e dependendo do tipo de conta, o cliente possui mais ou menos recursos.

Uma pesquisa sobre manutenção de cargas de trabalho e escalonamento é apresentada por [Liu et al. \(2014b\)](#). De acordo com os autores, os objetivos de escalonar uma carga de trabalho são diferentes dependendo da aplicação. Alguns algoritmos são para otimizar a relação entre o custo e o tempo de resposta enquanto que outros podem considerar confiança, disponibilidade, eficiência energética e otimização na alocação de recursos.

De acordo com [Kumar et al. \(2014\)](#), a computação em nuvem está aumentando sua popularidade devido às promessas de redução de custos com infraestrutura e a oferta de elasticidade em demanda. O objetivo desse estudo foi minimizar as sobrecargas com comunicação, diminuição na quantidade de recursos utilizados e redução de energia. Os atributos de QoS utilizados foram: o desempenho, para os clientes e redução de custos, para os provedores.

Segundo [Han et al. \(2014\)](#), a maioria das técnicas de escalabilidade podem manter a QoS para as aplicações mas não lidam de forma eficiente com a redução de custos para os clientes, o que torna essas técnicas caras. Segundo os autores, em um ambiente computacional que é fornecido como utilidade, é necessário que os custos sejam os mais baixos possíveis para o desempenho desejado para as aplicações. A métrica de QoS utilizada pelos autores foi o tempo de resposta visando o atributo custo.

[Kourtesis et al. \(2014\)](#) apresentam um trabalho sobre manutenção de QoS em nuvem baseado em semântica. Segundo os autores, as questões ainda abertas na computação em nuvem são: cumprimento de SLA, predição de cargas de trabalho, processamento de grandes volumes de dados, tomada de decisão em tempo real, adaptação dinâmica e provisionamento de recursos. Em seu trabalho, os autores consideram os três tipos de serviço de nuvem (SaaS, PaaS e IaaS). Os atributos de QoS utilizados para IaaS foram o custo, desempenho e economia de energia.

[Bruneo \(2014\)](#) apresenta um modelo estocástico para investigar o desempenho de data centers e a QoS em serviços de IaaS. Segundo os autores, a investigação sobre a manutenção de infraestrutura de nuvem é importante devido à grande quantidade e heterogeneidade de estratégias que podem ser aplicadas. Segundo autores, várias métricas podem ser utilizadas na infraestrutura de nuvem como a utilização, disponibilidade, tempo de espera e tempo de resposta.

[Jula, Sundararajan e Othman \(2014\)](#) realizaram uma pesquisa sobre composição de serviços de nuvem e parâmetros de QoS. Foram levantados 14 diferentes parâmetros de QoS,

sendo: custo, tempo de resposta, confiança, disponibilidade, reputação, durabilidade, vazão, energia, responsabilidade, manutenibilidade, similaridade de função, latência, controle de dados e segurança. Segundo autores, os atributos mais recorrentes foram o custo e o desempenho (representado pelos tempos de resposta). Os autores argumentam que a relação entre custo e desempenho ainda é uma questão aberta.

Abdelmaboud *et al.* (2015) apresentam uma revisão sistemática sobre os assuntos relacionados à QoS na nuvem. Nesse estudo os autores exploram os três tipos de serviços de nuvem (PaaS, SaaS e IaaS) e, para os serviços de IaaS, foi identificado que os principais atributos de QoS são o custo e o desempenho.

Para Rehman, Hussain e Hussain (2015) é preciso considerar outros atributos de QoS, além do desempenho. Para os autores, é necessário que haja preocupações com a interoperabilidade e usabilidade dos recursos. De acordo com os autores, essa característica pode facilitar a escolha entre os vários provedores de serviços, utilizando outros atributos além do custo para a tomada de decisão.

Na Tabela 3 é apresentada uma visão geral sobre os trabalhos discutidos e que consideram a avaliação de desempenho de recursos providos pela nuvem.

Tabela 3 – Tipos de avaliação considerando a computação em nuvem

Trabalho	Tipo de serviço	Foco	Atributos de QoS
Armstrong e Djemame (2011)	IaaS	Desempenho na propagação de imagens e Entrada e saída paravirtualizada	Desempenho
Ramezani, Lu e Hussain (2013)	IaaS	Manutenção de recursos virtualizados	Desempenho, Custo e garantia
Deye, Slimani e Sene (2013)	IaaS	Manutenção e controle de cargas para aplicações multi instância	Desempenho
Kumar <i>et al.</i> (2013)	IaaS	Alocação de recursos e escalonamento adaptativo	Desempenho
Dong, Nan e Xu (2013)	IaaS	Modelo de QoS baseado em disponibilidade de recursos	Disponibilidade
Macías e Guitart (2014)	-	Políticas de maximização de receitas e classificação de clientes	Custo
Liu <i>et al.</i> (2014b)	IaaS	Manutenção de cargas de trabalho	Desempenho e Custo
Kumar <i>et al.</i> (2014)	IaaS	Sistemas de manutenção que considerem a alocação dos dados e a seleção de réplicas	Desempenho Custo
Han <i>et al.</i> (2014)	SaaS, PaaS IaaS	Custo e elasticidade adaptativa para aplicações multi-tier	Desempenho Custo
Kourtesis <i>et al.</i> (2014)	SaaS, PaaS IaaS	Manutenção de QoS em nuvem baseado em semântica	Desempenho, Custo, Energia
Bruneo (2014)	IaaS	Modelo estocástico para investigar o desempenho de <i>data centers</i>	Disponibilidade Desempenho
Jula, Sundararajan e Othman (2014)	SaaS, PaaS IaaS	Composição de serviços de nuvem e parâmetros de QoS	Desempenho Custo
Abdelmaboud <i>et al.</i> (2015)	SaaS, PaaS, IaaS	Revisão sistemática sobre QoS em nuvem	Desempenho Custo
Rehman, Hussain e Hussain (2015)	IaaS	Manutenção de serviços com foco no cliente	Interoperabilidade e Usabilidade

Fonte: Dados da pesquisa.

Na Tabela 3 são sumarizados os estudos relacionados ao tema de QoS em nuvem. É possível notar que a maioria se preocupa com o desempenho. O desempenho abordado em cada

estudo pode ter métricas diferentes, como: tempo de resposta, interferência da elasticidade no *throughput* do sistema e tempo de escrita e leitura. Esse fato demonstra que há a necessidade de oferecer um bom desempenho, independente do propósito do sistema. Outro ponto de destaque é o custo. Como o modelo econômico da nuvem considera o aluguel, é necessário que os serviços alugados satisfaçam as expectativas dos clientes.

Outros atributos tão importantes quanto o custo e o desempenho também são abordados, como é o caso da garantia, disponibilidade, usabilidade e interoperabilidade. Todos esses atributos podem ser abordados nos três tipos de serviços (SaaS, PaaS e IaaS). Como o foco deste doutorado é em IaaS, o atributo a ser considerado será o desempenho. Um estudo sobre os serviços de IaaS se torna necessário para definir como o desempenho será considerado.

3.4 Infraestrutura como serviço

A infraestrutura como serviço ou IaaS, refere-se aos serviços mais básicos e que possuem a menor transparência aos clientes (LU; CHANG; LI, 2015) (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015). Esses serviços normalmente referem-se às VMs. Nessas máquinas é possível implementar qualquer tipo de serviço, desde simples páginas Web até frameworks de desenvolvimento (WEINGÄRTNER; BRÄSCHER; WESTPHALL, 2015).

De acordo com (ZHANG; CHENG; BOUTABA, 2010), na camada de IaaS os clientes contratam a infraestrutura (VM) e armazenamento e a partir desse ponto o cliente tem liberdade para criar sua plataforma bem como interagir com os serviços. Dessa forma, verificar como o desempenho é mensurado e aplicado se torna necessário para prover serviços com qualidade e desempenho.

Salah *et al.* (2011) apresentam uma avaliação sobre os provedores Amazon, ElasticHosts e Bluelock. No estudo foram feitas comparações entre desempenho de CPU, memória e disco. *Benchmarks* foram utilizados para avaliar os recursos. As avaliações foram realizadas de forma quantitativa e não apresentaram métodos estatísticos de comparação.

Uma abordagem integrada para o gerenciamento automático de recursos virtualizados em ambientes de nuvem é apresentada por (LI *et al.*, 2011). De acordo com os autores, um problema a ser considerado é como oferecer manutenção de recursos minimizando operações de *hardware* e custos operacionais. Outro ponto é como alocar as VMs em ambientes de nuvem. Utilizando o virtualizador KVM os autores demonstram a utilização de um sistema realimentado que melhora o desempenho das VMs se baseando na alocação das mesmas.

Em Navaux *et al.* (2012), os provedores Amazon, Azure e Rackspace são avaliados. Os autores comparam as configurações disponibilizadas pelos provedores. Navaux *et al.* (2012) também investigaram o desempenho e preço cobrado por cada instância virtualizada.

Casalicchio e Silvestri (2013) apresentam um mecanismo para provisionamento de SLA

em provedores de serviço baseados em nuvem. Segundo os autores, os provedores de nuvem têm preocupações em otimizar a utilização dos recursos e reduzir os custos. Já o cliente tem preocupações relacionadas ao custo pago pelos recursos virtualizados. Para a avaliação da proposta, [Casalicchio e Silvestri \(2013\)](#) implementam um testbed utilizando os serviços EC2 da Amazon e demonstram políticas que podem ser utilizadas para a diminuição dos custos.

A alocação de VMs é estudada por [Amarante et al. \(2013\)](#). Os autores modelam o problema como sendo um problema de otimização do tipo *NP-hard* e propoem a utilização do problema da mochila para melhorar a alocação de VMs. Por meio de simulação, os autores demonstram que o algoritmo proposto pode ser utilizado para economizar energia sem prejudicar o desempenho das máquinas virtuais.

[Lin, Chen e Lin \(2014\)](#) apresentam em seu trabalho uma forma de integrar QoS com a virtualização em sistemas de computação em nuvem para aplicações sensíveis a atraso. Segundo os autores, se duas ou mais VMs são alocadas no mesmo servidor físico pode haver certo grau de interferência entre as VMs devido ao compartilhamento de recursos não divisíveis, como dispositivos de entrada e saída. Os recursos não divisíveis são: disco, rede e cache L2 ([LIN; CHEN; LIN, 2014](#)). Simulações foram utilizadas, considerando como parâmetro as configurações do virtualizador Xen.

[Sun et al. \(2013\)](#) apresentam um algoritmo para alocação de VMs baseado em transformação de matriz. Segundo os autores, a alocação de VMs é um problema a ser solucionado para a manutenção e utilização de recursos em nuvem. [Sun et al. \(2013\)](#) demonstram os algoritmos utilizados e as classes dos mesmos. Os experimentos foram realizados em uma infraestrutura de nuvem própria. Os autores não descrevem qual tipo de recurso é disponibilizado pelo data center apenas a quantidade dos recursos.

Uma abordagem para alocação de recursos com diferentes traços de desempenho é apresentada por [Kang e Wang \(2013\)](#). Entre os diferentes tipos de traços de desempenho, podem-se destacar as cargas de entrada e saída intensivas e as cargas orientadas à CPU ([KANG; WANG, 2013](#)). Por meio de simulação os autores demonstram a eficiência do algoritmo proposto.

Segundo [Li et al. \(2014\)](#), devido à heterogeneidade dos diversos tipos de serviços que podem ser executados na nuvem, diferentes VMs no mesmo hospedeiro podem apresentar diferentes tempos de término de serviço. Para os autores, esse tipo de problema, alocar VMs com cargas de trabalho diferentes, pode ser modelado como o problema da mochila ([LI et al., 2014](#)). Simulações foram utilizadas para demonstrar a eficiência da solução proposta. Segundo os autores, a alocação de VMs é um problema ainda aberto devido aos vários atributos que podem ser considerados, incluindo desempenho, disponibilidade, rede e custo.

[Zhang e Boutaba \(2014\)](#) demonstram que a manutenção efetiva de cargas de trabalho em computação em nuvem continua desafiadora. Os autores apresentam a relação entre heterogeneidade de recursos e qualidade de serviço uma vez que a principal dificuldade é determinar onde os

serviços serão alocados minimizando o custo e garantindo os tempos de resposta. Nesse trabalho é apresentada uma forma de manutenção dinâmica em ambientes de nuvens heterogêneas.

[Alhamazani et al. \(2014\)](#) apresentam uma visão geral sobre as ferramentas de monitoramento de nuvem. Segundo os autores, monitorar a qualidade da VM entre os provedores de nuvem é uma atividade complexa. A QoS contratada para as VMs precisa ser constantemente monitorada para garantir e evitar anomalias no sistema. [Alhamazani et al. \(2014\)](#) consideram que o monitoramento de nuvem pode assumir duas formas: alto nível e baixo nível. O alto nível se refere ao estado da plataforma de virtualização. Já o baixo nível é relacionado ao estado real dos componentes de *hardware* como a velocidade e utilização da CPU, vazão de disco, tempo de aquisição e liberação das VMs.

Uma pesquisa sobre manutenção de recursos para serviços de IaaS é apresentada por [Manvi e Krishna Shyam \(2014\)](#). Para os autores a manutenção dos recursos de infraestrutura é um grande desafio para a computação em nuvem é relacionado à alocação, provisionamento, adaptação, descoberta e modelagem. O atendimento dessas necessidades pode conduzir os serviços de IaaS a vários benefícios como: escalabilidade, utilização ótima, redução de sobrecargas, melhora na vazão, redução de latência, efetividade de custo e interfaces simplificadas.

Segundo [Manvi e Krishna Shyam \(2014\)](#), o principal problema encontrado nos serviços de IaaS é fornecer acesso aos recursos compartilhados sem revelar detalhes de localização e *hardware* para os clientes. Para os autores, as questões relacionadas aos serviços de IaaS se referem à manutenção de recursos, manutenção da infraestrutura de rede, virtualização e multicamadas, manutenção dos dados e interoperabilidade.

[Cerotti et al. \(2014\)](#) apresentam um estudo sobre cargas de trabalho visando consolidação e replicação. Segundo os autores, um dos problemas relacionados à virtualização é que o cliente tem pouco ou nenhum controle sobre o *hardware* alocado para a sua VM. Esse problema adicionado à heterogeneidade das cargas de trabalho e a natureza não determinística das taxas de chegada de requisições fazem com que a atividade de encontrar o desempenho desejado se torne custosa ([CEROTTI et al., 2014](#)). Por meio de modelagem analítica os autores demonstram a eficiência da proposta mostrando ganhos de desempenho quando se tem controle de determinados parâmetros da nuvem.

Um estudo sobre desempenho e controle energético em servidores virtualizados é apresentado por [Zhou e Jiang \(2014\)](#). Segundo os autores, devido à grande dinamicidade das cargas de trabalho, a complexidade das aplicações e a complexidade das infraestruturas compartilhadas faz com que os *data centers* encarem significantes desafios no mapeamento de desempenho, mantendo a eficiência na utilização e reduzindo os custos com energia.

Ferramentas de monitoramento de nuvem são apresentadas por [Fatema et al. \(2014\)](#). Segundo [Fatema et al. \(2014\)](#), as técnicas de monitoramento são indispensáveis na manutenção de nuvens em larga escala e no fornecimento de QoS para os clientes. De acordo com os autores,

as áreas operacionais da nuvem e que podem ser monitoradas incluem: contabilidade e cobrança, manutenção de SLA, provisionamento de recursos, planejamento de capacidade, manutenção de configuração, segurança e manutenção de falhas.

McEvoy, Mury e Schulze (2014) apresentam uma definição e alocação de VMs para aplicações de alto desempenho na nuvem. Segundo os autores, uma forma de aumentar o desempenho das cargas de trabalho do tipo *High Performance Computing* (HPC) em ambientes de nuvem é realizando um mapeamento de recursos físicos para os processos de aplicação que estão executando na VM.

Wei e Blake (2014) introduzem uma forma proativa de manutenção de recursos virtualizados para cargas de trabalho de nuvem. Para os autores a preocupação com a manutenção de recursos pode ser determinante para o sucesso da nuvem uma vez que o acesso conveniente a recursos cria a ilusão de recursos infinitos. No entanto, como os serviços são pagos pela sua utilização, é inviável alocar um serviço usando quantidades de recursos absurdas sem considerar o custo, por exemplo.

Uma pesquisa sobre manutenção de recursos em nuvem é apresentado por Weingärtner, Bräscher e Westphall (2015). Segundo os autores, o uso de modelos e ferramentas para criar perfis de aplicações é essencial para a criação de modelos de previsão, o que pode melhorar a forma como os recursos são alocados para os diferentes tipos de cargas de trabalho. Segundo Weingärtner, Bräscher e Westphall (2015), as necessidades de perfis podem ser aplicados à manutenção de recursos, manutenção de aplicações e manutenção dos custos.

