Uma taxonomia para corretagem de nuvens IaaS baseada na migração de infraestruturas virtuais

Euclides Cardoso Junior ¹ Charles Christian Miers ¹ Guilherme Piegas Koslovski ¹

Resumo: A computação em nuvem propõe um modelo no qual o usuário tem a ideia de recursos ilimitados, que podem ser reservados dinamicamente, pagando somente o que for consumido. Esse modelo de tarifação é conhecido como *pay-as-you-go*. Estes e outros fatores, levaram a uma ampla adoção dos serviços oferecidos pelas nuvens computacionais por parte dos usuários. Porém, ao migrarem aplicações para a nuvem, os usuários se deparam com uma quantidade expressiva de provedores, recursos, corretores e serviços oferecidos. Através da análise de material coletado em uma pesquisa referenciada, este trabalho propõe uma taxonomia para a classificação de atores relacionados com a corretagem de nuvens de acordo com as suas principais características. Nas referências são citados diversos recursos e tipos de serviços, porém este trabalho mantém o foco na migração de infraestruturas virtuais entre provedores de *Infrastructure-as-a-Service* (IaaS).

Palavras-chave: Corretagem, IaaS, Nuvens, Corretores, Migração, Interoperabilidade

Abstract: Cloud computing offers a model in which the user has the idea of unlimited resources. Allied with this idea, cloud computing also provides another advantage that allows the user to pay only what is consumed, this model is also known as pay as you go. These and other factors led to a broad adoption of services offered by cloud computing. However, when the users migrates their applications to the cloud, they faced several possibilities of providers, resources, brokers, and services. Through a analysis of articles and books relevant to the subject, we propose a taxonomy in order to classify cloud brokers according to their main features. The references mention several features, but this work keeps focused in migration of virtual infrastructures between Infrastructure-as-a-Service (IaaS) providers.

Keywords: Brokerage, IaaS, Clouds, Brokers, Migration, Interoperability

1 Introdução

O desenvolvimento e a entrega de soluções e serviços computacionais foi sensivelmente alterado pelo paradigma de nuvens computacionais. Recursos de computação, armazenamento e comunicação são dinamicamente provisionados, permitindo a composição de serviços elásticos, aptos a atender a variação de demanda computacional [44]. Nesse cenário, provedores de infraestrutura baseados em *Infrastructure-as-a-Service* (IaaS) tipicamente disponibilizam recursos através da abstração de Máquinas Virtuais (MVs), enquanto clientes contratantes são tarifados conforme a capacidade virtual provisionada e o tempo total de reserva.

Provedores IaaS oferecem aos clientes a possibilidade de configuração e reserva conjunta de recursos computacionais e de comunicação. Assim, um cliente pode compor uma Infraestrutura Virtual (IV) especificamente configurada para hospedar suas aplicações. Em suma, uma IV é composta por um conjunto de MVs interconectadas por recursos virtuais de comunicação (*e.g.*, enlaces, comutadores e roteadores).

Internamente em um provedor, a orquestração dos recursos é realizada por mecanismos específicos, comumente otimizados para gerenciar os recursos de acordo com os serviços oferecidos pelo provedor. Sendo assim,

http://dx.doi.org/10.5335/rbca.v9i1.6445

¹Programa de Pós-Graduação em Computação Aplicada (PPGCA), Departamento de Ciência da Computação (DCC), Universidade do Estado de Santa Catarina (UDESC), Joinville/SC - Brasil euclides.c.jr@hotmail.com, charles.miers@udesc.br, guilherme.koslovski@udesc.br

um cliente contratante compõe uma IV, teoricamente independente da arquitetura física hospedeira, entretanto, otimizações aplicadas por provedores criam *âncoras* fixando a IV ao provedor. Exemplificando, as interfaces e regras para definição de balanceamento de carga, configurações de segurança e resolução de endereços, diferem consideravelmente entre os provedores.

Entre o cliente e o provedor, existe um importante ator no contexto de computação em nuvem, conhecido como corretor de nuvem (*cloud broker*). Instituições de renome como *Gartner* [59] e *National Institute of Standards and Technology* (NIST) [44] definiram os corretores de nuvem como atores de importância relevante em nuvens computacionais. Através destes corretores torna-se possível, por exemplo, agregar, integrar e até mesmo personalizar diversos serviços de diferentes provedores.

Sobretudo, corretores atuam na migração de IVs entre provedores. Em suma, a migração pode ser originada por motivos econômicos (redução de custos), qualidade do serviço hospedado (proximidade dos clientes do serviço), confiabilidade, entre outros. Embora conceitualmente essencial para os clientes, a migração de recursos virtuais entre provedores depende da padronização de modelos e mecanismos, que permitam tal interoperabilidade.

A partir de uma revisão sistemática realizada, constatou-se que diversos aspectos de computação em nuvem carecem de melhores classificações. Sobretudo, na revisão da literatura especializada, não foram encontrados trabalhos que abordassem os temas *Unidades e Mecanismos de Migração* e *Confidencialidade do Canal*. A revisão sistemática foi realizada utilizando o sistema de busca do Google, o portal da Capes e também as bases de dados científicas *ACM Digital Library* e *Science Direct*. Foram selecionados os artigos que tratavam sobre computação em nuvem com foco em IaaS, tendo como critério de seleção a leitura do resumo e da conclusão para identificar se o artigo atenderia ao objetivo do trabalho. Os artigos que não atendiam a esse critério eram descartados.

Desta maneira, este trabalho abrange as principais características dos corretores de nuvens, fornecendo uma classificação dos atores envolvidos na corretagem de nuvens computacionais, de acordo com o tipo de serviço que oferecem; o modelo conceitual; e também identificando algumas soluções disponíveis atualmente (*open source* ou proprietárias). Em suma, a contribuição deste trabalho consiste em uma taxonomia para classificação de corretores de nuvens baseada na migração de infraestruturas virtuais entre provedores IaaS.

O restante deste artigo está organizado da seguinte forma: a Seção 2 apresenta as definições e motivações para o desenvolvimento de uma taxonomia voltada a classificação de corretores de acordo com as tecnologias e métodos para migração de infraestruturas virtuais. Na sequência, a Seção 3 revisa as taxonomias existentes e os trabalhos relacionados. A taxonomia proposta é apresentada na Seção 4, enquanto sua aplicação e validação são abordadas na Seção 5, através da classificação de diversos corretores de nuvem já existentes. Por fim, a Seção 6 apresenta as considerações finais e trabalhos futuros.

2 Definições e Motivação

A taxonomia proposta neste trabalho visa auxiliar aos clientes no processo de seleção das soluções para gerenciamento de nuvens computacionais. Este auxílio é fornecido através da classificação e organização dos diversos conceitos relacionados ao gerenciamento e corretagem de nuvens de modo sistematizado, que permitem ao usuário prontamente associar uma série de características uma vez que a solução é classificada usando a taxonomia proposta. O modelo de computação em nuvem proporcionou ao usuário a percepção de recursos como se fossem infinitos, de maneira que estes estão sempre disponíveis e possam ser rapidamente aprovisionados e escalados. Estas características, entre outras, tornaram a computação e nuvem um modelo em ascensão, aumentando assim o número de consumidores adeptos desse mercado. Este crescimento tornou o mercado complexo, e assim como o crescimento de usuários ocorreu um expressivo crescimento de provedores de serviços, fazendo com o usuário final tenha que lidar com diversos esquemas de preços, modelos de oferecimento de MVs, interfaces de nuvens, etc. Neste contexto, o ator do corretor ganha importância e destaque.

O NIST [34] define que um corretor fornece somente serviços de intermediação, arbitragem e agregação. Diferenciando um pouco do NIST, o *Gartner* [59] preconiza que um corretor fornece três serviços principais: agregação, integração e personalização. Percebe-se que, em todas as definições, o corretor é o principal responsável pela intermediação de todos os relacionamentos existentes entre o consumidor e o provedor de nuvem. Constatase igualmente, a importância do corretor na interoperabilidade entre os provedores. Ainda, este ator tem como

objetivo sempre auxiliar na busca por soluções orientado às necessidades das aplicações dos usuários.

O corretor torna-se essencial para transformar o mercado heterogêneo de nuvens computacionais em uma *commodity* [67], similar aos serviços de água, eletricidade, gás e telefonia [10, 30]. Portanto, um corretor é uma entidade que possui como principal papel, no ambiente de nuvens computacionais, agregar, gerenciar e entregar os mais variados serviços fornecidos pelos mais diversos provedores de serviços. Em outras palavras: um corretor é um agente geralmente posicionado entre o consumidor e o provedor de nuvem e desempenha um papel de mediador entre estes atores através da compra de recursos de um provedor e sublocando estes recursos para o consumidor [9, 10]. Dessa maneira, um corretor simplifica a entrega de serviços complexos para os consumidores [51] e também proporciona a interoperabilidade e a portabilidade de aplicações entre diversos provedores de serviços [38]. Um corretor auxilia, assim, o consumidor de serviços de nuvens computacionais a evitar potenciais *vendor lock-in* [42].

A interoperabilidade entre provedores está diretamente relacionada com a possibilidade de migração dos serviços provisionados, ou seja, uma IV inicialmente provisionada e ativa no provedor P1 deve ser passível de migração para um provedor P2. A migração de uma IV compreende a migração conjunta de MVs e recursos virtuais de comunicação, incluindo as configurações relacionadas com balanceamento de carga, resolução de endereços, armazenamento remoto e segurança.