Os trabalhos discutidos nesta seção são dispostos, de forma resumida, na Tabela 4.

Tabela 4 – Trabalhos relacionados à infraestrutura como serviço

Trabalho	Objeto de estudo	Tipo de avaliação
Salah <i>et al.</i> (2011)	Avaliação entre Provedores	Quantitativa
Li <i>et al.</i> (2011)	Gerência de recursos virtualizados	Quantitativa
Navaux <i>et al.</i> (2012)	Avaliação entre provedores	Qualitativa e Quantitativa
Casalicchio e Silvestri (2013)	Manutenção de SLA	Quantitativa
Amarante <i>et al.</i> (2013)	Alocação de máquinas virtuais	Quantitativa
Lin, Chen e Lin (2014)	Integração de QoS com virtualização	Quantitativa
Sun <i>et al.</i> (2013)	Alocação de máquinas virtuais	Quantitativa
Kang e Wang (2013)	Alocação de recursos físicos	Quantitativa
Li <i>et al.</i> (2014)	Alocação de máquinas virtuais considerando diferentes cargas de trabalho	Quantitativa
Zhang e Boutaba (2014)	Manutenção de cargas de trabalho	Quantitativa
Alhamazani <i>et al.</i> (2014)	Ferramentas de monitoramento de nuvem	Qualitativa
Manvi e Krishna Shyam (2014)	Manutenção de recursos de infraestrutura	Qualitativa
Cerotti <i>et al.</i> (2014)	Descoberta de desempenho para consolidação de cargas de trabalho	Quantitativa
Zhou e Jiang (2014)	Análise de desempenho e consumo energético	Qualitativa Quantitativa
Fatema <i>et al.</i> (2014)	Ferramentas de monitoramento	Qualitativa
McEvoy, Mury e Schulze (2014)	Alocação de máquinas virtuais para aplicações de alto desempenho	Quantitativa
Wei e Blake (2014)	Manutenção proativa de recursos virtualizados	Quantitativa
Weingärtner, Bräscher e Westphall (2015)	Manutenção de recursos de infraestrutura	Qualitativa

Fonte: Dados da pesquisa.

Como apresentado na Tabela 4, vários trabalhos abordam os serviços de IaaS tanto de forma quantitativa como de forma qualitativa. As avaliações qualitativas visam identificar quais são as qualidades ou funcionalidades fornecidas aos clientes. Essas funcionalidades vão desde o oferecimento de interfaces amigáveis até o fornecimento de informações sobre a infraestrutura dos provedores.

As avaliações quantitativas são, em sua maioria, relacionadas às VMs. O estudo sobre o comportamento da infraestrutura é importante em várias áreas, como: computação de alto desempenho, computação verde e monitoramento de infraestrutura.

As comparações entre provedores e quais as funcionalidades disponibilizadas por eles é de extrema importância, uma vez que recursos financeiros são envolvidos. Esse fato evidencia que é importante manter um histórico de comparações, dada a natureza dinâmica da nuvem.

Outro ponto de destaque é a alocação de VM e tipos de virtualizadores utilizados pelos provedores de nuvem. A forma como as VMs são dispostas nos hospedeiros pode determinar um melhor ou pior desempenho, influenciando assim na QoS. Verificar quais as diferenças

entre os virtualizadores também se torna necessário. Como cada virtualizador possui uma arquitetura e objetivos diferentes, é necessário definir em quais momentos utilizar ou um ou outro virtualizador. Inclusive, cada provedor pode utilizar um virtualizador diferente, dificultando assim a interoperabilidade e a definição do melhor provedor de serviços.

3.5 Virtualização

Segundo [Che et al. \(2008\)](#), a virtualização traz vantagens, entretanto traz também algumas desvantagens que levam à sobrecarga no desempenho. Esse fato faz com que vários estudos sejam voltados a identificar os desafios proporcionados pela virtualização.

[Che et al. \(2008\)](#) avaliam os virtualizadores Xen e KVM, considerando fatores como: uso intensivo de CPU e desempenho em operações de leitura e escrita. O resultado obtido demonstrou que o Xen possui desempenho melhor do que o KVM em atividades estritamente *CPU-bound* e um desempenho pior nos testes que envolveram o uso intensivo de operações de entrada e saída. Foram avaliadas também as principais características das VMs com relação ao isolamento e escalabilidade. A maior diferença entre os virtualizadores foi relacionada à escalabilidade.

No trabalho de [Xu et al. \(2008\)](#), além dos virtualizadores Xen e KVM, o virtualizador Vmware foi utilizado na avaliação. Essa avaliação foi composta por: avaliação do desempenho global, isolamento e escalabilidade. Os resultados obtidos em relação aos fatores avaliados mostraram que a maior diferença de desempenho entre os virtualizadores foi constatada no teste de uso intensivo de rede, onde tanto o Vmware quanto o Xen apresentaram bons resultados.

Já o trabalho apresentado por [Chierici e Veraldi \(2010\)](#) apresenta uma análise quantitativa entre os virtualizadores KVM e Xen. No caso do Xen, os autores utilizaram as duas formas de virtualizar permitidas pelo Xen (virtualização completa e paravirtualização). A motivação para a avaliação foi perceber que o KVM possui bons resultados com aplicações científicas. Os testes realizados foram relacionados ao uso de CPU, acesso a disco e rede. Os experimentos apresentam o Xen (tanto virtualização completa como paravirtualização) como a melhor solução entre os três testes realizados.

[White e Pilbeam \(2010\)](#) mencionam o fato de haver divergências de resultados em outros trabalhos sobre virtualização devido à não maturidade das técnicas utilizadas. No trabalho, foram discutidas as formas de virtualização de recursos como: emulação, virtualização nativa, paravirtualização, virtualização em nível de sistema operacional, virtualização de recursos (por exemplo, rede e vídeo) e virtualização de aplicações. Segundo os autores, a emulação é a técnica que possui o pior desempenho, enquanto que as técnicas que garantem o melhor desempenho são a paravirtualização (Xen) e a virtualização no nível do sistema operacional KVM.

Uma análise de tecnologias de virtualização para ambientes de computação de alto desempenho é apresentada por [Younge et al. \(2011\)](#). A avaliação dos autores considerou aspectos

funcionais de quatro virtualizadores, como suporte à paravirtualização e virtualização completa, quantidade de memória suportada e quantidade de núcleos suportados. Além dessa avaliação, foi abordada uma avaliação quantitativa entre o Xen, o KVM e o VirtualBox, onde os melhores desempenhos foram obtidos pelo virtualizador KVM.

Leite *et al.* (2012) apresentam uma avaliação de desempenho sobre os virtualizadores que podem ser utilizados na nuvem. Os autores avaliaram os virtualizadores Xen e KVM em relação às cargas de trabalho de CPU e sistema. As comparações de desempenho foram realizadas comparando uma VM com a máquina física, com a finalidade de estimar a porcentagem de degradação causada pelo uso da camada de virtualização e em relação às quantidades de máquinas virtuais concorrendo por recursos. Nos resultados o virtualizador KVM foi o que obteve os melhores resultados.

O estudo realizado por Babu *et al.* (2014) apresenta uma comparação entre os virtualizadores Xen e Openvz. Neste estudo os autores utilizaram o Xen tanto como virtualização completa, como paravirtualização. Dessa forma, os autores avaliam os três principais modelos de virtualização (paravirtualização, virtualização completa e virtualização por contêiner). Os resultados obtidos demonstraram que o virtualizador Openvz possui bons desempenhos quando os acessos ao processador são considerados. No entanto, a virtualização completa do Xen foi a que obteve os melhores desempenhos.

Em Carvalho e Bellezi (2014), uma análise sobre os virtualizadores Xen e KVM é apresentada. Nesse estudo os autores avaliaram: a escrita e leitura em disco, acesso à memória, CPU, largura de banda de rede e escalabilidade. Os resultados demonstraram que o Xen foi superior ao KVM, inclusive nos acessos ao disco.

Nadeem e Qaiser (2015) apresentam uma avaliação sobre três diferentes plataformas de nuvem privada. Nesse estudo cada plataforma possui um virtualizador diferente, sendo: KVM, Xen e Openvz. Foram considerados: o tempo de resposta, *throughput* do processador, desempenho de memória, memória cache e operações de entrada e saída. Nos resultados, os autores indicam que não há uma solução melhor que outra. Na verdade, para cada tipo de situação um virtualizador pode ser melhor que o outro.

A Tabela 5 apresenta um resumo dos trabalhos apresentados sobre virtualização.

Como apresentado na Tabela 5, há vários trabalhos que consideram a avaliação ou comparação de desempenho entre virtualizadores. Essa avaliação é necessária uma vez que esses virtualizadores possuem arquiteturas diferentes, como é o caso do Xen, KVM e Openvz. Esses estudos demonstram que as tecnologias de virtualização estão em constante evolução. Outro ponto é a utilização dessa tecnologia como suporte a outros sistemas computacionais, como é o caso da computação em nuvem.

Tabela 5 – Trabalhos relacionados à avaliação de desempenho utilizando virtualização

Trabalho	Tipo de Avaliação	Virtualizador
Che et al. (2008)	Avaliam os virtualizadores considerando o uso de CPU e operações de leitura e escrita	Xen KVM
Xu et al. (2008)	Avaliação do desempenho global, isolamento e escalabilidade	Xen, KVM Vmware
Chierici e Veraldi (2010)	Avaliação considerando aplicações científicas.	Xen e KVM
White e Pilbeam (2010)	Descrição das técnicas de emulação, virtualização nativa, paravirtualização, virtualização em nível de sistema operacional e virtualização de recursos	Xen KVM
Younge et al. (2011)	Análise de tecnologias de virtualização para ambientes de computação de alto desempenho	Xen, KVM Virtualbox
Leite et al. (2012)	Avaliação de desempenho sobre os monitores de máquinas virtuais que podem ser utilizados na nuvem	Xen, KVM
Babu et al. (2014)	Avaliação sobre os três modelos de virtualização	Xen e OpenVz
Carvalho e Bellezi (2014)	Avaliação dos virtualizadores considerando diversos benchmarks para CPU, memória e entrada e saída.	Xen e KVM
Nadeem e Qaiser (2015)	Apresentam uma avaliação sobre três tipos de nuvens privadas onde cada uma utiliza um tipo de virtualização diferente	Xen, KVM e OpenVz

Fonte: Dados da pesquisa.

3.6 Considerações finais

Neste capítulo foram explorados os trabalhos que estão relacionados ao problema investigado nesta tese de doutorado, identificando os principais conceitos que podem ser utilizados.

Por se tratar de um trabalho experimental, foi necessário fazer um levantamento das metodologias existentes na avaliação de sistemas e que podem ser aplicadas na computação em nuvem.

Outro ponto é a relação entre o desempenho e a infraestrutura utilizada. Nesse contexto, foram pesquisados trabalhos sobre QoS em nuvem e como os autores utilizam os conceitos relacionados. Nesse estudo foi verificado que há vários atributos de QoS que são utilizados na nuvem. Sendo os atributos desempenho e custo os mais recorrentes. Além disso, nos trabalhos relacionados à QoS, a metodologia aplicada define um limite de desempenho e qual a porcentagem de ganho ou perda que cada novo método garante.

Os métodos aplicados à QoS em nuvem, e que são relacionados ao desempenho, consideram que a manutenção da infraestrutura virtualizada pode garantir um aumento ou manutenção da QoS já estabelecida.

Além disso, os trabalhos relacionados demonstram a importância da camada de virtualização e como a mesma pode ser decisiva no fornecimento dos serviços. Os serviços hospedados em VMs podem ter desempenho diferente, dependendo do virtualizador utilizado. As características observadas nos trabalhos relacionados conduziram à seleção de uma ferramenta para o desenvolvimento de um protótipo de nuvem, onde as características de virtualização possam ser exploradas, tanto em relação a comparações de virtualizadores, quanto em relação à manutenção dos recursos virtualizados. Essas características são apresentadas no Capítulo 4.

PROTÓTIPO PARA AVALIAÇÃO DE COMPUTAÇÃO EM NUVEM

4.1 Contextualização

Neste capítulo será apresentado o protótipo de nuvem desenvolvido. Esse protótipo utiliza a ferramenta Cloudstack para o desenvolvimento de uma nuvem onde há suporte a mais de uma tecnologia de virtualização.

Na Figura 6 são apresentados os componentes que compõem cada uma das três camadas oferecidas pela computação em nuvem. O foco deste trabalho está na camada de infraestrutura, mais especificamente na entrega de serviços virtualizados. Na Figura 6, os componentes considerados nesta tese estão circundados por uma linha pontilhada.

Figura 6 – Componentes considerados neste projeto de doutorado

Fonte: Elaborada pelo autor.

Os componentes apresentados na Figura 6 representam a divisão dos serviços de nuvem e o que cada camada fornece. Como a camada de virtualização será considerada, os pontos relacionados à avaliação de desempenho manutenção de SLA e monitoramento serão explorados.

O desempenho na entrega de serviços virtualizados pode ser essencial para a obtenção de QoS em nuvem. Como já apresentado, a virtualização é a base para o desenvolvimento dos demais serviços. Os pontos considerados neste trabalho apresentam alguns desafios, como:

- **Avaliação de desempenho** - Não há limites de desempenho específicos para a nuvem, ou seja, não se sabe ao certo qual desempenho esperar de uma VM. De acordo com [Ward e Barker \(2014\)](#), alguns provedores como o Amazon EC2 e o Google apresentam algumas formas de tentar solucionar esse problema introduzindo as unidades de computação (CPU *unit*). No entanto, essas unidades são utilizadas em apenas alguns tipos de VMs e representam a capacidade de um núcleo lógico (*hyperthread*). Realizar avaliações de desempenho é importante para definir os níveis adequados de desempenho que podem ser obtidos.
- **Manutenção de SLA** - Os SLA consideram poucos atributos de QoS e não apresentam um motivo claro para a não utilização de um conjunto de atributos de QoS ([CASALICCHIO; SILVESTRI, 2013](#)) ([WARD; BARKER, 2014](#)).
- **Monitoramento** - Vários tipos de cargas de trabalho são impostos à nuvem ([HAN et al., 2014](#)) ([LIU et al., 2014b](#)) ([FATEMA et al., 2014](#)) ([LIANG et al., 2014](#)). Monitorar como é feita a alocação de máquinas virtuais nos hospedeiros se torna essencial no fornecimento de qualidade aos serviços de infraestrutura.

Quando os clientes contratam infraestrutura de nuvem, é necessário que eles obtenham toda a informação possível sobre os recursos contratados, para ter certeza de que os provedores irão fornecer as garantias de desempenho requeridas.

Essas informações podem ser utilizadas na relação custo/benefício que os clientes podem considerar ao utilizar um serviço. Por exemplo, quais são as diferenças, além do preço, entre um provedor A e um provedor B para uma VM que contém o mesmo número de vCPU, memória e disco.

Atualmente os clientes têm apenas os custos como base para tomar uma decisão de qual provedor é o melhor. Diferente da computação em nuvem, na computação tradicional, quando os clientes adquirem uma infraestrutura física, eles usam informações técnicas para selecionar o melhor recurso.

Assim, surge a seguinte questão: por que poucas informações técnicas são fornecidas em computação em nuvem? Embora a computação em nuvem funcione como um serviço, os clientes precisam saber quais são as opções e assim tomar decisões.

Com base nessas considerações, neste trabalho são avaliados alguns pontos que podem influenciar na qualidade dos serviços, focando principalmente nos serviços de IaaS, e mais especificamente na virtualização.

4.2 Desenvolvimento da infraestrutura de nuvem

Como a camada de infraestrutura de nuvem possui vários componentes, desenvolver um ambiente que pode ser utilizado como protótipo se torna um desafio. Sendo assim, a ferramenta Cloudstack foi utilizada para o desenvolvimento do protótipo utilizado neste doutorado.

A Cloudstack é uma plataforma de código aberto que reúne funcionalidades para o desenvolvimento de nuvens públicas, privadas e híbridas e o fornecimento de serviços de IaaS (KUMAR *et al.*, 2014). Todos os componentes necessários para uma nuvem funcionar, com: rede interna, armazenamento e serviços de computação são gerenciados pelo Cloudstack.

Uma das vantagens na utilização da Cloudstack é a possibilidade de utilização de mais de um virtualizador no processo de gerenciamento do serviço de computação, sendo suportados os virtualizadores: Xen, VMware, KVM e Openvz (KUMAR *et al.*, 2014). Além disso, é possível desenvolver uma série de topologias e distribuição da infraestrutura, uma vez que há a possibilidade de realizar o gerenciamento de rede dos nós que compoem a infraestrutura. Dessa forma, a topologia apresentada na Figura 7 foi utilizada.

Figura 7 – Modelo de desenvolvimento utilizado

Fonte: Elaborada pelo autor.