O usuário responsável pela IV migrante busca a realização do serviço com baixo tempo de indisponibilidade do serviço hospedado, ou nenhum tempo de indisponibilidade. Entretanto, as diversas características específicas do provedor P1 criam $\hat{a}ncoras$ que dificultam a migração dos recursos. A complexidade do cenário é aumentada pela falta de informação quanto aos procedimentos, mecanismos e modelos existentes para migração entre provedores. Devido a ampla documentação existente e não catalogada, um usuário contratante do provedor P1 enfrenta dificuldades em compreender quais mecanismos e soluções de migração estão disponíveis para mover sua IV ao provedor P2. Neste trabalho, é proposta uma taxonomia para classificação de corretores que auxiliem no processo de migração de IVs entre provedores IaaS facilitando a identificação dos recursos e principais aspectos relacionados na migração.

3 Trabalhos Relacionados

Mansour *et.al.* (2016) apresentam uma pesquisa sobre a interoperabilidade em nuvens computacionais, assim como, uma taxonomia sobre as abordagens de interoperabilidade em todos os níveis de serviço de uma nuvem (*Infrastructure-as-a-Service* (IaaS), *Platform-as-a-Service* (PaaS) e *Software-as-a-Service* (SaaS)) [47]. Essa taxonomia organiza os tipos de interoperabilidade em dois grupos principais: centrada no provedor e centrada no usuário. O tipo de interoperabilidade centrada no provedor é obtida através da adoção de padrões pela comunidade que suporta as nuvens federativas, nuvens híbridas e *cloud bursting* (reserva temporária de recursos em uma nuvem pública para atender picos de carga). Já a interoperabilidade centrada no usuário não depende dos padrões dos provedores, entretanto, recai sobre o próprio pessoal de Tecnologia da Informação (TI) interno ou em um terceiro ator, que neste caso desempenha o papel de corretor de nuvem. Neste contexto, o presente trabalho, classifica as abordagens centradas no usuário, conforme discutido na Seção 4.

O *Open Data Center Alliance* (ODCA) [54] disponibilizou no ano de 2014 um modelo de uso. Esse documento tem como principal objetivo fornecer uma definição para os corretores de nuvens e também descrever suas características divididas em quatro níveis distintos, sendo eles: nível de tecnologia, nível de serviços, nível de processos e seus aspectos comerciais. Neste aspecto, o ODCA forneceu um modelo de posicionamento do corretor na cadeia de serviços, utilizado no presente trabalho como base para o modelo de corretagem na Seção 4.8.

Por sua vez, Buyya, Yeo e Venugopal (2008) descrevem definições e arquiteturas para um mercado orientado a nuvens computacionais [9]. Os autores também apresentaram uma comparação de plataformas de nuvens a qual teve impacto relevante no desenvolvimento do presente trabalho. Baseado neste estudo, foi possível extrair informações referentes à algumas plataformas de nuvens e seus respectivos mecanismos de virtualização e suas interfaces de integração com os usuários. De forma complementar, Grozev e Buyya (2014) descrevem uma taxonomia para arquiteturas *Inter-Cloud* ou nuvens federativas [30]. No trabalho, os autores expõem que nuvens federativas possuem a característica de distribuição e coordenação da carga entre um conjunto de provedores de serviços. Neste contexto, os corretores de nuvens são facilitadores na coordenação e distribuição entre os provedores de serviços da nuvem federativa sendo que estes corretores podem ser diretamente gerenciados ou indiretamente

através de Service Level Agreement (SLA).

Fowley *et.al.* (2013) comparam as soluções *open source* de corretagem de acordo com as suas principais capacidades, arquitetura e interoperabilidade [24]. É disponibilizada uma proposta de taxonomia para os serviços de corretagem em nuvem baseados nos tipos de serviços (IaaS, PaaS e SaaS) e capacidades de corretagem de nuvem, como a descoberta, a integração, agregação, personalização, garantia de qualidade e otimização. Entretanto, os autores não abordaram a migração de IVs entre provedores. Recentemente, Gonzales (2016) apresentou uma revisão sistemática de artigos e taxonomias para orquestração de nuvem, contudo o estudo foca apenas em *workflows* de serviços em nuvens IaaS sem considerar aspectos de migração do serviço [27].

Conforme discutido, os trabalhos fornecem uma classificação ou taxonomia para os corretores de nuvens computacionais [44], [59], [47], [54], [9], [30], [24], [50]. Entretanto, observando a Tabela 1, é possível constatar que, os referidos trabalhos fornecem informações somente sobre alguns aspectos dos corretores. Na Tabela 1 foram reunidas as abordagens identificadas nas referências analisadas. Adicionalmente, foram inclusas na Tabela 1 abordagens que são consideradas relevantes na literatura de nuvens [34, 78] mas não foram mencionadas nestes trabalhos correlatos.

rabbia 1. 1 lobitaigons dos trabalitos folacionados.						
Abordagem/Referência	NIST	Gartner	ODCA	Buyya, Yeo e Venugopal	Grozev e Buyya	Fowley et.al.
Unidade de Migração	-	-	-	-	-	-
Mecanismo de Virtualização	-	-	-	Sim	-	-
Mecanismo de Migração	-	-	-	-	-	-
Mecanismo de Corretagem	-	-	-	-	Sim	-
Tipo de Serviço	Sim	Sim	-	-	-	Sim
Interface de Interação	-	-	-	Sim	-	-
Confidencialidade do Canal	-	-	-	-	-	-
Modelo de Corretagem	-	-	Sim	-	-	-

Tabela 1: Abordagens dos trabalhos relacionados.

Analisando a Tabela 1, percebe-se que abordagens relacionadas a *Unidade de Migração*, *Mecanismo de Migração de MVs* e *Confidencialidade do Canal* não são mencionadas em nenhum dos trabalhos correlatos aqui abordados. É importante ressaltar que referente a confidencialidade do canal pode-se fazer o uso de uma VPN, entretanto, existem métodos de cifragem e envio que não necessariamente dependem dos protocolos de redes utilizados. Neste sentido, o presente trabalho também apresenta como contribuição uma classificação mais abrangente, incluindo informações sobre como é realizada a interação entre os consumidores e provedores de serviços através do corretor, informações sobre os mecanismos de migração de Máquina Virtual (MV), modelo de corretagem e o mecanismo de corretagem utilizado pelos corretores. Além disso, percebe-se a necessidade de uma taxonomia que reuna todas as abordagens já descritas nos trabalhos correlacionados e que busque incluir as demais abordagens relevantes no processo de analisar soluções de migração IVs entre provedores IaaS.

4 Taxonomia Proposta

Visando auxiliar nos estudos futuros referentes aos corretores, bem como na tomada de decisão de migração entre provedores, este trabalho propõe uma taxonomia para classificar os diversos tipos de corretores. Com base nas abordagens listadas na Tabela 1 (Seção 3) foram definidas as categorias do primeiro nível da taxonomia aqui proposta (Figura 1) e seguidamente foi realizada uma pesquisa referenciada em uma quantidade significativa de materiais (artigos, documentações, *etc.*) buscando identificar as demais subcategorias com base na sua relevância no processo de migração de IVs. O material fornecido pela pesquisa referenciada embasou a análise e identificação de categorias e subcategorias da taxonomia proposta. A elaboração da taxonomia utilizou alguns dos conceitos propostos em [44], [59], [47], [24] e nos demais estudos referenciados neste trabalho. Como resultado desta análise, tem-se a organização da taxonomia ilustrada na Figura 1.

Como é possível observar na Figura 1, o modelo proposto agrupou os corretores em oito categorias principais. Cada categoria, por sua vez, é dividida em tantas subcategorias o quanto julgou-se necessário detalhar, de modo a fornecer diversos níveis de granularidade no uso da taxonomia proposta. Além disso, esta abordagem permite também que novas categorias e subcategorias possam ser adicionadas sem a necessidade de reformular a taxonomia como um todo.

Agregação Gerenciado Indiretamente Arbitragem Interação com Provedor Gerenciado Diretamente Orguestração Tipo de Servico Combinação Exata Mediação Integração Mecanismo de Corretagem Valor em um Conjunto Transformação Baseado em SLA Baseado em Limiares Pró-Ativo Baseada em Agentes Interface de Interação Live Migration Mecanismo de Migração de MVs Corretor laaS Interface WEB Pós-cópia Linguagens de Especificação Sem Confidencialidade Paravirtualização/Virtualização Completa Aninhados Mecanismo de Virtualização SSH Confidencialidade do Canal Padronizados Proprietário Máguinas Virtuais Unidade para Migração Modelo Linear

Modelo de Corretagem

Modelo Matriz

Figura 1: Taxonomia para classificação dos atores relacionados com a corretagem em nuvens IaaS, com foco na migração de recursos virtuais entre provedores.