Como apresentado na Figura 7, a infraestrutura desenvolvida conta com: um *firewall*, um servidor de manutenção, um servidor de armazenamento e os nós de computação, que são os componentes básicos para o desenvolvimento de um ambiente de nuvem.

Na topologia apresentada, o *firewall* é um servidor independente que faz a ligação entre as redes internas e externas. Ele foi desenvolvido utilizando o sistema ClearOs, um sistema operacional voltado para o gerenciamento de rede e que possui funcionalidades como, encaminhamento de porta, desenvolvimento de redes virtuais privadas (*Virtual Private Network* (VPN)) e outras funções relacionadas ao gerenciamento de manutenção da rede (FUERTES *et al.*, 2014).

O servidor de acesso ou de manutenção é o nó principal e contém todas as informações sobre as contas dos clientes e o controle sobre a infraestrutura disponível. No desenvolvimento deste protótipo, o servidor de armazenamento ficou alocado junto ao servidor de manutenção.

Com relação ao armazenamento, a Cloudstack utiliza dois tipos de armazenamento, chamados de armazenamentos primário e secundário. No armazenamento primário ficam as imagens das VMs, tanto de sistema como de clientes. Já o armazenamento secundário é utilizado para manter cópias de VMs, quando necessário. O acesso aos dois armazenamentos é realizado por meio do sistema de arquivo em redes do Linux (*Network File System* (NFS)) (KUMAR *et al.*, 2014).

Os nós de computação são os responsáveis pela execução das VMs dos clientes. Esses nós são ligados por agentes Cloudstack ao nó de manutenção. Esses agentes permitem que os nós de computação façam acesso ao banco de dados e tenham as informações relacionadas à cada VM dos clientes. Todos os nós envolvidos no desenvolvimento do protótipo possuem a mesma configuração, apresentada na Tabela 6.

Tabela 6 – Características físicas dos hospedeiros

Quantidade de nós	10
Memória	16 GB (2 x 8 GB) 1333 MHz DDR3
Processador	Intel Core i7 870 2.93 GHz
Cores	4
Total de Threads	8
Cache	8 MB
Disco	SATA 1 TB 7.200 rpm
Sistema Operacional	Ubuntu Server x64 12.04 LTS Kernel 3.11

Como apresentado na Tabela 6, todas as máquinas físicas possuem o mesmo tipo de hardware e suporte à virtualização. Esse suporte à virtualização é necessário para o funcionamento das VMs e para a ligação entre os nós de computação e o servidor de manutenção.

Todo o gerenciamento da infraestrutura e dos clientes é apoiado pelo uso de banco de dados. Todas as atividades são registradas no banco de dados da Cloudstack. Por se tratar de um banco de dados extenso, apenas o gerenciamento da infraestrutura e o gerenciamento das VMs serão explorados.

4.2.1 Gerenciamento da infraestrutura

A Cloudstack utiliza um banco de dados para gerenciar todas as informações relacionadas à nuvem. Nesse banco de dados são salvos desde as informações da infraestrutura até o cadastro dos clientes.

Com a implantação da Cloudstack como protótipo, foi possível acessar o banco de dados utilizado pela ferramenta e assim explorar quais funcionalidades são importantes e que são registradas em banco. Dessa forma, na Figura 8 são apresentadas as tabelas que são utilizadas para fazer o gerenciamento da infraestrutura.

Figura 8 – Tabelas utilizadas para o gerenciamento das informações do protótipo

Fonte: Elaborada pelo autor.

Como apresentado na Figura 8, a tabela *datacenter* possui todas as informações relacionadas aos clientes (*account*), aos clusters e aos domínios (*domain*). É possível notar uma grande quantidade de informação sobre elementos de rede, provedores de dados e estados de alocação. Já na tabela *cluster*, as informações sobre propriedade, tipos de virtualizadores e informações de manutenção são utilizadas para manter os vários clusters que podem compor o *data center*. No modelo utilizado pela Cloudstack é possível ter um ambiente heterogêneo com relação à

virtualizadores, possibilitando que cada *cluster* possua seu próprio tipo de virtualização.

Nas tabelas clientes e domínio, as informações e regras de acesso são definidas. É possível ter dois tipos de contas: administrador e cliente sem privilégios. Dessa forma, o cliente tem acesso apenas ao processo de criação da VM e o acesso a ela. Outro ponto importante é a possibilidade de domínios, com esses domínios é possível classificar os clientes ou mesmo fazer uma separação por tipo de serviço, facilitando assim a alocação de VMs nos *clusters*. Além disso, essa classificação permite focar em QoS tanto em relação à classificação de clientes fornecendo possíveis diferenciações de serviços como em agrupamento de VMs com o mesmo propósito.

4.2.2 Gerenciamento de máquinas virtuais

Assim como as informações do *data center*, as informações das máquinas virtuais também são gerenciadas por meio de banco de dados. A Cloudstack utiliza um conjunto de tabelas para fazer o gerenciamento das VMs, como apresentado na Figura 9.

Figura 9 – Tabelas utilizadas no gerenciamento das VMs

Fonte: Elaborada pelo autor.

Como apresentado na Figura 9, há a definição da VM e nessa tabela há a presença de todas as informações relacionadas ao uso da VM. Esse conjunto de informações é importante, uma vez que há a presença de dois tipos de VMs; VMs do sistema e VMs dos clientes. As VMs do sistema são utilizadas para fazer o gerenciamento de rede e gerenciamento de armazenamento, sendo acessadas apenas pelos usuários administradores. Já as VMs dos clientes são manipuladas

apenas pelos mesmos, sendo que ao administrador é possível apenas migrar a VM de um nó para outro.

As informações das máquinas virtuais são utilizadas para o gerenciamento de quantidade de recursos e alocação nos nós de computação disponíveis. Essa alocação é feita com base nas informações de utilização do sistema. As principais tabelas que armazenam as informações de utilização são apresentadas na Figura 10.

usage_vm_instance	cloud_usage	usage_event	usage_job
usage_type INT(1)	id BIGINT(20)	id BIGINT(20)	id BIGINT(20)
zone_id BIGINT(20)	zone_id BIGINT(20)	type VARCHAR(32)	host VARCHAR(255)
account_id BIGINT(20)	account_id BIGINT(20)	account_id BIGINT(20)	pid INT(5)
vm_instance_id BIGINT(20)	domain_id BIGINT(20)	created DATETIME	job_type INT(1)
vm_name VARCHAR(255)	description VARCHAR(1024)	zone_id BIGINT(20)	scheduled INT(1)
service_offering_id BIGINT(20)	usage_display VARCHAR(255)	resource_id BIGINT(20)	start_millis BIGINT(20)
template_id BIGINT(20)	usage_type INT(1)	resource_name VARCHAR(255)	end_millis BIGINT(20)
hypervisor_type VARCHAR(255)	raw_usage DOUBLE	offering_id BIGINT(20)	exec_time BIGINT(20)
start_date DATETIME	vm_instance_id BIGINT(20)	template_id BIGINT(20)	start_date DATETIME
end_date DATETIME	vm_name VARCHAR(255)	size BIGINT(20)	end_date DATETIME
cpu_speed INT(10)	offering_id BIGINT(20)	resource_type VARCHAR(32)	success INT(1)
cpu_cores INT(10)	template_id BIGINT(20)	processed TINYINT(4)	heartbeat DATETIME
memory INT(10)	usage_id BIGINT(20)	virtual_size BIGINT(20)	
	type VARCHAR(32)		
	size BIGINT(20)		
	network_id BIGINT(20)		
	start_date DATETIME		
	end_date DATETIME		
	virtual_size BIGINT(20)		
	cpu_speed INT(10)		
	cpu_cores INT(10)		
	memory INT(10)		
Indexes	Indexes	Indexes	Indexes

Figura 10 – Tabelas com as informações de utilização de recursos

Fonte: Elaborada pelo autor.

Como apresentado na Figura 10, as informações de utilização dos recursos são armazenadas em diferentes tabelas. As informações de utilização da nuvem e das VMs são úteis na elaboração de políticas de alocação de VMs. Esse fato se deve à quantidade significativa de informações sobre os recursos utilizados, como: memória e processador. Já as tabelas utilização por evento (event) e trabalho (job) fazem o registro das atividades de gerenciamento dos recursos disponíveis, como: quem realizou determinada ação e quais recursos foram utilizados.

As informações apresentadas até o momento são relacionadas apenas à visão do administrador da nuvem e são utilizadas para realizar as configurações necessárias do sistema. Na Figura 11 são apresentadas as funcionalidades que o administrador do sistema pode utilizar.

	Nome	Descrição	Valor	Ações
Instâncias				
Grupos de Afinidade				
Storage				
Rede				
Templates				
Eventos				
Projetos				
Contas				
Dominios				
Regiões				
Infra-estrutura				
Configurações Globais				
Oferta de Serviços				
	host.capacityType.to.order.clusters	The host capacity type (CPU or RAM) is used by deployment planner to order clusters during VM resource allocation	CPU	
	host.reservation.release.period	The interval in milliseconds between host reservation release checks	300000	
	host.retry	Number of times to retry hosts for creating a volume	2	
	host.stats.interval	The interval (in milliseconds) when host stats are retrieved from agents.	60000	
	hypervisor.gurus.exclude	Extensions to exclude from being registered		
	hypervisor.list	The list of hypervisors that this deployment will use.	Hyperv,KVM,...	
	incorrect.login.attempts.allowed	Incorrect login attempts allowed before the user is disabled	5	
	instance.name	Name of the deployment instance.	VM	
	integration.api.port	Default API port		
	internalbvm.service.offering	Uuid of the service offering used by internal lb vm; if NULL - default system internal lb offering will be used		
	interval.baremetal.securitygroup.agent.echo	Interval to echo baremetal security group agent, in seconds	10	

Figura 11 – Configuração das características dos hospedeiros no protótipo

Fonte: Elaborada pelo autor.

Na Figura 11 são apresentadas as configurações globais da Cloudstack, nela é possível definir qual o tipo de recurso será utilizado para alocar as VMs aos hospedeiros. No caso deste trabalho de doutorado, o tipo utilizado foi o processamento (CPU). Além dessa informação, outras informações relacionadas ao gerenciamento do hospedeiro são utilizadas, como o tempo para a reserva de recursos, o número de tentativas de criar uma VM no hospedeiro e questões relacionadas ao acesso. Na Figura 12 é apresentado o conjunto de recursos que podem ser disponibilizados.

	Nome	Descrição	Valor	Ações
Instâncias				
Grupos de Afinidade				
Storage				
Rede				
Templates				
Eventos				
Projetos				
Contas				
Domínios				
Regiões				
Infra-estrutura				
Configurações Globais				
Oferta de Serviços				
	max.account.cpus	The default maximum number of cpu cores that can be used for an account	40	
	max.account.memory	The default maximum memory (in MiB) that can be used for an account	40960	
	max.account.networks	The default maximum number of networks that can be created for an account	20	
	max.account.primary.storage	The default maximum primary storage space (in GiB) that can be used for an account	200	
	max.account.public.ips	The default maximum number of public IPs that can be consumed by an account	20	
	max.account.secondary.storage	The default maximum secondary storage space (in GiB) that can be used for an account	400	
	max.account.snapshots	The default maximum number of snapshots that can be created for an account	20	
	max.account.templates	The default maximum number of templates that can be deployed for an account	20	
	max.account.user.vms	The default maximum number of user VMs that can be deployed for an account	20	
	max.account.volumes	The default maximum number of volumes that can be created for an account	20	
	max.account.vpcs	The default maximum number of vpcs that can be created for an account	20	

Figura 12 – Definição do limite de recursos disponibilizados por conta

Fonte: Elaborada pelo autor.

Como apresentado na Figura 12, é possível definir quais os limites máximos de recursos que poderão ser disponibilizados por conta. Esses recursos envolvem o número de VMs, a quantidade de armazenamento, a quantidade de memória, a quantidade de endereços de rede e quantidade de processamento.

Todos os recursos disponíveis podem ser configurados pelo administrador do sistema, dessa forma, a utilização correta dos recursos pode levar ao fornecimento de serviços com qualidade e desempenho.

As características apresentadas são relacionadas ao desenvolvimento padrão da Cloudstack. Como essa ferramenta pode ser utilizada para o desenvolvimento de nuvem, tornou-se conveniente o seu uso. Por se tratar de um protótipo os serviços de virtualização não estão disponíveis externamente. Esses recursos estão disponibilizados apenas em uma rede local, para a realização de experimentos, e podem futuramente serem disponibilizados.

Nesse trabalho de doutorado as funcionalidades de rede e de armazenamento não foram consideradas. Essas funcionalidades estão presentes no protótipo, no entanto, o objetivo foi investigar como ocorre a alocação de VMs, como avaliar seu desempenho e como melhorar o desempenho das mesmas.

4.3 Considerações finais

Neste capítulo foram apresentadas as características consideradas utilizadas no desenvolvimento do protótipo. Dentre as características apresentadas estão: a identificação dos pontos que foram explorados e serão validados nos experimentos e a apresentação de detalhes do protótipo utilizado para realizar a pesquisa.

Com a utilização do protótipo foi possível verificar que a nuvem possui uma grande quantidade de informações que devem ser gerenciadas. Essas informações vão desde a classificação do cliente até o gerenciamento de máquinas virtuais. Várias funcionalidades podem ser exploradas, como a definição de parâmetros para serem utilizados nas configurações globais da ferramenta.

Apesar da ferramenta proporcionar um ambiente que possui várias funcionalidades, o objetivo é implementar esse protótipo para verificar como a virtualização pode influenciar no desempenho final de aplicações em nuvem.

No próximo capítulo serão apresentadas avaliações referentes aos provedores de nuvem e a validação do protótipo. Além disso, as tecnologias de virtualização, Xen, KVM e Openvz que são suportados pela ferramenta Cloudstack, serão avaliados.

RESULTADOS

5.1 Considerações iniciais

Este capítulo apresenta a metodologia utilizada e os resultados obtidos na validação das hipóteses levantadas nesta tese de doutorado. Foram utilizados um ambiente controlado e um ambiente não controlado. O ambiente não controlado é o ambiente de nuvem pública, onde, os experimentos foram realizados em apenas uma VM, a fim de minimizar as influências externas e possibilitando a comparação com os resultados obtidos no ambiente controlado.

Os resultados foram obtidos de forma empírica e experimental, onde os resultados experimentais seguiram uma metodologia para a execução e análise dos dados, com a finalidade de validá-los estatisticamente.

5.2 Planejamento de experimento

Foram utilizadas duas formas de avaliação para a obtenção dos resultados apresentados nesta tese de doutorado, uma avaliação empírica e avaliações experimentais. Na avaliação empírica, uma pesquisa sobre os provedores de IaaS foi realizada. Para essa pesquisa foi utilizado um conjunto de questões, com a finalidade de realizar o levantamento de informações de *hardware* e *software* sobre esses provedores. As questões foram:

- **Informações de processador** - Tipo, frequência e *hypherthread*;
- **Informações de memória** - Tipo, frequência e quantidade;
- **Informações de armazenamento** - Tipo;
- **Informações de software** - Virtualizador disponibilizado.

Essas informações contribuíram para as definições do ambiente controlado e para as demais comparações entre os provedores que foram utilizados.

Já as avaliações experimentais seguiram a metodologia apresentada por Jain (1991). A metodologia utilizada propõe que o ambiente seja conhecido e o experimento seja controlado, possibilitando assim a reprodução dos resultados obtidos e a validação dos dados de forma estatística.

Neste trabalho de doutorado são considerados cinco pontos principais. Esses pontos são: definição dos experimentos, definição da carga, arquitetura alvo, log dos eventos e análise dos dados. Esses cinco pontos, que foram utilizados neste projeto, são ilustrados na Figura 13.

Figura 13 – Etapas consideradas na elaboração desta pesquisa

Fonte: Elaborada pelo autor.

Na Figura 13 são apresentadas cinco etapas que devem ser abordadas de forma independente e ao final da realização das cinco etapas é possível obter resultados que podem ser repetidos e que exploram partes específicas do sistema avaliado.

A primeira etapa é definir qual parte do sistema será testada. Em sistemas como a nuvem, que é composta por pilhas de componentes e *softwares*, nem sempre é possível definir fatores fixos com apenas dois níveis ou variações, como é o caso dos virtualizadores, por exemplo. Os experimentos consideram todas as combinações possíveis entre fatores e níveis utilizados em um determinado experimento. Nesta pesquisa, os principais fatores são:

- **Tipos de provedores** - Amazon EC2 e Microsoft Azure.
- **Tipos de virtualizadores** - Xen, KVM e OpenVz ;
- **Quantidade de máquinas virtuais** - variação de $(1...N)$ máquinas virtuais;
- **Quantidade de CPU virtual (vCPUs)** - variação entre $(1, 4, 8)$ vCPU, que é o limite da infraestrutura utilizada ;

A segunda etapa é a definição da carga de trabalho. Como a QoS é considerada, *benchmarks* foram utilizados como cargas de trabalho sintéticas, a fim de explorar ao máximo a capacidade das VMs. Dessa forma, é analisado o comportamento do *benchmark* em apenas uma VM e o resultado obtido é tido como base, ou seja, para cada configuração de VM é realizado um teste base para definir os limites de desempenho esperados. Esse desempenho é relacionado diretamente à QoS esperada da VM. Nesse caso, cada *benchmark* representa um tipo de aplicação hospedada na VM. Sendo assim, utilizando a ferramenta *Phoronix Test Suite* (LARABEL; TIPPETT, 2011), foram considerados os seguintes *benchmarks*:

- **N-queens** - Essa carga é relacionada ao tempo de solução do problema das N rainhas em um tabuleiro de tamanho 18. É um problema combinatório exponencial que necessita de uma elevada carga de processamento, logo é uma carga de trabalho do tipo CPU-bound. Para a solução da mesma, é utilizada técnica de concorrência com OpenMP.
- **SmallPt** - Essa carga é relacionada a um renderizador de iluminação global. A iluminação global é realizada por caminhos imparciais de Monte Carlo. Essa carga, apesar de possuir suporte à paralelismo, foi utilizada de forma sequencial.
- **Ramspeed** - Essa carga é relacionada às operações de escrita, reescrita e leitura realizadas na memória.
- **Apache 1.6** - Essa carga é relacionada às solicitações por segundo que um determinado sistema pode atender quando são realizados 1.000.000 pedidos, sendo 100 solicitações realizadas simultaneamente. Essa carga de trabalho pode ser considerada de sistema, pois há uma grande quantidade de utilização de memória, disco e processador.

Os *benchmarks* citados foram utilizados para verificar a utilização da VM, como um indicador de qualidade do serviço, e outros benchmarks poderiam ser utilizados. No entanto, o escopo do projeto não está relacionado diretamente à aplicação e sim ao comportamento da VM quando uma aplicação está em uso.

Da arquitetura alvo (computação em nuvem) foi selecionada a virtualização como objeto de estudo. Essa especificação permite definir de forma clara o escopo a ser seguido, evitando considerar sistemas como a rede, que são tão importantes como a virtualização, porém estão fora do escopo deste trabalho. Os Logs são parte integrante da arquitetura alvo e servem para reutilizar ou explorar resultados sem a necessidade da realização de novos experimentos.

A análise dos resultados é apoiada pela ferramenta Minitab. Essa ferramenta auxilia na análise dos dados por possuir várias funções estatísticas já definidas e por facilitar a geração de gráficos.

O processo de avaliação, desde o planejamento dos experimentos até a sua análise passa então pelas cinco etapas já discutidas. Esse processo é melhor é apresentado na Figura 14.

Figura 14 – Atividades realizadas no desenvolvimento dos experimentos

Fonte: Elaborada pelo autor.

De acordo com a Figura 14, é possível verificar que uma atividade só é realizada após o término da atividade anterior, exceto as atividades numeradas como 5 e 6. Quando não há nenhum experimento no Log, a definição da carga de trabalho (2) só será realizada após a definição dos experimentos (1). Essas atividades são essenciais, pois no planejamento de experimentos é definido o conjunto de experimentos e qual será a variável de resposta (ou o objeto de estudo) do experimento. A exceção é quando a informação já está no Log, nesse caso é possível ir direto para a atividade (5) e analisar os dados na atividade (6).

O processo descrito foi seguido em todos os conjuntos de experimentos, incluindo os experimentos realizados entre provedores de IaaS. Para todos os conjuntos de experimentos foram consideradas N execuções do *benchmark* (carga de trabalho) selecionado, possibilitando assim realizar as análises estatísticas. As N execuções são utilizadas para a composição das médias e obtenção do desvio padrão e erro obtido nas observações. O desvio e o erro são descritos de acordo com as Equações 5.1 e 5.2.

$$D_p = \sqrt{\sum_{i=1}^n \frac{(M - a_i)^2}{(N - 1)}} \quad (5.1)$$

$$\left(\frac{D_p * Const}{\sqrt{N}} \right) \quad (5.2)$$

O desvio padrão (D_p) apresentado na Equação 5.1 é calculado a partir da observação entre a média obtida (M) e cada amostra ($a_1 \dots a_n$) em um conjunto de N execuções. Já o erro é obtido pela relação entre a quantidade de observações e os desvios obtidos e a aplicação da constante *t-student*. Em ambos os casos os cálculos são realizados pela ferramenta Minitab,

sendo utilizados como parâmetros de comparação estatística entre os resultados obtidos. Essas funções não serão amplamente discutidas uma vez que são utilizadas apenas para validar os dados.

5.3 Informações sobre as VMs e desempenho entre provedores de IaaS

Nesta seção serão analisados os resultados obtidos por observações empíricas e os resultados obtidos de forma experimental. Em ambos os casos, os provedores de nuvem mais citados na literatura foram abordados.

5.3.1 Informações sobre a contratação de VMs

Esse primeiro estudo tem como objetivo identificar quais informações são disponibilizadas pelos provedores de IaaS e como essas informações podem ser utilizadas pelos clientes para a definição de qual serviço contratar.

Como descrito na metodologia deste capítulo, um levantamento foi realizado considerando algumas questões sobre informações de *hardware* e *software*. Os resultados obtidos são apresentados na Tabela 7.

Tabela 7 – Informação de infraestrutura fornecida pelos provedores

Provedor	Proc	Virt.	Disco
Amazon EC2	Intel Xeon E5-2670 TH	XEN (<i>Full</i> /PV)	SSD
Google	-	KVM	HDD e SSD
Azure	-	-	-
Rackspace	<i>Thread Hardware</i>	-	SSD
GoGrid	Intel Xeon 5600 series	XEN	-
HP	Intel Xeon 2.60GHz TH	KVM	-

Fonte: Dados da pesquisa.

De acordo com a Tabela 7, é possível notar que nem sempre toda a informação é disponibilizada. Essa característica pode se tornar um problema na contratação de serviços. Na Tabela 8 é apresentada uma relação de VMs com propósitos semelhantes e seus respectivos preços.

Tabela 8 – Comparação de preço considerando VMs com o mesmo propósito

Provedor	VM (vCPU - Memória)	SLA	Desconto	Preço (Linux)
Amazon Ec2	2 vCPU – 3,75 GB	Disponibilidade	30% se $\leq 90\%$	\$0.11 por hora
Google	1 vCPU – 3,75 GB	Disponibilidade	50% se $\leq 95\%$	\$0.063 por hora
Microsoft	2 vCPU – 3,5 GB	Disponibilidade	-	\$0.12 por hora
Rackspace	2 vCPU – 3,75 GB	Disponibilidade	30% se $\leq 99\%$	\$0.10 por hora
Gogrid	2 vCPU – 2 GB	Disponibilidade	-	\$0.12 por hora
HP	2 vCPU – 4 GB	Disponibilidade	30% se $< 99\%$	\$0.12 por hora

Fonte: Dados da pesquisa.

Como pode ser observado na Tabela 8, nem sempre é possível obter uma VM com capacidades semelhantes entre os provedores. Normalmente os provedores estabelecem capacidades fixas para as VMs e as classificam para propósitos específicos, como: orientado à computação, ao uso intensivo de memória e uso intensivo de disco. Essas classificações normalmente são relacionadas ao tipo de aplicação que será desenvolvida na VM.

Nota-se que há duas deficiências na tomada de decisão, a primeira é relacionada à ausência de informações de *hardware* e *software* que, podem revelar características de desempenho e a segunda é a forma fixa de composição das VMs em relação aos recursos disponíveis.

Tomando como base os resultados apresentados, foi proposto um conjunto de experimentos considerando dois dos provedores descritos (EC2 e Microsoft). Esse conjunto de experimentos visa a identificação de desempenho que cada provedor fornece quando uma determinada carga de trabalho é imposta nas VMs.

5.3.2 Comparação de desempenho entre os provedores Amazon e Microsoft

Esse conjunto de experimentos considera os provedores Amazon e Microsoft uma vez que os dois possuem VMs com as configurações semelhantes, sendo que a VM da Microsoft possui menos memória e um preço maior por hora do que o apresentado pela Amazon.

Além disso, esse primeiro estudo visa verificar se o comportamento obtido nos provedores de nuvem é semelhante ao do protótipo utilizado para a realização deste trabalho de doutorado, validando assim a infraestrutura utilizada para os demais experimentos. Para a composição dos experimentos foram utilizadas as configurações apresentadas na Tabela 9.

Tabela 9 – Configuração das máquinas virtuais

Infraestrutura	vCPU	Memória	Sistema Operacional
Microsoft	2	3,5 GB	Ubuntu 12.04
Microsoft	4	7 GB	Ubuntu 12.04
Amazon	2	3,75 GB	Ubuntu 12.04
Amazon	4	7,5 GB	Ubuntu 12.04
Protótipo	2	3,75 GB	Ubuntu 12.04
Protótipo	4	7,5 GB	Ubuntu 12.04

Na Tabela 9 são apresentadas as configurações utilizadas. É importante destacar que as VMs possuem o mesmo propósito e preços diferentes. Como apenas a quantidade de memória não é igual, esse fator é fixado pelos provedores e não há a possibilidade de modificação, os experimentos consideraram cargas de trabalho orientadas a sistema e à CPU. As cargas orientadas a disco não foram consideradas pela limitação do protótipo e pela ausência de informação por parte dos provedores.

Os *benchmarks* utilizados neste estudo foram o N-queens e o Apache, sendo as variáveis de resposta o **tempo de resposta** e a **quantidade de requisições atendidas**. Considerando os tipos de VMs utilizadas e os *benchmarks* utilizados, foi gerado um conjunto de 12 experimentos com a variação do número de VMs e tipo de infraestrutura. Dessa forma, a Figura 15 apresenta os resultados obtidos na execução da carga de trabalho N-queens.

Figura 15 – Desempenho obtido com a variação de infraestrutura e número de VMs para a carga de trabalho N-queens

Fonte: Dados da pesquisa.

Há diferenças entre as abordagens utilizadas, como apresentado na Figura 15. Neste primeiro momento é importante considerar que a pilha de componentes dos provedores e da solução local não foi considerada. Como a avaliação é no desempenho da VM e não na

comunicação entre os clientes de infraestrutura, o sistema local foi o que obteve os melhores resultados.

Nesse ponto é importante considerar que as proporções de ganho de desempenho são mantidas entre as infraestruturas. Como a carga de trabalho utilizada é orientada a CPU, os resultados demonstram que há diferenças de *hardware* entre as três abordagens. Essa afirmação pode ser feita, pois todas as VMs possuem número igual de vCPU, tendo diferenciação apenas na quantidade de memória, que não influencia tanto nos resultados obtidos. Na Figura 16 são apresentados os comportamentos médios de cada fator.

Figura 16 – Médias de tempos de resposta observadas nos fatores para a carga N-queens

Fonte: Dados da pesquisa.

De acordo com a Figura 16, o melhor desempenho é obtido pela infraestrutura local. No entanto, o fato a ser considerado é que houve diferenças entre os provedores de nuvem, sendo o provedor Amazon EC2 o que obteve os melhores tempos de resposta. Os resultados obtidos pela carga de trabalho Apache são explorados na Figura 17.

Figura 17 – Desempenho obtido com a carga de trabalho Apache

Fonte: Dados da pesquisa.

Na Figura 17 é possível notar que o comportamento geral é mantido, e que a solução local acompanhou a tendência apresentada pelos provedores de nuvem. Há algumas características a serem consideradas nesse resultado. A primeira é que nesse caso a VM da Microsoft obteve um desempenho melhor que o a VM da Amazon em 2.93%. Já para as VMs com mais recursos, a diferença foi de 9,94% sendo a VM da Amazon melhor que a da Microsoft. No gráfico da Figura 18 é possível verificar, apesar de haver diferenças, qual provedor oferece o melhor desempenho.

Figura 18 – Média de requisições atendidas nas variações entre capacidade de Vm e tipos de provedores

Fonte: Dados da pesquisa.

De acordo com a Figura 18, apesar de haver um caso que o Azure foi melhor, no geral o EC2 foi melhor, considerando o comportamento geral dos resultados obtidos. A partir da VM é

possível obter algumas informações sobre o hospedeiro de cada VM. As informações coletadas estão dispostas na Tabela 10.

Tabela 10 – Informação da infraestrutura utilizada

Ambiente	Processador	Memória	Virtualizador
Local	Intel Core i7 870 2.93 GHz	16 GB DDR3	Xen 4.1-amd64
Azure	AMD Opteron 4171 2.09GHz	-	Microsoft Hyper-V Server
EC2	Intel Xeon E5-2680 v2 2.80GHz	-	Xen 4.2.amazon

Fonte: Dados da pesquisa.

A partir dos dados da Tabela 10, é possível fazer uma relação melhor de custo benefício. Definir apenas a quantidade de vCPU não garante de fato como será o comportamento de uma VM. É necessário conhecer também o *hardware* que hospeda esses serviços e quais *softwares* fazem o gerenciamento desses recursos.

O protótipo desenvolvido obteve resultados próximos aos dos obtidos nos provedores de nuvem, validando assim o seu uso. Outro ponto a ser destacado é a presença de mais de um tipo de virtualizador. Esse fato será explorado com mais atenção, pois é possível que um mesmo conjunto de *hardware* tenha seu desempenho melhorado com o uso de um *software* específico.

5.4 Comparação entre virtualizadores

Como já apresentado, há grande variação nas características de *hardware* entre os provedores. Outra característica que pode prover diferenças de desempenho é o uso de tipos específicos de virtualizadores. Neste conjunto de experimentos foram considerados três virtualizadores distintos. Os três virtualizadores possuem arquiteturas diferentes, como já discutido no Capítulo sobre computação em nuvem.

O objetivo deste conjunto de experimentos é demonstrar que mesmo utilizando o mesmo conjunto de *hardware* é possível obter desempenho diferente, variando apenas o *software* de gerenciamento de VMs. Os experimentos foram realizados em apenas um nó devido às características não determinísticas da rede. Sendo assim, características como migração de máquinas não foram contempladas. O foco dos resultados é a influência causada na qualidade com que esses serviços são executados nas VMs. As combinações utilizadas e a configuração das VMs é a apresentada na Tabela 11.

Tabela 11 – Planejamento de experimentos sobre diferenças de virtualizadores

Fatores			
Virtualizador	vCPU	Memória (GB)	Número de VMs
Xen	2 e 4	3,5 e 7	1 e 2
KVM	2 e 4	3,5 e 7	1 e 2
OpenVz	2 e 4	3,5 e 7	1 e 2

A definição dos níveis apresentados na Tabela 11 é relacionada ao tamanho de VMs encontradas nos provedores de IaaS. No protótipo utilizado é possível variar as características de vCPU, memória e disco e também de virtualizadores. A definição da VM é realizada de acordo com a Figura 19.

+ Adicionar Instância

1 Configuração 2 Seleção de template 3 Oferta de Computação 4 Oferta de Disco 5 Afinidade 6 Rede 7 Revisar

Por favor revise a informação abaixo e confirme que sua instância virtual está correta antes de executá-la.

Nome (Opcional)	MinhaVm	
Adicionar ao grupo (Opcional)		
Zona	Zona1	Editar
Hipervisor	KVM	Editar
Template	Umini	Editar
Oferta de Computação	Small Instance	Editar
Oferta de Disco	Small	Editar
Grupos de Afinidade	(Nenhum)	Editar
Grupos de segurança	default	Editar

Anterior Cancelar Executar VM

Figura 19 – Definição das VMs

Fonte: Dados da pesquisa.

Como apresentado na Figura 19, há algumas configurações que devem ser feitas para desenvolver a VM. Entre elas estão a definição de onde a VM será alocada, o tipo de virtualizador (hipervisor), oferta de computação (onde são definidas as quantidades de memória e vCPU), tamanho do disco e grupos de segurança.

É importante destacar que como há uma grande variação de tipos de armazenamento (*Solid-State Drive* (SSD), *Hard Disk Drive* (HDD), *Serial Advanced Technology Attachment* (Sata), *Network-Attached Storage* (NAS), *Serial Attached SCSI* (SAS)) as cargas de entrada e saída não serão consideradas nos estudos apresentados.

As variáveis de resposta consideradas neste primeiro conjunto de experimentos são: **tempo de resposta** e **requisições por segundo**. As médias obtidas nesse conjunto de experimentos foram obtidas a partir da execução dos *benchmarks* N-queens e Apache.

Para obter a média e, conseqüentemente, os gráficos de erro (com 95% de confiabilidade),

os *benchmarks* foram executados de 1 até N vezes, onde 10 execuções se mostraram suficientes para demonstrar o comportamento do sistema considerado. Na Figura 20 são apresentados os resultados obtidos na execução da carga de trabalho N-queens.