4.1 Unidade para Migração

Infraestruturas Virtuais

Devido a ampla gama de corretores e mecanismos existentes para gerenciamento de recursos na nuvem, deve-se identificar quais componentes podem ser migrados. Na elaboração desta taxonomia são consideradas como unidades elementares de migração MV, processos ou IV:

- Processos. A migração de processos compreende o desacoplamento de um processo em execução sobre um sistema operacional e sua disponibilização integral em um novo hospedeiro. Nesse mecanismo, o estado interno do processo é transferido e reinicializado no sistema operacional de destino. Em alguns casos, o processo migrado mantém comunicação com o sistema operacional de origem, principalmente para realizar acesso a dados localmente armazenados [45].
- Máquina Virtual (MV). Quanto a migração de MV, o desacoplamento compreende o sistema operacional hospedado em conjunto com seus processos. A MV é migrada para o hospedeiro destinatário e a execução resumida [48].
- Infraestrutura Virtual (IV). A migração de infraestruturas virtuais acrescenta um nível de complexidade: a configuração de interconexão entre as MVs deve ser simultaneamente migrada [48] [4].

Ressalta-se que, a unidade de migração é relacionada com o mecanismo de migração implementado pelos provedores. A unidade de migração também pode ser um contêiner [49], que dependendo da abordagem de implementação usada, pode enquadrar-se em *Processos* ou *MV*.

4.2 Mecanismos de Virtualização

Os mecanismos de virtualização possuem um papel fundamental na computação em nuvem, principalmente quando se referem a entrega de soluções de IaaS na qual se tem como objetivo a virtualização do hardware. Estas técnicas e conceitos possibilitam a criação de um ambiente abstrato, isolado e separado do hospedeiro no qual é possível a execução de aplicações [62]. Neste contexto, é possível classificar em três os mecanismos de virtualização de hardware.

4.2.1 Paravirtualização/Virtualização Completa

No mecanismo de paravirtualização, o hipervisor não emula o hardware como ele realmente é, mas sim fornece uma interface de software ligeiramente diferente do hospedeiro, otimizada para a MV. Em consequência

disso, o sistema operacional convidado (executado na MV) necessita ser alterado para realizar chamadas também ao hipervisor e não apenas ao hardware [46, 62]. Devido a estas alterações, o sistema operacional convidado fica ciente de que está executando em um ambiente virtual e não em uma plataforma real. Já o mecanismo de virtualização completa possibilita a execução de sistemas operacionais em uma MV sem a necessidade de alteração, ou seja, esta técnica baseia-se em todo o hardware subjacente e oferece isolação e segurança para as MV [46, 62].

4.2.2 Virtualização Aninhada

A Virtualização Aninhada, também conhecida como recursiva, ocorre quando uma nova virtualização, seja ela de MV ou de redes, é gerada a partir de uma outra virtualização [39]. Neste tipo de virtualização, é gerada uma hierarquia com relacionamentos do tipo pai/filho. Inicialmente, a virtualização aninhada foi explorada por provedores de redes virtuais para permitir a composição de múltiplas camadas de virtualização [13]. Recentemente, este tipo de virtualização permitiu a criação de MVs em um ambiente complemente virtual.

4.2.3 Contêineres

Na tecnologia de virtualização por contêineres, o sistema operacional do hospedeiro separa os processos em espaços isolados [49] [76]. Desta maneira, as aplicações compartilham o mesmo sistema operacional porém não compartilham os mesmos recursos (bibliotecas do sistema e serviços, por exemplo). Esse modelo de virtualização é particularmente interessante em nuvens IaaS, permitindo que as aplicações implantadas utilizam menos espaço em memória principal, pois um hospedeiro pode alojar centenas de contêineres [76]. Atualmente, existem diversas abordagens de implementação de contêineres que diferem substancialmente na forma como fornecem isolamento e consome os recursos do hospedeiro.

4.3 Mecanismo de Migração

A migração de recursos virtuais é uma das técnicas utilizadas para implementação de elasticidade em nuvens IaaS [21]. No contexto de migração entre provedores, é necessário identificar quais tecnologias são suportadas por cada mecanismo, corretor ou *framework* de gerenciamento. Usualmente, provedores oferecem recursos para migração interna de recursos virtuais, permitindo o balanceamento de carga no substrato computacional.

Essa técnica pode ser explorada por usuários que desejam migrar suas aplicações hospedadas (Processos, MV ou IVs) para um novo provedor que melhor atenda os requisitos da aplicação [70]. Referente a migração dos recursos virtuais, principalmente quanto a memória, as soluções de virtualização oferecem três tipos de migração: *stop-and-copy*, *live migration* e pós-cópia [29, 48]. É importante ressaltar que a migração pode ser executada de diversas formas, dependendo da unidade de migração e serviço. Dessa forma, é possível classificar os corretores de nuvens computacionais de acordo com o tipo de migração oferecida.

4.3.1 Migração por Stop-and-Copy

A técnica de migração por *stop-and-copy* consiste em interromper a execução do recurso virtual e copiar todo estado da origem para o sistema de destino [29]. Tipicamente, o tempo de interrupção do serviço é prejudicial às aplicações hospedadas [23]. Entretanto, a implementação de uma migração *stop-and-copy* requer uma baixa complexidade de gerenciamento entre provedores.

4.3.2 Live Migration

A live migration é um mecanismo de migração de MVs que garante um tempo mínimo de indisponibilidade e interrupção dos serviços na visão do cliente. Esta técnica reduz o tempo de inatividade significantemente, entretanto este tipo de migração pode perdurar um tempo superior quando comparado com a técnica de *stop-and-copy*, impactando no desempenho das aplicações em execução [29]. O processo de *live migration* é basicamente composto por três fases: *push*, *stop-and-copy* e *pull*.

Inicialmente, na fase *push*, o conteúdo de memória é transferido do hospedeiro de origem para o hospedeiro de destino. Durante essa fase de cópia, o sistema de origem continua respondendo e processando todas as aplicações que estão em execução [33]. Na sequência, são retransmitidos todos os conteúdos da memória que even-

tualmente tenham sido alterado durante o processo de migração da memória [14]. Em um determinado momento é necessário interromper o serviço e reativar no hospedeiro de destino. Assim, a aplicação de *stop-and-copy* é necessária para guardar o estado atual da MV (memória e processador) e posteriormente, instanciar no hospedeiro de destino [29, 14]. Esse tempo de suspensão é proporcional ao tamanho da MV e a largura de banda de rede disponível para a transferência [14]. Por fim, uma ativação final (*pull*) inicia a execução no novo hospedeiro.

4.3.3 Migração por Pós-Cópia

Na técnica de migração por pós-cópia, primeiramente é realizada a transferência do estado atual do recurso virtual para o sistema hospedeiro de destino [33]. Após realizada essa transferência, inicia-se a MV no destino sem nenhum conteúdo na memória. A transferência da memória ocorre na medida em que a MV necessita de algum conteúdo que ainda não foi transferido, neste caso, a MV solicita o conteúdo faltante que é então transferido do hospedeiro original.

4.4 Mecanismos de Corretagem

Os mecanismos de corretagem representam a maneira como os corretores implementam os serviços de corretagem, que basicamente podem ser implementados através de interação com o provedor ou baseado em SLA [30]. Os mecanismos de corretagem por gerenciamento direto são os mais comumente utilizados quando não existe um mediador entre os usuários e os provedores de nuvens. Neste modelo, os corretores de gerenciamento diretos estão hospedados separadamente e precisam manter o controle das características de desempenho das aplicação. Já os corretores que realizam os serviços de corretagem baseados nas métricas estabelecidas em um SLA [20, 65], usualmente oferecem três tipos de seleção de provedores de acordo com as restrições ou condições estabelecidas no acordo. O primeiro tipo corresponde a seleção exata: o usuário fornece o SLA para o corretor, o qual, irá se responsabilizar por encontrar um ou mais provedores de serviços que atendem completamente ao solicitado pelo usuário, como por exemplo requisitos de largura de banda, segurança e qualidade de serviço. Por sua vez, a busca de valores em um conjunto é uma forma abrangente de pesquisa, permitindo que o corretor auxilie na decisão da configuração final.

No modelo baseado em limiares, o usuário estabelece valores no SLA que ao serem atingidos acionam o corretor para que ele execute alguma ação. Este método ainda pode ser subdividido em reativo e pró-ativo. No pró-ativo o corretor não espera o limiar definido pelo usuário ser alcançado, mas realiza alguma ação preventiva. No reativo, o corretor somente efetuaria alguma ação após o limiar definido pelo usuário ter sido atingido.

Em todos os casos, somente após a aceitação do usuário é estabelecido uma acordo com o provedor selecionado e os recursos desejados reservados [37]. Usualmente, a seleção é guiada por algoritmos de decisão utilizados para selecionar o provedor mais adequado para os requisitos definidos pelo cliente [75, 20].

4.5 Tipo de Serviço

Uma outra maneira a qual é possível classificar os atores de corretagem é quanto ao tipo de serviço que estes fornecem. Lui e seus colegas [44] definem que um corretor fornece serviços somente em três categorias: serviços de intermediação, serviços de agregação e por fim serviços de arbitragem. Entretanto, de acordo com Wadhwa et al. [70], tem-se que um corretor oferece serviços de agregação, serviços de integração, serviços de governança e serviços de personalização. Recentemente, Fowley et al. (2013) sugerem que um corretor, além do serviço de agregação, pode fornecer serviços de personalização e de intermediação [24].

Desta maneira é possível constatar que alguns dos serviços, como por exemplo, o serviço de agregação, são definidos por todos estes autores e os demais serviços se complementam. Assim, este trabalho propõe uma classificação que agrega os conceitos fornecidos por [44, 70, 24] e acrescenta outros como os serviços de orquestração, mediação e transformação.