Figura 20 – Comparação de desempenho para a carga de trabalho N-queens

Fonte: Dados da pesquisa.

Na Figura 20, o eixo X é relacionado à variação de níveis considerando assim a quantidade VMs, quantidade de vCPU e os virtualizadores. O eixo Y é relacionado à variável de resposta que é o tempo de resposta obtido em segundos. No gráfico é possível observar que os intervalos de confiança não se sobrepuseram demonstrando que há diferença entre os três virtualizadores utilizados. Nesses experimentos o melhor resultado é obtido pelo virtualizador com os menores tempos de resposta.

O comportamento dos fatores em relação à mudança de níveis e o desempenho geral dos fatores são explorados em mais detalhes na Figura 21.

Figura 21 – Desempenho geral obtido pelos fatores para a carga de trabalho N-queens

Fonte: Dados da pesquisa.

Na Figura 21 a linha que corta cada quadro é referente aos valores médios obtidos. Como os resultados considerados são relacionados ao tempo de resposta, quanto mais abaixo da linha média melhor o resultado.

O principal ponto nesses resultados é o fato de os tempos de resposta terem sofrido alteração com a variação do virtualizador. Os demais resultados apresentaram o comportamento esperado para esse tipo de carga de trabalho. Sendo que, com o aumento de recursos a tendência é que os tempos de resposta melhorem. Em contrapartida, quando a concorrência por recursos é aumentada, o tempo de resposta tende a piorar.

No segundo conjunto de experimentos é abordada uma carga de trabalho voltada para a utilização do sistema. A Figura 22 apresenta os resultados obtidos na execução da carga de trabalho Apache.

Figura 22 – Desempenho obtido pelas VMs na execução da carga de trabalho Apache

Fonte: Dados da pesquisa.

Na Figura 22, o eixo X refere-se aos fatores considerados e o eixo Y refere-se a quantidade de requisições atendidas por segundo. Para esses resultados, quanto maior a quantidade de requisições melhor.

Nesse conjunto de experimentos, que consideram a carga de sistema, o virtualizador OpenVz foi o que apresentou melhores desempenhos. O desempenho geral é demonstrado na Figura 23, que apresenta as variações nos níveis utilizados.

Figura 23 – Desempenho geral obtido nas VMs para a carga de trabalho Apache

Fonte: Dados da pesquisa.

Os resultados apresentados na Figura 23 demonstram que, para esse conjunto de experimentos, os melhores resultados foram obtidos pelo virtualizador OpenVz, seguido pelo

virtualizador Xen. É importante destacar que, houve uma mudança de desempenho com a mudança da carga de trabalho que foi aplicada nas VMs.

Mesmo com as mesmas configurações, houve diferença de desempenho entre as VMs, tanto na carga de trabalho orientada a CPU como na carga de trabalho orientada a sistema. Esse fato demonstra que é necessário haver uma relação entre o serviço que será alocada à VM e o tipo de *software* que irá gerenciar essa VM.

Os resultados apresentados demonstram um comportamento comum em relação ao aumento de VMs no hospedeiro, sendo que, para as duas cargas utilizadas, houve um decréscimo de desempenho. Essa característica pode ser determinante no fornecimento de serviços com qualidade.

5.5 Comparação de desempenho com relação ao aumento de concorrência

Neste conjunto de experimentos o desempenho é avaliado quando há competição por recursos do hospedeiro em que as VMs são alocadas. A metodologia utilizada para sobrecarregar o sistema é o seguinte: uma VM é configurada com uma quantidade de vCPU, memória e disco e é imposta a ela uma carga de trabalho. Em cada experimento uma VM com as mesmas configurações é adicionada à máquina física. Além disso, a carga de trabalho da VM adicionada é semelhante à da máquina virtual já existente.

A configuração utilizada para a realização dos experimentos considera o número total de núcleos disponíveis - físicos e virtuais. A descrição dessas configurações permite compreender melhor os resultados obtidos. A Tabela 12 apresenta as configurações utilizadas nas máquinas virtuais.

Tabela 12 – Configuração das máquinas virtuais

Atributos	KVM
Memória	3,5GB
Processador	1 vCPU (2,93 GHz)
Disco	20GB

As informações contidas na Tabela 12 são iguais para todas as VMs que serão adicionadas ao hospedeiro. Além disso, é importante destacar que o hospedeiro possui recursos suficientes para alocar até 8 VMs com as configurações descritas.

As cargas de trabalho Apache, SmallPt e N-queens, foram utilizadas para sobrecarregar as VMs. A seleção das cargas de trabalho são relacionadas diretamente à capacidade de processamento do hospedeiro. Sendo assim, o foco principal dos resultados é na degradação de desempenho apresentado pela concorrência por uso do processador. O gráfico da Figura 24 apresenta o resultado obtido na execução da carga de trabalho SmallPt.

Figura 24 – Desempenho do conjunto de VMs utilizada

Fonte: Dados da pesquisa.

De acordo com a Figura 24, o aumento da quantidade de VMs no hospedeiro causa degradações de desempenho. Nesse caso, o melhor desempenho será obtido quando houver apenas uma VM no hospedeiro. No entanto, é preciso considerar o ambiente de nuvem como um ambiente compartilhado. Para esse conjunto de experimentos o ideal seria considerar o uso de duas ou quatro VMs, aumentando assim a utilização do hospedeiro.

Se o desempenho for considerado como atributo de QoS pelo provedor, nem sempre a maior quantidade de VMs por hospedeiro será alcançada. Isso porque a diferença de desempenho entre quatro VMs e oito VMs fica em torno de 49%. Se considerar a diferença de desempenho entre 8 VMs e apenas uma VM (desempenho ideal) a diferença fica em torno de 69.9%.

Apesar de a virtualização permitir o compartilhamento de recursos, esse compartilhamento deve ser bem estudado para que os clientes não sofram interferências no desempenho de suas VMs. No gráfico da Figura 25 são explorados os resultados obtidos na execução da carga de trabalho N-queens.

Figura 25 – Tempo de resposta obtido com o aumento de concorrência no hospedeiro

Fonte: Dados da pesquisa.

No gráfico da Figura 25 é possível observar que, assim como na carga de trabalho SmallPt, as diferenças de desempenho são mais acentuadas quando o limite de núcleos físicos é excedido. Conforme é aumentado o número de VMs o desempenho vai diminuindo.

Para esse conjunto de experimentos, a diferença entre quatro VMs e oito VMs ficou em torno de 31%. Já o desempenho entre oito VMs e o desempenho ideal (uma VM) ficou em torno de 48%. Esse fato demonstra que é necessário haver uma preocupação com a relação entre carga de trabalho e tipo e quantidade de VMs alocadas. Isso porque em dois tipos de cargas de trabalho orientadas à CPU os resultados obtidos foram diferentes.

A tendência observada é a mesma, uma perda de desempenho mais acentuada quando o limite de núcleos físicos é superado. No entanto, as porcentagens de perda de desempenho são diferentes. A fim de verificar outros tipos de cargas, uma carga de sistema foi utilizada para verificar as tendências de perda de desempenho. Na Figura 26 são apresentados os resultados obtidos na carga de trabalho Apache.

Figura 26 – Requisições por segundo obtidas com o aumento de concorrência no hospedeiro

Fonte: Dados da pesquisa.

Nos resultados apresentados, na Figura 26, é possível verificar que o compartilhamento de recursos faz com que o número de VMs por hospedeiro seja diminuído, isso se for considerada a QoS para os clientes. Essa QoS, com base em desempenho, pode ser pensada como a menor perda no ambiente compartilhado. Sendo a diferença de desempenho entre uma VM e duas VMs em torno de 8%. Aumentando o número de VMs para quatro, o desempenho fica em torno de 53% pior.

Esse comportamento demonstra que para prover QoS, em cargas de sistema, é necessário alocar um número reduzido de VMs por hospedeiro. Os resultados apresentados nas cargas de trabalho SmallPt, N-queens e Apache demonstram que deve haver uma preocupação também com a forma como as VMs são alocadas.

Além da quantidade de VMs por hospedeiro, é necessário levar em consideração que as VMs podem ter configurações diferentes (maior quantidade de recursos disponibilizados) sendo que essas configurações podem influenciar no grau de concorrência dos recursos alocados, interferindo assim na QoS dos serviços.

5.6 Planejamento de capacidade

Como o desempenho é diretamente relacionado aos recursos que são fornecidos à VM, neste estudo são utilizadas cargas de trabalho que utilizam uso intensivo de CPU, para verificar como é o comportamento dos tempos de resposta com a concorrência por recursos. O planejamento de experimentos é apresentado na Tabela 13.

Tabela 13 – Configurações dos experimentos realizados

Exp	VM	vCPUs
1	1	1
2	2	1
3	1	4
4	2	4
5	1	8
6	2	8

Como apresentado na Tabela 13, os experimentos consideram o limite de núcleos do servidor e as configurações de disco e de memória são as mesmas utilizadas na comparação de concorrência no hospedeiro. Dessa forma, há a necessidade de verificar os ganhos de desempenho com o aumento de núcleos, visando identificar as melhores configurações de desempenho e quais os níveis de sobrecarga obtidos.

Os experimentos foram elaborados para verificar qual o desempenho obtido quando há concorrência de máquinas virtuais em um mesmo servidor. Esse tipo de experimento é necessário para verificar como o planejamento de capacidade pode ser realizado em servidores com configurações semelhantes às utilizadas neste trabalho de doutorado.

Normalmente utiliza-se a quantidade de recursos físicos pela quantidade de serviços em execução, dado por P/N . O valor de P representa a quantidade de recursos disponíveis e o valor de N representa a quantidade de serviços em execução. É necessário verificar que a carga a ser representada por N deve ser a quantidade de VMs, pois elas devem fornecer aos seus clientes sua capacidade plena de processamento. A Figura 27 apresenta o desempenho obtido nas máquinas virtuais com a carga de trabalho SmallPt.

Figura 27 – Carga SmallPt

Fonte: Dados da pesquisa.

A Figura 27 possibilita a comparação de proporcionalidade oferecida pela relação de recursos por serviços. O mesmo comportamento é observado na carga de trabalho N-queens, apresentada na Figura 28.

Figura 28 – Carga N-queens

Fonte: Dados da pesquisa.

De acordo com as Figuras 27 e 28, a proporcionalidade sugerida por P/N não é obtida. Esse fato é observado quando o limite de VMs vai de uma para duas VMs com apenas uma vCPU, observa-se uma pequena alteração no desempenho. Quando o limite de recursos é aumentado, o desempenho melhora. No entanto, como a concorrência é maior, as perdas de desempenho quando uma nova VM é adicionada também é mais significativa. Na Figura 29 são apresentados os comportamentos médios de tempo de resposta para a carga de trabalho Smallpt.

Figura 29 – Carga SmallPt

Fonte: Dados da pesquisa.

Na Figura 30 são apresentados os comportamentos médios de tempo de resposta obtidos pela carga *N-queens*.

Figura 30 – *N-queens*

Fonte: Dados da pesquisa.

Nas Figuras 29 e 30 é possível observar que o comportamento foi semelhante para as duas cargas de trabalho. No entanto, o desempenho apresentado para cada carga de trabalho foi diferente. Esses estudos demonstram a importância de considerar a infraestrutura no fornecimento de serviços com qualidade. Nesses experimentos foi considerada apenas a presença da mesma carga de trabalho na VM. Outro ponto de vista é a alocação de VMs com cargas de trabalho distintas. Essa forma de alocação pode conduzir a uma maior utilização do hospedeiro sem

influenciar no desempenho final fornecido aos clientes.

5.7 Alocação de máquinas virtuais com base na carga de trabalho

Esse conjunto de experimentos tem a finalidade de verificar a influência no desempenho quando cargas de trabalho distintas são consideradas nas VMs. A metodologia proposta para esse conjunto de experimentos é a seguinte: uma VM é selecionada com uma carga de trabalho específica para ser o objeto de estudo e outras VMs são adicionadas ao hospedeiro. As VMs adicionadas possuem cargas de trabalho diferente da VM que é alvo do estudo. Nos experimentos, as VMs possuem 2 vCPUs, 3,5GB de memória e 30GB de disco. Na Figura 31 são apresentados os resultados obtidos.

Figura 31 – Comparação de desempenho entre as VMs.

Fonte: Dados da pesquisa.

Na Figura 31, o eixo Y representa as solicitações por segundo atendida pela VM que é observada como parâmetro. O eixo X representa a presença ou ausência de uma máquina virtual com carga específica (Apache, SmallPt ou Ramspeed).

Os melhores resultados são obtidos quando há a presença de uma VM com a carga SmallPt e a VM alvo do estudo. O segundo melhor desempenho é apresentado pela presença das cargas Ramspeed e SmallPt e a VM alvo. Esses resultados demonstram que é possível obter bons níveis de desempenho mesmo aumentando a utilização do hospedeiro.

Os resultados apontam que cada carga de trabalho emprega certa quantidade de degradação de desempenho, que pode ser maior ou menor. Na Figura 32 essa característica é explorada.

Figura 32 – Influência das cargas de trabalho no desempenho da VM

Fonte: Dados da pesquisa.

Na Figura 32 o eixo X representa a influência que cada fator teve nos resultados e a linha identificada com o valor 2 identifica se o fator ou a combinação de fatores teve alguma influência no desempenho da VM. Os fatores mais influentes foram os fatores B e C que representam as cargas Apache e RamSpeed.

É importante observar que a carga de trabalho semelhante à carga utilizada na VM alvo foi a que mais influenciou nos resultados. Visando à maximização da utilização do hospedeiro foi sugerido um conjunto de experimentos que considera uma combinação diferente de cargas de trabalho usando a capacidade máxima do hospedeiro; essa combinação é apresentada na Tabela 14.

Tabela 14 – Combinação de cargas de trabalho

Exp	VM1	VM2	VM3	VM4
1	Apache	Apache	SmallPT	RamSpeed
2	Apache	SmallPT	SmallPT	RamSpeed
3	Apache	Apache	Apache	RamSpeed
4	Apache	RamSpeed	RamSpeed	Apache
5	Apache	RamSpeed	RamSpeed	SmallPt
6	Apache	Apache	Apache	SmallPt
7	Apache	SmallPT	SmallPT	SmallPt
8	Apache	RamSpeed	RamSpeed	RamSpeed
9	Apache	Apache	Apache	Apache
10	Apache	Apache	SmallPT	SmallPt
11	Apache	-	-	-

O experimento 11, apresentado na Tabela 14 é utilizado como base de comparação quando não há nenhuma outra VM concorrendo por recursos, ou seja, é o melhor desempenho

que pode ser obtido. Nos outros experimentos foram alocadas nas VMs 2, 3 e 4 as três cargas de trabalho adotadas considerando todas as combinações possíveis entre elas. O gráfico da Figura 33 apresenta os resultados obtidos e a Tabela 15 apresenta as médias e desvios padrão obtidos.

Figura 33 – Desempenho obtido na comparação com cargas de trabalho distintas

Tabela 15 – Médias e desvios dos resultados obtidos com relação às requisições atendidas

Exp.	Média	Desv. Padrão
1	3883,5	25,6
2	4103,7	47,7
3	4360,4	94,2
4	2889,8	78,4
5	2591,1	45,7
6	5303,7	85,5
7	5403	115,3
8	1770,7	22,7
9	4853,8	65,3
10	4954,2	31,5
11	7978,3	90,0

Em todos os experimentos apresentados na Figura 33, mesmo que a configuração da VM1 seja mantida, observa-se uma degradação no desempenho da carga de trabalho executada nessa VM. Essa degradação depende da carga de trabalho imposta às VMs 2, 3 e 4. Essa característica mostra que é necessário considerar também o tipo de carga de trabalho que será atribuída às VMs e não apenas se as VMs estão ou não ociosas.

Os resultados menos comprometidos são obtidos quando há pouca carga de memória e pouca carga de sistema. Quando o SmallPt está presente nas VMs, a concorrência observada é menor que para as outras cargas de trabalho. Por outro lado, a carga de trabalho Ramspeed (memória) apresenta a maior influência no desempenho da aplicação executando na VM1.

Nos resultados foi demonstrado que o desempenho da VM é afetado de acordo com a concorrência pelos recursos disponíveis. Com isso a QoS fornecida será afetada, dependendo de como os recursos são alocados. A utilização de modelos que consideram as características não só de recursos mas também de cargas de trabalho se torna essencial.

5.8 Considerações finais

Neste capítulo foram apresentados os resultados obtidos nesta pesquisa. O ponto principal é como as informações de *software* e *hardware* podem ser determinantes no fornecimento de QoS aos clientes. Para demonstrar essa relação, um estudo empírico sobre as informações disponibilizadas pelos provedores foi realizado. Nesse estudo foi demonstrado que pouca informação é disponibilizada aos clientes.