Em suma, os serviços oferecidos por um corretor podem ser agrupados em quatro categorias principais: agregação, personalização, integração e governança. Por sua vez, os serviços de arbitragem são classificados como sendo serviços de personalização. Por fim, os serviços de orquestração, mediação e transformação são serviços que se encaixam na categoria de integração.

4.5.1 Serviços de Agregação

Um corretor de nuvem que presta serviços de agregação é aquele que realiza a agregação de dois ou mais serviços de diversos provedores para, possivelmente, vários usuários [24, 44]. O corretor fica responsável pelo relacionamento entre os usuários e o provedor de serviços, enquanto os usuários utilizam os serviços proporcionais às suas necessidades. Em serviços de agregação, os usuários pagam somente uma fatura ao corretor referente aos serviços entregues (que podem ser providos por múltiplos provedores) [70]. Opcionalmente um corretor de nuvem pode oferecer um gerenciamento das questões referentes a segurança e ao SLA para o cliente [24, 70], assim como, oferecer funcionalidades de descoberta de recursos [77].

4.5.2 Serviços de Personalização

Os corretores que atuam nesta categoria alteram ou adicionam capacidades, ou ainda, criam extensões para alguma funcionalidade especifica oferecida no serviço. Sendo assim, sua responsabilidade compreende a monitoração para verificar se as extensões estão em perfeito funcionamento de acordo com os requisitos informados no SLA [70, 24]. Nesta categoria tem-se os serviços de arbitragem como sendo uma subcategoria do serviços de personalização. O serviço de arbitragem é semelhante ao serviço de agregação, porém difere-se no fato de que os serviços oferecidos pelo corretor não são fixos, ou seja, o corretor tem a possibilidade de escolher os serviços de diversas agências ou provedores de serviços.

4.5.3 Serviços de Integração

Os corretores pertencentes à esta categoria combinam dois ou mais serviços com a realização de orquestração conjunta. O serviço resultante é visto como uma oferta combinada, geralmente implementada como uma integração vertical da pilha de nuvem, ou a integração de dados/processos de dentro de uma mesma camada [24]. Dentro desta categoria tem-se soluções como mediação, orquestração e transformação [77].

O termo orquestração é utilizado em computação para descrever a coordenação e o gerenciamento de sistemas computacionais complexos. O NIST também define orquestração de serviços como sendo o arranjo, coordenação e gestão da infraestrutura de nuvem para fornecer serviços para atender os requisitos de negócios e de TI [44]. Na mediação, os corretores fornecem serviços com um valor agregado ou melhoram um serviço através de um aperfeiçoamento nas funcionalidades [44]. Esta melhoria pode ser aplicada ao gerenciamento de acesso aos serviços em nuvem, gerenciamento de identidade, relatórios de desempenho, segurança, entre outras.

4.5.4 Serviços de Governança

Os corretores que atuam fornecendo serviços de governança lidam essencialmente com as tarefas administrativas básicas do sistema [70]. Em suma, seu principal objetivo é garantir que o provedor de serviço trate os dados do cliente de acordo com os regulamentos e políticas estabelecidas, bem como respeite as regras de seguranças para manipulação e armazenamento dos dados.

4.6 Interface de Integração

Esta categoria classifica os atores de corretagem de nuvem de acordo com o tipo de interface utilizada para realizar a interação entre usuários, corretores e provedores de nuvem. Usualmente, os corretores utilizam as próprias *Application Programming Interface* (API) dos provedores de serviços para invocar e operar os seus serviços, entretanto, também é comum encontrar corretores que desenvolvem as suas próprias APIs para realizar essa iteração com os provedores de nuvens. Estas APIs podem ser tanto proprietárias como de código aberto. Para uma melhor classificação, essa categoria foi subdividida ou outras duas subcategorias: interface baseada em agentes e interfaces autônomas [54].

4.6.1 Interface Baseada em Agentes

O modelo de interface baseada em agentes considera a existência de um mecanismo independente, sem dependência de provedores, capaz de efetuar as solicitações indicadas pelo usuário contratante [54]. Em suma, os agentes atuam como facilitadores, abstraindo as interfaces de provedores de serviço e as interfaces dos usuários

(quanto existentes). Devido a ampla gama de interfaces individualizadas por provedores, o desenvolvimento de agentes que atuem na interoperabilidade é uma tarefa desafiante.

4.6.2 Interface Autônomas

Esta categoria refere-se as APIs de código aberto, utilizadas para gerenciar serviços de nuvens através de uma API remota. Essa interface opera entre o provedor de serviços e o corretor [54]. Esta categoria pode ser subdividida em *Command Line Interface* (CLI), interfaces web e linguagens de especificação. As CLIs permitem a execução de tarefas rotineiras como instanciação, destruição, reconfiguração e migração de recursos virtuais. Uma representação gráfica das ações é oferecida aos usuários através de interfaces web. Por fim, linguagens de especificação podem ser utilizadas para descrever o comportamento esperado dos recursos virtuais durante seu ciclo de vida. Essas linguagens permitem uma representação abstrata, independente de provedores.