A relação das informações de *hardware* e *software* com o tipo de VM e a quantidade de recursos que essa VM irá possuir pode conduzir o cliente à contratação de serviços que atendam as suas necessidades. Uma questão que deve ser levada em consideração é que o cliente paga pelos recursos que utiliza. Nesse contexto, apresentar apenas as características de vCPU, memória e disco que uma VM irá possuir não garante que essa VM cumpra as expectativas do cliente.

Nos estudos experimentais entre os provedores de nuvem foi demonstrado que duas VMs, uma na Amazon e uma na Microsoft Azure, que possuem o mesmo número de recursos possuem desempenho diferente e custo semelhante. Com isso, pode-se afirmar que a relação de custo e recursos não é bem estabelecida na nuvem.

Outro ponto observado é a utilização de apenas um tipo de virtualizador por provedor. Um estudo foi realizado para verificar se há diferenças entre a utilização de um virtualizador ou outro. Esse estudo foi realizado em um ambiente controlado e demonstrou que, dependendo do tipo do serviço, o mesmo pode ter seu desempenho melhorado com a utilização de um virtualizador ao invés de outro.

Para a QoS foi considerada que, quanto mais desempenho a VM tiver, melhor. Nesse sentido, além do *software* de virtualização, a quantidade de VMs alocadas no hospedeiro e os tipos de cargas de trabalho nas VMs também foi considerado.

Nos resultados foi demonstrado que há influência de desempenho entre as VMs alocadas no mesmo hospedeiro. Essas influências contribuíram no desenvolvimento de um modelo de alocação com base em descontos e um modelo que considera a alocação de VMs com cargas de trabalho distintas no mesmo hospedeiro, a fim de manter o melhor desempenho para as VMs.

CONCLUSÕES

6.1 Considerações iniciais

A computação em nuvem trouxe grandes mudanças no fornecimento de serviços de computação. Essa tecnologia possui um grande potencial por ser de fácil acesso, por meio da Internet, e possibilitar a economia monetária, provendo o “aluguel” de recursos computacionais cobrando apenas por aqueles que forem utilizados. Nesta pesquisa foram explorados os conceitos de qualidade de serviço e serviços de infraestrutura de nuvem. Ao longo deste documento foram apresentados conceitos isolados sobre os dois temas principais e conceitos que fazem a ligação entre os dois.

A nuvem se torna mais atraente quando apresenta aspectos de QoS, pois promove maior confiança por parte de seus utilizadores que podem contar com serviços, como o próprio nome sugere, de qualidade. Há várias preocupações a respeito da adoção ou não da nuvem e talvez a principal seja a questão econômica e de segurança. Nesta pesquisa, apenas os aspectos relacionados às questões econômicas foram levados em consideração. Serviços ou modelos computacionais que envolvem dinheiro devem ser acompanhados de qualidade, pois, se um serviço tem um preço, o mínimo esperado é que ele cumpra ou faça valer o preço pago por ele. Nesta pesquisa os serviços de infraestrutura foram considerados e, com base nos resultados obtidos, foi possível comprovar que a infraestrutura física é determinante no fornecimento de serviços com desempenho.

6.2 Conclusões

Com a elaboração desta tese foi possível comprovar que, atualmente, faltam mecanismos que auxiliam na seleção de serviços de infraestrutura com base em custo-benefício. Uma boa prática que pode ser utilizada pelos provedores é o fornecimento de informações sobre o *hardware*

e o *software* utilizado para fazer o gerenciamento das VMs.

O levantamento entre os provedores de nuvem demonstrou que a maioria dos preços é semelhante e os tipos de VMs também são semelhantes, como número de vCPU, quantidade de memória e disco. No entanto, foi demonstrado que cada provedor possui um *hardware* diferente, gerando um desempenho diferente para cada tipo de VM. Além disso, cada provedor utiliza apenas um tipo de virtualizador, forçando o cliente a usar apenas um tipo de solução.

Na verificação do custo cobrado por VM, os provedores Amazon EC2 e Microsoft Azure foram avaliados utilizando tipos de VMs semelhantes. Os resultados demonstraram que para a carga de trabalho orientada à CPU, a diferença da VM1 foi de 9,57% e da VM2 foi de 11,70%. Para a carga de trabalho de sistema, a diferença entre a VM1 foi de 2,93% e da VM2 foi de 9,94%. Ou seja, dois provedores vendendo VMs similares, porém, com diferenças de desempenho.

Esse comportamento reforça a necessidade de deixar as informações de *hardware* e *software* mais evidentes, facilitando assim a tomada de decisão. Para o problema de custo da VM foram considerados apenas dois provedores, ou seja, há a possibilidade de haver diferenças ainda maiores. Como a parte financeira está ligada diretamente à escolha do provedor, essas considerações se tornam interessantes na seleção do provedor mais adequado, considerando o tipo de carga de trabalho que será utilizada e as preocupações com a qualidade com que os serviços serão prestados.

Para os estudos sobre virtualizadores, foi demonstrado que, para um mesmo conjunto de *hardware*, o desempenho sofreu alterações de acordo com o *software* de virtualização. Nos resultados, o virtualizador Xen apresentou os melhores desempenhos na variação de máquinas virtuais e na variação de quantidade de núcleos por VM, sendo 33,85% melhor que o virtualizador OpenVz e 9,08% melhor que o virtualizador KVM para uma VM. Para duas VMs, o desempenho do virtualizador Xen foi 43,75% superior ao virtualizador OpenVz e 8,12% superior ao virtualizador KVM.

Essa preocupação com o virtualizador é importante, pois, a nuvem tem como tecnologia base a virtualização. Prover o maior número de possibilidades de virtualizadores, com base no tipo de serviço do cliente, pode ser determinante.

A alocação de VMs nos hospedeiros também foi considerada e mostrou que várias diferenças podem ser obtidas, dependendo do monitoramento e gerenciamento. No planejamento de capacidade foi demonstrada que, apesar de ser possível alocar mil recursos por uma hora, essa alocação nem sempre é viável devido à natureza financeira da nuvem. A alocação de máquinas virtuais com base em cargas de trabalho distintas também foi levada em consideração e demonstrou que, dependendo da forma como são alocadas, as máquinas virtuais podem afetar o desempenho de outras máquinas virtuais.

Conclui-se então que considerar o ambiente virtualizado e a forma como o mesmo é monitorado e mantido pode ser o diferencial no fornecimento ou não de serviços com qualidade.

A qualidade observada foi a capacidade das máquinas virtuais em utilizar o máximo dos recursos disponibilizados a ela, ou seja, se há um gasto por um tipo de recursos, espera-se que esse recurso seja utilizado ao máximo. Essas considerações foram realizadas considerando que o serviço de IaaS conta com o aluguel de infraestrutura e a mesma deve apresentar o desempenho que foi contratado pelo cliente.

6.3 Contribuições

Com o desenvolvimento deste projeto de doutorado várias contribuições foram alcançadas. Dentre essas contribuições estão:

- **Revisão dos trabalhos relacionados** - Vários trabalhos foram explorados nas áreas de computação em nuvem, virtualização e qualidade de serviço.
- **Virtualização** - Os conceitos de virtualização foram explorados e avaliações de desempenho foram utilizadas para demonstrar as diferenças obtidas entre as soluções de virtualização utilizadas. Além disso, a descrição da arquitetura dos virtualizadores utilizados foi apresentada, facilitando a compreensão dos mesmos.
- **Modelo de cobrança** - Foi demonstrado que, apesar de haver recursos suficientes, há a possibilidade de perda de desempenho por parte dos clientes, tornando necessário haver preocupações com as perdas de desempenho no custo final repassado aos clientes.
- **Planejamento de capacidade** - Um modelo de planejamento de capacidade foi proposto. Esse modelo leva em consideração a arquitetura atual dos processadores (*multicore*) e a ausência de uma forma concreta de definir a capacidade de um hospedeiro, ou seja, a quantidade de máquinas virtuais que podem ser alocadas simultaneamente. Neste modelo, apenas cargas do tipo CPU-bound foram consideradas por serem menos complexas de realizar generalizações.
- **Alocação de máquinas virtuais** Foi identificado que, apesar de a máquina física prover recursos suficientes para alocar uma quantidade de máquinas virtuais, a carga de trabalho que cada máquina virtual utiliza pode ter influência no desempenho das demais máquinas virtuais. Demonstrando assim que há a preocupação também em definir o melhor conjunto de cargas de trabalho que pode ser alocada de forma conjunta.

6.4 Publicações

As publicações obtidas no desenvolvimento deste doutorado foram:

- **LEITE, D. M.; PEIXOTO, M. L. M. ; SANTANA, M. J. ; SANTANA, R. H. C. . Performance Evaluation of Virtual Machine Monitors for Cloud Computing.** In: XIII

Simpósio em Sistemas Computacionais (WSCAD-SSC), 2012, Petrópolis RJ. XIII Simpósio em Sistemas Computacionais (WSCAD-SSC), 2012. p. 65-71.

- **LEITE, D. M.;** PEIXOTO, M. L. M. ; SANTANA, M. J. ; SANTANA, R. H. C. . **Avaliação de desempenho em ambiente computacional voltado para computação em nuvem com foco em aspectos de planejamento de capacidade;**. Revista RETEC, v. 4, p. 1-10, 2013.
- CANDIDO, P. ; **LEITE, D. M.;** SANTANA, M. J. ; SANTANA, R. H. C. ; PEIXOTO, M. L. M. . **Alocação Proativa de Recursos Virtualizados Aplicada à Computação em Nuvem.** In: XI Workshop de Computação em Clouds e Aplicações, 2013, 2013, Brasília. XXXI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos. Brasília, 2013. p. 104-115.
- FERREIRA, C. H. G. ; **LEITE FILHO, D. M.;** RIBEIRO, J. B. ; BRANQUINHO JUNIOR, W. D. ; ESTRELLA, J. C. ; PEIXOTO, M. L. M. . **Identificação de gargalos de desempenho em ambientes virtuais para uso em computação em nuvem.** In: XXXIV Congresso da Sociedade Brasileira de Computação, 2014, Brasília - DF. WPerformance - XIII Workshop em Desempenho de Sistemas Computacionais e de Comunicação, 2014. v. XIII. p. 1975-1988.

Além dos artigos publicados, um artigo foi aceito para publicação e outros três artigos estão em processo de revisão, que são:

- Bruno Guazzelli Batista, Julio Cezar Estrella, Carlos Henrique Gomes Ferreira, **Dionísio Machado Leite Filho**, Luis Hideo Vasconcelos Nakamura, StephanReiff-Marganiec, Marcos Jose Santana, Regina Helena Carlucci Santana. **Performance Evaluation of Resource Management in CloudComputing Environments.** Plos One journal, 2015.p. 1 - 16. **Aceito para publicação.**
- **A cloud computing price model based on virtual machine performance degradation,** Journal of Cloud Computing: Advances, Systems and Applications.
- **The use of knapsack problem to solve resource allocation in cloud computing by considering virtual machine workloads,** Journal of Network and Systems Management.
- **Understanding black box services in cloud computing providers,** Network Operations and Management Symposium (NOMS).

6.5 Dificuldades encontradas

No desenvolvimento desta pesquisa várias dificuldades foram encontradas. A primeira foi a definição do ambiente para realizar os experimentos. Apesar de haver provedores de

nuvem e *testbeds* de nuvem, nenhum deles permite a troca de virtualizadores. Os provedores atualmente utilizam ou um ou outro. Esse problema inviabilizou os experimentos considerando os virtualizadores na nuvem. Além disso, as instâncias atuais são fixas e no desenvolvimento do projeto algumas modificações nas instâncias foram necessárias para definir o grau de influência de componentes como memória e processador.

Outra dificuldade foi em encontrar informações técnicas sobre os provedores de serviços de infraestrutura. Nem sempre a informação é fornecida de forma clara. Outro problema é na contratação de serviços, muitas vezes os cartões de crédito não são aceitos ou não há a presença do campo de código de segurança o que leva a necessidade de entrar em contato com a operadora de cartão para liberar uma operação que deveria ser realizada de forma mais fácil possível.

A ausência de material prático sobre virtualização também foi uma dificuldade. Em artigos são descritos o funcionamento, as características e outras informações. A parte prática não possui tanta documentação, sendo que a maior parte das informações é encontrada em wikis e nem sempre possuem manuais para todas as distribuições, sendo necessário utilizar a abordagem de tentativa e erro.

Os *middlewares* de nuvem também foram uma dificuldade e que demandaram uma boa quantidade de tempo. Assim como a virtualização, os *middlewares* são bastante descritos em artigos, no entanto, na parte prática ainda há necessidade de documentação. A curva de aprendizado para a utilização de uma ferramenta de nuvem é muito alta. O OpenStack, por exemplo, que era o *middleware* que estava sendo utilizado como protótipo, possui suporte a vários virtualizadores. No entanto, apenas o KVM vem com um manual bem descrito pelos desenvolvedores. A integração com os demais virtualizadores demanda tempo.

Criar um protótipo com características semelhantes às encontradas em data centers também é uma dificuldade devido, principalmente, aos custos dos equipamentos e a topologia que pode ser adotada. Por esse motivo este trabalho não utilizou as características de rede da nuvem, avaliando apenas os tipos de aplicações.

6.6 Trabalhos futuros

Como trabalhos futuros, é possível destacar como pontos importantes:

- Utilização de outras métricas de QoS, como a disponibilidade, reputação e segurança;
 - A utilização de cada atributo irá depender do foco que a aplicação terá e quais as suas necessidades. Uma vez que se trabalha com a qualidade do serviço, o objetivo desse serviço deve ser levado em consideração. Neste trabalho consideramos apenas as perdas de desempenho, mas poderia ser considerada a elasticidade e a relação da mesma com desempenho.

- Utilização de migração na alocação de máquinas virtuais e recomposição de desempenho;
 - Uma vez que é possível haver migração entre nós que hospedam as máquinas virtuais, é possível que essa atividade afete o desempenho e disponibilidade das VMs.
- Incorporação de novas técnicas de avaliação e descoberta de informação, como o uso de mineração de dados;
 - Nesta tese foram utilizados experimentos controlados, o objetivo de se ter um protótipo é a possibilidade de explorar característica sobre o funcionamento da nuvem que nem sempre podem ser controlados. Como a parte relacionada a tolerância a falhas, identificar como elas ocorrem e a possibilidade de desenvolver mecanismos de reocupação se tornam importantes.
- Elaboração de uma arquitetura de teste em nuvem, assim como já acontece com outros modelos como SOA;
 - A disponibilização para outros grupos de pesquisa de infraestrutura pode auxiliar no estudo de fenômenos que não são facilmente percebidos por quem controla a infraestrutura. Dessa forma, é interessante que outras pessoas possam fazer manipulações com a finalidade de contribuir para o desenvolvimento da infraestrutura.
- Utilização de técnicas estatísticas ou de inteligência artificial para a alocação e distribuição de cargas entre hospedeiros;
 - O estudo de cargas de trabalho e como modelar as mesmas ainda é um ponto em aberto na computação em nuvem. Observando o ambiente de nuvem em funcionamento pode facilitar essa tarefa.
- Estudo sobre como adotar a nuvem para HPC *High Performance Computing*
 - Outro ponto que ainda esta em aberto é a utilização de infraestruturas de nuvem para prover serviços de computação de alto desempenho. Questões como disponibilidade e interferência de desempenho são pontos que podem ser determinantes para a total adoção da nuvem para esse tipo de serviço.

REFERÊNCIAS

ABDELMABOUD, A.; JAWAWI, D. N.; GHANI, I.; ELSAFI, A.; KITCHENHAM, B. Quality of service approaches in cloud computing: A systematic mapping study. **Journal of Systems and Software**, Elsevier, v. 101, p. 159–179, 2015. Citado 2 vezes nas páginas 29 e 51.

AHMED, A.; SABYASACHI, A. S. Cloud computing simulators: A detailed survey and future direction. In: IEEE. **Advance Computing Conference (IACC), 2014 IEEE International**. [S.l.], 2014. p. 866–872. Citado na página 42.

AL-AYYOUB, M.; JARARWEH, Y.; DARAGHMEH, M.; ALTHEBYAN, Q. Multi-agent based dynamic resource provisioning and monitoring for cloud computing systems infrastructure. **Cluster Computing**, Springer, v. 18, n. 2, p. 919–932, 2015. Citado 2 vezes nas páginas 24 e 25.

ALHAMAZANI, K.; RANJAN, R.; MITRA, K.; RABHI, F.; JAYARAMAN, P.; KHAN, S.; GUABTNI, A.; BHATNAGAR, V. An overview of the commercial cloud monitoring tools: research dimensions, design issues, and state-of-the-art. **Computing**, Springer Vienna, v. 96, p. 1–21, 2014. ISSN 0010-485X. Disponível em: <<http://dx.doi.org/10.1007/s00607-014-0398-5>>. Citado 4 vezes nas páginas 24, 30, 54 e 56.

AMARANTE, S. R. M.; ROBERTO, F. M.; CARDOSO, A. R.; CELESTINO, J. Using the multiple knapsack problem to model the problem of virtual machine allocation in cloud computing. In: IEEE. **Computational Science and Engineering (CSE), 2013 IEEE 16th International Conference on**. [S.l.], 2013. p. 476–483. Citado 2 vezes nas páginas 53 e 56.

ARMBRUST, M.; FOX, A.; GRIFFITH, R.; JOSEPH, A. D.; KATZ, R. H.; KONWINSKI, A.; LEE, G.; PATTERSON, D. A.; RABKIN, A.; STOICA, I.; ZAHARIA, M. **Above the Clouds: A Berkeley View of Cloud Computing**. [S.l.], 2009. Disponível em: <<http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html>>. Citado 3 vezes nas páginas 23, 24 e 36.

ARMSTRONG, D.; DJEMAME, K. Performance issues in clouds: An evaluation of virtual image propagation and I/O paravirtualization. **The Computer Journal**, Br Computer Soc, v. 54, n. 6, p. 7, 2011. Citado 3 vezes nas páginas 34, 49 e 51.

ATAS, G.; GUNGOR, V. C. Performance evaluation of cloud computing platforms using statistical methods. **Computers & Electrical Engineering**, v. 40, n. 5, p. 1636–1649, 2014. ISSN 0045-7906. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0045790614000718>>. Citado na página 36.

BABU, L.; KRISHNA, P. V. Honey bee behavior inspired load balancing of tasks in cloud computing environments. **Applied Soft Computing**, Elsevier, p. 2292–2303, 2013. Citado na página 24.

BABU, S. A.; HAREESH, M.; MARTIN, J. P.; CHERIAN, S.; SASTRI, Y. System performance evaluation of para virtualization, container virtualization, and full virtualization using xen, openvz, and xenserver. In: IEEE. **Advances in Computing and Communications (ICACC)**,

2014 Fourth International Conference on. [S.l.], 2014. p. 247–250. Citado 4 vezes nas páginas 33, 34, 58 e 59.

BAUN, C.; KUNZE, M. Performance Measurement of a Private Cloud in the OpenCirrusTM Testbed. In: SPRINGER. **Euro-Par 2009–Parallel Processing Workshops.** [S.l.], 2010. p. 434–443. Citado na página 40.

BERNERS-LEE, T.; FISCHETTI, M.; BY-DERTOUZOS, M. L. F. **Weaving the Web: The original design and ultimate destiny of the World Wide Web by its inventor.** [S.l.]: Harpe-Information, 2000. Citado na página 23.

BRUNEO, D. A stochastic model to investigate data center performance and qos in iaas cloud computing systems. **Parallel and Distributed Systems, IEEE Transactions on,** IEEE, v. 25, n. 3, p. 560–569, 2014. Citado 2 vezes nas páginas 50 e 51.

BUYA, R.; RANJAN, R.; CALHEIROS, R. N. InterCloud: Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services. **Proceedings of the 10th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP 2010, Busan, South Korea, May 21-23), LNCS, Springer, Germany, 2010.,** IEEE Press, Leipzig, Germany, abs/1003.3920, 2010. Citado na página 30.

CALHEIROS, R.; RANJAN, R.; De Rose, C.; BUYA, R. Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services. **Arxiv preprint arXiv:0903.2525,** Wiley Press, v. 40, 2009. Citado na página 42.

CAROLAN, J.; GAEDE, S.; BATY, J.; BRUNETTE, G.; LICHT, A.; REMMELL, J.; TUCKER, L.; WEISE, J. Sun Microsystems - Introduction to Cloud Computing Architecture. In: **White paper 1st edition, pp.9-12, june 2009. Retrieved in December 07, 2009, from: <http://www.sun.com/featured-articles/CloudComputing.pdf>.** [S.l.: s.n.], 2009. Citado 3 vezes nas páginas 30, 34 e 38.

CARVALHO, F. L. de; BELLEZI, M. A. Avaliação de desempenho dos hypervisors xen e kvm utilizando administração simplificada através do libvirt. **Revista TIS,** v. 3, n. 1, 2014. Citado 4 vezes nas páginas 31, 32, 58 e 59.

CASALICCHIO, E.; SILVESTRI, L. Mechanisms for SLA provisioning in cloud-based service providers. **Computer Networks,** Elsevier, v. 57, n. 3, p. 795–810, 2013. Citado 7 vezes nas páginas 25, 34, 38, 52, 53, 56 e 62.

CEROTTI, D.; GRIBAUDO, M.; PIAZZOLLA, P.; SERAZZI, G. Matching performance objectives for open and closed workloads by consolidation and replication. **Annals of Operations Research,** Springer, p. 1–24, 2014. Citado 2 vezes nas páginas 54 e 56.

CHANG, V.; WALTERS, R. J.; WILLS, G. The development that leads to the Cloud Computing Business Framework. **International Journal of Information Management,** v. 33, n. 3, p. 524–538, 2013. ISSN 0268-4012. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S026840121300008X>>. Citado 3 vezes nas páginas 24, 46 e 47.

CHARD, K. **DRIVE: A Distributed Economic Meta-Scheduler for the Federation of Grid and Cloud Systems.** Wellington, Nova Zelândia: Tese de Doutorado - Victoria University of Wellington, 2011. Citado na página 48.

CHE, J.; HE, Q.; GAO, Q.; HUANG, D. Performance Measuring and Comparing of Virtual Machine Monitors. In: IEEE. **Embedded and Ubiquitous Computing, 2008. EUC'08. IEEE/IFIP International Conference on**. [S.l.], 2008. v. 2, p. 381–386. Citado 2 vezes nas páginas 57 e 59.

CHE, J.; YU, Y.; SHI, C.; LIN, W. A Synthetical Performance Evaluation of OpenVZ, Xen and KVM. In: IEEE. **Services Computing Conference (APSCC), 2010 IEEE Asia-Pacific**. [S.l.], 2010. p. 587–594. Citado 2 vezes nas páginas 30 e 33.

CHIERICI, A.; VERALDI, R. A quantitative comparison between xen and kvm. **Journal of Physics: Conference Series**, v. 219, n. 4, p. 042005, 2010. Disponível em: <<http://stacks.iop.org/1742-6596/219/i=4/a=042005>>. Citado 3 vezes nas páginas 31, 57 e 59.

CHIEU, T. C.; MOHINDRA, A.; KARVE, A. A.; SEGAL, A. Dynamic Scaling of Web Applications in a Virtualized Cloud Computing Environment. In: **ICEBE '09: Proceedings of the 2009 IEEE International Conference on e-Business Engineering**. Washington, DC, USA: IEEE Computer Society, 2009. p. 281–286. ISBN 978-0-7695-3842-6. Citado na página 24.

CORRADI, A.; FANELLI, M.; FOSCHINI, L. Vm consolidation: A real case based on openstack cloud. **Future Generation Computer Systems**, Elsevier, v. 32, p. 118–127, 2014. Citado 2 vezes nas páginas 25 e 41.

CRAWLEY, E.; NAIR, R.; RAJAGOPALAN, B.; SANDICK, H. A Framework for QoS-based Routing in the Internet. **RFC2386**, v. 37, 1998. Citado na página 47.

CROSBY, S.; BROWN, D. The virtualization reality. **Queue**, v. 4, n. 10, p. 34–41, 2006. Citado na página 32.

DEYE, M. M. O.; SLIMANI, Y.; SENE, M. Load Balancing Approach for QoS Management of Multi-instance Applications in Clouds. In: IEEE. **Cloud Computing and Big Data (CloudCom-Asia), 2013 International Conference on**. [S.l.], 2013. p. 119–126. Citado 2 vezes nas páginas 49 e 51.

DONG, W. E.; NAN, W.; XU, L. QoS-Oriented Monitoring Model of Cloud Computing Resources Availability. In: IEEE. **Computational and Information Sciences (ICCIS), 2013 Fifth International Conference on**. [S.l.], 2013. p. 1537–1540. Citado 3 vezes nas páginas 49, 50 e 51.

DUBEY, S.; AGRAWAL, S. Methods to Ensure Quality of Service in Cloud Computing Environment. **International Journal**, v. 3, n. 6, p. 406–411, 2013. Citado 2 vezes nas páginas 23 e 24.

FATEMA, K.; EMEAKAROHA, V. C.; HEALY, P. D.; MORRISON, J. P.; LYNN, T. A survey of Cloud monitoring tools: Taxonomy, capabilities and objectives. **Journal of Parallel and Distributed Computing**, Elsevier, v. 74, n. 10, p. 2918–2933, 2014. Citado 4 vezes nas páginas 38, 54, 56 e 62.

FITTKAU, F.; FREY, S.; HASSELBRING, W. Cloud user-centric enhancements of the simulator cloudsims to improve cloud deployment option analysis. In: **Service-Oriented and Cloud Computing**. [S.l.]: Springer, 2012. p. 200–207. Citado na página 42.

- FUERTES, W.; ZAMBRANO, P.; SÁNCHEZ, M.; SANTILLÁN, M.; VILLACÍS, C.; TOULKE-RIDIS, T.; TORRES, E. Repowering an open source firewall based on a quantitative evaluation. **International Journal of Computer Science and Network Security (IJCSNS)**, International Journal of Computer Science and Network Security, v. 14, n. 11, p. 118, 2014. Citado na página 64.
- FURHT, B. Cloud Computing Fundamentals. In: FURHT, B.; ESCALANTE, A. (Ed.). **Handbook of Cloud Computing**. Boston, MA: Springer US, 2010. cap. 1, p. 3–19. ISBN 978-1-4419-6523-3. Disponível em: <http://dx.doi.org/10.1007/978-1-4419-6524-0_1>. Citado 3 vezes nas páginas 34, 38 e 40.
- GANNON, D.; FAY, D.; GREEN, D.; TAKEDA, K.; YI, W. Science in the cloud: lessons from three years of research projects on microsoft azure. In: ACM. **Proceedings of the 5th ACM workshop on Scientific cloud computing**. [S.l.], 2014. p. 1–8. Citado na página 40.
- GARG, S.; VERSTEEG, S.; BUYYA, R. A framework for ranking of cloud computing services. **Future Generation Computer Systems**, v. 1, n. 1, p. 12, 2012. ISSN 0167-739X. Citado 3 vezes nas páginas 23, 24 e 48.
- GARG, S. K.; BUYYA, R. NetworkCloudSim: modelling parallel applications in cloud simulations. In: IEEE. **Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference on**. [S.l.], 2011. p. 105–113. Citado na página 42.
- GIENGER, P. **Hybrid methods: KVM**. 2011. <http://southbrain.com/south/virtualization/6-hybrid-methods-kvm.html>. Citado na página 32.
- GROSSMAN, R.; GU, Y.; SABALA, M.; BENNET, C.; SEIDMAN, J.; MAMBRATTI, J. The open cloud testbed: a wide area testbed for cloud computing utilizing high performance network services. **arXiv preprint arXiv:0907.4810**, 2009. Citado na página 40.
- GUSTEDT, J.; JEANNOT, E.; QUINSON, M. Experimental methodologies for large-scale systems: a survey. **Parallel Processing Letters**, World Scientific, v. 19, n. 03, p. 399–418, 2009. Citado 2 vezes nas páginas 45 e 47.
- HAN, H.; DEYUI, Q.; ZHENG, W.; BIN, F. A Qos Guided task Scheduling Model in cloud computing environment. In: IEEE. **Emerging Intelligent Data and Web Technologies (EIDWT), 2013 Fourth International Conference on**. [S.l.], 2013. p. 72–76. Citado na página 47.
- HAN, R.; GHANEM, M. M.; GUO, L.; GUO, Y.; OSMOND, M. Enabling cost-aware and adaptive elasticity of multi-tier cloud applications. **Future Generation Computer Systems**, Elsevier, v. 32, p. 82–98, 2014. Citado 5 vezes nas páginas 36, 37, 50, 51 e 62.
- JAIN, R. The art of computer systems performance analysis. **John Wiley & Sons Chichester**, v. 182, p. 274 – 281, 1991. Citado na página 72.
- JARARWEH, Y.; ALSHARA, Z.; JARRAH, M.; KHARBUTLI, M.; ALSALEH, M. Teachcloud: a cloud computing educational toolkit. In: **Proceedings of the 1st International IBM Cloud Academy Conference (ICA CON 2012)**, IBM, Research Triangle Park, NC, USA. [S.l.: s.n.], 2012. p. 1 – 16. Citado na página 42.
- JULA, A.; SUNDARARAJAN, E.; OTHMAN, Z. Cloud computing service composition: A systematic literature review. **Expert Systems with Applications**, Elsevier, v. 41, n. 8, p. 3809–3824, 2014. Citado 2 vezes nas páginas 50 e 51.

KANG, Z.; WANG, H. A Novel Approach to Allocate Cloud Resource with Different Performance Traits. In: IEEE. **Services Computing (SCC), 2013 IEEE International Conference on**. [S.l.], 2013. p. 128–135. Citado 4 vezes nas páginas 30, 34, 53 e 56.

KEAHEY, K.; FIGUEIREDO, R.; FORTES, J.; FREEMAN, T.; TSUGAWA, M. Science clouds: Early experiences in cloud computing for scientific applications. **Cloud computing and applications**, v. 2008, p. 825–830, 2008. Citado na página 40.

KESSACI, Y.; MELAB, N.; TALBI, E.-G. A multi-start local search heuristic for an energy efficient vms assignment on top of the opennebula cloud manager. **Future Generation Computer Systems**, Elsevier, v. 36, p. 237–256, 2014. Citado na página 41.

KIM, K.; BELOGLAZOV, A.; BUYYA, R. Power-aware provisioning of cloud resources for real-time services. In: ACM. **Proceedings of the 7th International Workshop on Middleware for Grids, Clouds and e-Science**. [S.l.], 2009. p. 1. Citado 2 vezes nas páginas 29 e 30.

KOKKINOS, P.; VARVARIGOU, T.; KRETSIS, A.; SOUMPLIS, P.; VARVARIGOS, E. Sumo: Analysis and optimization of amazon ec2 instances. **Journal of Grid Computing**, Springer, v. 13, n. 2, p. 255–274, 2014. Citado na página 40.

KOURTESIS, D.; RODRÍGUEZ, A.; MARÍA, J.; PARASKAKIS, I. Semantic-based QoS management in cloud systems: Current status and future challenges. **Future Generation Computer Systems**, Elsevier, v. 32, p. 307–323, 2014. Citado 2 vezes nas páginas 50 e 51.

KRISHNAN, S.; GONZALEZ, J. L. U. Google app engine. In: **Building Your Next Big Thing with Google Cloud Platform**. [S.l.]: Springer, 2015. p. 83–122. Citado na página 40.

KUMAR, K. A.; QUAMAR, A.; DESHPANDE, A.; KHULLER, S. SWORD: workload-aware data placement and replica selection for cloud data management systems. **The VLDB Journal—The International Journal on Very Large Data Bases**, Springer-Verlag New York, Inc., v. 23, n. 6, p. 845–870, 2014. Citado 2 vezes nas páginas 50 e 51.

KUMAR, N.; CHILAMKURTI, N.; ZEADALLY, S.; JEONG, Y.-S. Achieving Quality of Service (QoS) Using Resource Allocation and Adaptive Scheduling in Cloud Computing with Grid Support. **The Computer Journal**, v. 57, n. 2, p. 281 – 290, 2013. Disponível em: <<http://comjnl.oxfordjournals.org/content/early/2013/03/11/comjnl.bxt024.abstract>>. Citado 3 vezes nas páginas 24, 49 e 51.

KUMAR, R.; GUPTA, S. Open source infrastructure for cloud computing platform using eucalyptus. **Global Journal of Computers & Technology Vol**, v. 1, n. 2, 2014. Citado na página 41.

KUMAR, R.; JAIN, K.; MAHARWAL, H.; JAIN, N.; DADHICH, A. Apache cloudstack: Open source infrastructure as a service cloud computing platform. In: **International Journal of advancement in Engineering technology, Management and Applied Science**. [S.l.: s.n.], 2014. v. 1, n. 2, p. 111–116. Citado 3 vezes nas páginas 42, 63 e 64.

KVM. **Kernel-based Virtual Machine. Site oficial da ferramenta**. 2011. http://www.linux-kvm.org/page/Main_Page. Citado 2 vezes nas páginas 32 e 33.

LAMPE, U.; HANS, R.; SELIGER, M.; PAULY, M. Pricing in Infrastructure Clouds—An Analytical and Empirical Examination. p. 1 –10, 2014. Citado na página 34.

- LARABEL, M.; TIPPETT, M. **Phoronix test suite**. Website, 2011. Disponível em: phoronix-test-suite.com. Citado na página 73.
- LAURSEN, J. V. The internet: Past, present and future internet & www history. **URL: <http://www.vissing.dk/inthist.html>**. Accessed, v. 11, n. 27, p. 01, 1997. Citado na página 23.
- LEITE, D.; PEIXOTO, M.; SANTANA, M.; SANTANA, R. Performance Evaluation of Virtual Machine Monitors for Cloud Computing. In: IEEE. **Computer Systems (WSCAD-SSC), 2012 13th Symposium on**. [S.l.], 2012. p. 65–71. Citado 2 vezes nas páginas 58 e 59.
- LI, K.; ZHENG, H.; WU, J.; DU, X. Virtual machine placement in cloud systems through migration process. **International Journal of Parallel, Emergent and Distributed Systems**, v. 29, n. 0, p. 1–18, 2014. Citado 3 vezes nas páginas 30, 53 e 56.
- LI, Q.; HAO, Q.; XIAO, L.; LI, Z. An integrated approach to automatic management of virtualized resources in cloud environments. **The Computer Journal**, Br Computer Soc, v. 54, n. 6, p. 905–919, 2011. Citado 3 vezes nas páginas 30, 52 e 56.
- LI, Z.; O'BRIEN, L.; CAI, R.; ZHANG, H. Towards a taxonomy of performance evaluation of commercial Cloud services. In: IEEE. **Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on**. [S.l.], 2012. p. 344–351. Citado 3 vezes nas páginas 35, 46 e 47.
- LIANG, Q.; ZHANG, J.; ZHANG, Y.-h.; LIANG, J.-m. The placement method of resources and applications based on request prediction in cloud data center. **Information Sciences**, v. 279, n. 0, p. 735–745, 2014. ISSN 0020-0255. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0020025514004733>. Citado 2 vezes nas páginas 29 e 62.
- LIM, S.-H.; SHARMA, B.; NAM, G.; KIM, E. K.; DAS, C. R. MDCSim: A multi-tier data center simulation, platform. In: IEEE. **Cluster Computing and Workshops, 2009. CLUSTER'09. IEEE International Conference on**. [S.l.], 2009. p. 1–9. Citado na página 42.
- LIN, J.-W.; CHEN, C.-H.; LIN, C.-Y. Integrating QoS awareness with virtualization in cloud computing systems for delay-sensitive applications. **Future Generation Computer Systems**, Elsevier, v. 37, p. 478–487, 2014. Citado 3 vezes nas páginas 30, 53 e 56.
- LINNOLAHTI, J. QoS routing for P2P networking. In: **HUT T-110.551 Seminar on Inter-networking**. [s.n.], 2004. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.58.7192&am>. Citado 2 vezes nas páginas 47 e 48.
- LIU, D.; ZHANG, Y. Y.; ZHANG, N.; HU, K. A research on kvm-based virtualization security. In: TRANS TECH PUBL. **Applied Mechanics and Materials**. [S.l.], 2014. v. 543, p. 3126–3129. Citado 2 vezes nas páginas 32 e 33.
- LIU, L.; ZHANG, M.; LIN, Y.; QIN, L. A Survey on Workflow Management and Scheduling in Cloud Computing. In: IEEE. **Cluster, Cloud and Grid Computing (CCGrid), 2014 14th IEEE/ACM International Symposium on**. [S.l.], 2014. p. 837–846. Citado 4 vezes nas páginas 35, 50, 51 e 62.
- LIU, Z.; WANG, S.; SUN, Q.; ZOU, H.; YANG, F. Cost-Aware Cloud Service Request Scheduling for SaaS Providers. **The Computer Journal**, 2013. Disponível em: <http://comjnl.oxfordjournals.org/content/early/2013/02/05/comjnl.bxt009.abstract>. Citado 3 vezes nas páginas 24, 34 e 36.

LU, C.-T.; CHANG, C.-W.; LI, J.-S. Vm scaling based on hurst exponent and markov transition with empirical cloud data. **Journal of Systems and Software**, Elsevier, v. 99, p. 199–207, 2015. Citado 2 vezes nas páginas 30 e 52.

MACÍAS, M.; GUITART, J. {SLA} negotiation and enforcement policies for revenue maximization and client classification in cloud providers. **Future Generation Computer Systems**, v. 41, n. 0, p. 19–31, 2014. ISSN 0167-739X. Citado 3 vezes nas páginas 34, 50 e 51.

MANVI, S. S.; Krishna Shyam, G. Resource management for Infrastructure as a Service (IaaS) in cloud computing: A survey. **Journal of Network and Computer Applications**, Elsevier, v. 41, p. 424–440, 2014. Citado 5 vezes nas páginas 25, 30, 36, 54 e 56.

MARINOS, A.; BRISCOE, G. Community cloud computing. In: JAATUN, M.; ZHAO, G.; RONG, C. (Ed.). **Cloud Computing**. Springer Berlin Heidelberg, 2009, (Lecture Notes in Computer Science, v. 5931). p. 472–484. ISBN 978-3-642-10664-4. Disponível em: <http://dx.doi.org/10.1007/978-3-642-10665-1_43>. Citado na página 38.

MCEVOY, G.; MURY, A. R.; SCHULZE, B. An analysis of definition and placement of virtual machines for high performance applications on Clouds. **Concurrency and Computation: Practice and Experience**, Wiley Online Library, p. 26, 2014. Citado 3 vezes nas páginas 30, 55 e 56.

MELL, P.; GRANCE, T. **The NIST Definition of Cloud Computing**. Sp 800-145. [S.l.], 2011. Disponível em: <<http://www.csrc.nist.gov/groups/SNS/cloud-computing>>. Citado 6 vezes nas páginas 24, 29, 34, 35, 37 e 38.

MOUSSA, R. Massive Data Analytics in the Cloud: TPC-H Experience on Hadoop Clusters. **IJWA**, v. 4, n. 3, p. 113–133, 2012. Citado na página 41.

NADEEM, F.; QAISER, R. An early evaluation and comparison of three private cloud computing software platforms. **Journal of Computer Science and Technology**, Springer, v. 30, n. 3, p. 639–654, 2015. Citado 4 vezes nas páginas 31, 42, 58 e 59.

NAVAUX, P. O.; CARISSIMI, A.; ROLOFF, E.; DIENER, M. High performance computing in the cloud: Deployment, performance and cost efficiency. In: IEEE COMPUTER SOCIETY. **Proceedings of the 2012 IEEE 4th International Conference on Cloud Computing Technology and Science (CloudCom)**. [S.l.], 2012. p. 371–378. Citado 2 vezes nas páginas 52 e 56.

PARIKH, K.; HAWANNA, N.; PK, H.; IYENGAR, N. C. S. *et al.* Virtual machine allocation policy in cloud computing using cloudsim in java. **International Journal of Grid and Distributed Computing**, v. 8, n. 1, p. 145–158, 2015. Citado na página 42.

PARTHASARATHY, M.; SOHI, R. S. Salesforce automation and the adoption of technological innovations by salespeople: theory and implications. **Journal of Business & Industrial Marketing**, MCB UP Ltd, v. 12, n. 3/4, p. 196–208, 1997. Citado na página 23.

PASCAL, G. **Xen: Hypervisor-based virtualization**. 2011.

<http://southbrain.com/south/virtualization/3-xen-hypervisor-based-virtual.html>. Citado na página 31.

PASCUAL, J.; LORIDO-BOTRÁN, T.; MIGUEL-ALONSO, J.; LOZANO, J. Towards a Greener Cloud Infrastructure Management using Optimized Placement Policies. **Journal of Grid Computing**, Springer Netherlands, p. 1–15, 2014. ISSN 1570-7873. Disponível em: <<http://dx.doi.org/10.1007/s10723-014-9312-9>>. Citado 2 vezes nas páginas 36 e 37.

PATIL, S.; LILJA, D. J. Statistical methods for computer performance evaluation. **Wiley Interdisciplinary Reviews: Computational Statistics**, Wiley Online Library, v. 4, n. 1, p. 98–106, 2012. Citado 3 vezes nas páginas 45, 46 e 47.

QIAN, L.; LUO, Z.; DU, Y.; GUO, L. Cloud computing: An overview. **Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)**, v. 5931 LNCS, p. 626–631, 2009. Cited By 23. Disponível em: <<http://www.scopus.com/inward/record.url?eid=2-s2.0-71749087772&partnerID=40&md5=4d6dbef7c4fb5698590dee59cdc11ddb>>. Citado na página 23.

RAMEZANI, F.; LU, J.; HUSSAIN, F. An online fuzzy Decision Support System for Resource Management in cloud environments. In: IEEE. **IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), 2013 Joint**. [S.l.], 2013. p. 754–759. Citado 2 vezes nas páginas 49 e 51.

RANJAN, R.; ZHAO, L.; WU, X.; LIU, A.; QUIROZ, A.; PARASHAR, M. Peer-to-peer cloud provisioning: Service discovery and load-balancing. In: **Cloud Computing**. [S.l.]: Springer, 2010. p. 195–217. Citado na página 40.

REHMAN, Z.-u.; HUSSAIN, O. K.; HUSSAIN, F. K. User-side cloud service management: state-of-the-art and future directions. **Journal of Network and Computer Applications**, Elsevier, p. 108–122, 2015. Citado 2 vezes nas páginas 30 e 51.

RIMAL, B. P.; CHOI, E.; LUMB, I. A Taxonomy and Survey of Cloud Computing Systems. In: IEEE. **NCM '09: Proceedings of the 2009 Fifth International Joint Conference on INC, IMS and IDC**. Washington, DC, USA: IEEE Computer Society, 2009. p. 44–51. ISBN 978-0-7695-3769-6. Citado 2 vezes nas páginas 35 e 36.

SAKELLARI, G.; LOUKAS, G. A survey of mathematical models, simulation approaches and testbeds used for research in cloud computing. **Simulation Modelling Practice and Theory**, Elsevier, p. 12, 2013. Citado 2 vezes nas páginas 40 e 41.

SALAH, K.; AL-SABA, M.; AKHDHOR, M.; SHAABAN, O.; BUHARI, M. Performance evaluation of popular Cloud IaaS providers. In: IEEE. **Internet Technology and Secured Transactions (ICITST), 2011 International Conference for**. [S.l.], 2011. p. 345–349. Citado 2 vezes nas páginas 52 e 56.

SHEN, Z.; SUBBIAH, S.; GU, X.; WILKES, J. Cloudscale: elastic resource scaling for multi-tenant cloud systems. In: ACM. **Proceedings of the 2nd ACM Symposium on Cloud Computing**. [S.l.], 2011. p. 5. Citado na página 41.

SKADRON, K.; MARTONOSI, M.; AUGUST, D. I.; HILL, M. D.; LILJA, D. J.; PAI, V. S. Challenges in computer architecture evaluation. **Computer**, IEEE, v. 36, n. 8, p. 30–36, 2003. Citado 2 vezes nas páginas 45 e 47.

STANTCHEV, V.; SCHRÖPFER, C. Negotiating and Enforcing QoS and SLAs in Grid and Cloud Computing. In: **Proceedings of the 4th International Conference on Advances in Grid and Pervasive Computing**. Berlin, Heidelberg: Springer-Verlag, 2009. (GPC '09, v. 1), p. 25–35. ISBN 978-3-642-01670-7. Citado na página 23.

SUN, M.; GU, W.; ZHANG, X.; SHI, H.; ZHANG, W. A Matrix Transformation Algorithm for Virtual Machine Placement in Cloud. In: IEEE. **Trust, Security and Privacy in Computing and**

Communications (TrustCom), 2013 12th IEEE International Conference on. [S.l.], 2013. p. 1778–1783. Citado 2 vezes nas páginas 53 e 56.

TANG, S.; YUAN, J.; WANG, C.; LI, X.-Y. A framework for amazon ec2 bidding strategy under sla constraints. **Parallel and Distributed Systems, IEEE Transactions on**, IEEE, v. 25, n. 1, p. 2–11, 2014. Citado na página 40.

TUDORAN, R.; COSTAN, A.; ANTONIU, G.; BOUGÉ, L. A performance evaluation of Azure and Nimbus clouds for scientific applications. In: ACM. **Proceedings of the 2nd International Workshop on Cloud Computing Platforms.** [S.l.], 2012. p. 4. Citado na página 42.

VAQUERO, L. M.; RODERO-MERINO, L.; CACERES, J.; LINDNER, M. A break in the clouds: towards a cloud definition. **SIGCOMM Comput. Commun. Rev.**, ACM, New York, NY, USA, v. 39, p. 50–55, December 2008. ISSN 0146-4833. Citado na página 23.

WARD, J. S.; BARKER, A. Observing the clouds: a survey and taxonomy of cloud monitoring. **Journal of Cloud Computing: Advances, Systems and Applications**, Springer, v. 3, n. 1, p. 40, 2014. Citado 4 vezes nas páginas 24, 25, 29 e 62.

WEI, Y.; BLAKE, M. Proactive virtualized resource management for service workflows in the cloud. **Computing**, Springer Vienna, p. 1–16, 2014. Citado 3 vezes nas páginas 30, 55 e 56.

WEINGÄRTNER, R.; BRÄSCHER, G. B.; WESTPHALL, C. B. Cloud resource management: A survey on forecasting and profiling models. **Journal of Network and Computer Applications**, Elsevier, v. 47, p. 99–106, 2015. Citado 8 vezes nas páginas 23, 24, 25, 31, 34, 52, 55 e 56.

WHITE, J.; PILBEAM, A. A survey of virtualization technologies with performance testing. **Arxiv preprint arXiv:1010.3233**, 2010. Citado 2 vezes nas páginas 57 e 59.

WU, L.; CHEN, Z.; SU, J.; LUO, X. The based-role PMI model for access control in large scale network system. In: IEEE. **Computer Design and Applications (ICDDA), 2010 International Conference on.** [S.l.], 2010. v. 2, p. V2–81–V2–84. Citado na página 29.

XU, X.; ZHOU, F.; WAN, J.; JIANG, Y. Quantifying performance properties of virtual machine. In: IEEE. **Information Science and Engineering, 2008. ISISE'08. International Symposium on.** [S.l.], 2008. v. 1, p. 24–28. Citado 2 vezes nas páginas 57 e 59.

YOUNGE, A. J.; HENSCHER, R.; BROWN, J. T.; LASZEWSKI, G. von; QIU, J.; FOX, G. C. Analysis of virtualization technologies for high performance computing environments. In: IEEE. **Cloud Computing (CLOUD), 2011 IEEE International Conference on.** [S.l.], 2011. p. 9–16. Citado 2 vezes nas páginas 57 e 59.

YOUNGE, A. J.; WALTERS, J. P.; CRAGO, S.; FOX, G. C. Evaluating gpu passthrough in xen for high performance cloud computing. In: IEEE. **Parallel & Distributed Processing Symposium Workshops (IPDPSW), 2014 IEEE International.** [S.l.], 2014. p. 852–859. Citado 2 vezes nas páginas 31 e 32.

ZAMAN, S.; GROSU, D. Combinatorial auction-based allocation of virtual machine instances in clouds. **Journal of Parallel and Distributed Computing**, Elsevier, p. 495–508, 2013. Citado na página 24.

ZHANG, Q.; BOUTABA, R. Dynamic workload management in heterogeneous Cloud computing environments. In: IEEE. **Network Operations and Management Symposium (NOMS), 2014 IEEE.** [S.l.], 2014. p. 1–7. Citado 3 vezes nas páginas 30, 53 e 56.

ZHANG, Q.; CHENG, L.; BOUTABA, R. Cloud computing: state-of-the-art and research challenges. **Journal of Internet Services and Applications**, Springer-Verlag, v. 1, n. 1, p. 7–18, 2010. ISSN 1867-4828. Disponível em: <<http://dx.doi.org/10.1007/s13174-010-0007-6>>. Citado 6 vezes nas páginas 30, 35, 36, 38, 40 e 52.

ZHANG, S.; YAN, H.; CHEN, X. Research on Key Technologies of Cloud Computing. **Physics Procedia**, Elsevier, v. 33, n. 0, p. 1791 – 1797, 2012. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1875389212015994>>. Citado 2 vezes nas páginas 29 e 42.

ZHENG, Y.; NICOL, D. M. A virtual time system for openvz-based network emulations. In: IEEE. **Principles of Advanced and Distributed Simulation (PADS), 2011 IEEE Workshop on**. [S.l.], 2011. p. 1–10. Citado 2 vezes nas páginas 33 e 34.

ZHOU, C.; LI, B. ihac: A hybrid access control framework for iaas clouds. In: IEEE COMPUTER SOCIETY. **Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing**. [S.l.], 2014. p. 853–858. Citado na página 41.

ZHOU, X.; JIANG, C.-J. Autonomic Performance and Power Control on Virtualized Servers: Survey, Practices, and Trends. **Journal of Computer Science and Technology**, Springer, v. 29, n. 4, p. 631–645, 2014. Citado 4 vezes nas páginas 30, 31, 54 e 56.