4.7 Confidencialidade do Canal

Ao realizar a migração entre dois provedores de serviço, é importante observar algumas características referentes ao canal de comunicação utilizado. A principal característica é a confidencialidade do canal que, neste cenário de migração, pode ser um canal aberto, ou seja, sem confidencialidade ou com algum tipo de confidencialidade. A implementação de um canal de comunicação seguro é importante para garantir a transmissão privada dos dados.

Um canal de comunicação com confidencialidade, no tocante a serviços de IaaS, pode ser obtido através de diversas soluções/abordagens (e.g., Secure Shell (SSH), Transport Layer Security (TLS) ou diversos tipos de Virtual Private Network (VPN)), sendo que algumas soluções podem ser baseadas em padrões ou podem ser soluções proprietárias [19]. Ressalta-se, que iniciativas globais como a Cloud Security Alliance (CSA) [19] recomendam o uso de padrões abertos de segurança para aumentar a interoperabilidade, porém também há abordagens para fins mais específicos que empregam soluções proprietárias [12].

4.8 Modelo de Corretagem

O modelo de corretagem determina o posicionamento do corretor no processo de descoberta e entrega de serviços de nuvens computacionais. Nesta categoria são listados três modelos: linear, matricial e compartilhado entre parceiros [54].

4.8.1 Modelo Linear

Este modelo permite que o usuário da nuvem gerencie o ciclo de vida dos serviços contratados através do corretor de nuvem, de maneira que este garante que o acesso do consumidor ao provedor é realizado exclusivamente através do corretor. Se conveniente, o corretor pode criar uma camada de abstração entre o provedor e o consumidor, escondendo assim o provedor subjacente do consumidor. Nestes casos o consumidor pode não saber a origem do serviço que está sendo prestado a ele é proveniente [54].

De acordo com o ODCA [54], este modelo possui os seguintes benefícios:

- (i) Interface única de consumo e gerenciamento dos serviços prestados;
- (ii) Possibilita que o consumidor selecione os serviços à partir de uma lista de provedores;
- (iii) Utiliza um único modelo de preço e faturamento para todos os provedores; e
- (iv) Fornece uma camada de governança para o consumo de serviços em nuvem.

4.8.2 Modelo Matricial

Neste modelo de corretagem, o próprio usuário fica responsável por realizar as funções do corretor, interagindo diretamente com o provedor através de camada de orquestração, que pode ser caracterizada como sendo uma API. A comunicação entre o corretor e os provedores pode ser realizada através dessa camada e de APIs proprietárias ou abertas, fornecidas pelos provedores de serviços [54].

4.8.3 Modelo Compartilhado entre Parceiros

No modelo compartilhado, parceiros de negócios disponibilizam recursos visando o benefício mútuo [54]. Ou seja, recursos ociosos em um determinado usuário ou provedor podem ser aproveitados por um terceiro.

5 Classificação de Mecanismos e Soluções

Com o objetivo de averiguar a aplicabilidade da taxonomia proposta, esta seção apresenta uma classificação de alguns corretores, soluções e mecanismos identificados durante a fase de pesquisa referenciada. Inicialmente, a Tabela 2 apresenta uma classificação de alguns mecanismos/corretores de acordo com a unidade de migração.

Tabela 2: Classificação dos mecanismos, soluções e corretores para nuvens IaaS seguindo a taxonomia de acordo com a unidade de migração.

Categoria		Corretor/Mecanismo	
	Processos	Xen-Blanket [73], LXC [32]	
Unidade para Migração	Maquinas Virtuais	Hyper-V [68], VirtualBox [72], KVM [40], Xen [6]	
	Infraestruturas Virtuais	HIPerNet [4], Chef [11], Puppet [60]	

É possível observar, na Tabela 2, que são exemplos de mecanismos que realizam a migração de Processos o Xen-Blanket e *Linux Containers* (LXC). Já como exemplo de mecanismos que consideram a MV como a unidade de migração tem-se o Hyper-V, VirtualBox, KVM e XEN. Por fim, o Hipernet, Chef e Puppet são capazes de realizar a migração de uma IV.

A Tabela 3 apresenta alguns corretores e soluções de acordo com os mecanismos de virtualização que eles implementam. O Xen Server opera com Paravirtualização, já VirtualBox e KVM fazem a Virtualização Completa e SuperCloud é capaz de fornecer Virtualização Aninhada. É importante observar que VMWare aparece em duas categorias sendo elas: Paravirtualização e Virtualização Completa. Tal fato é passível de ocorrer porque estas categorias não são ortogonais e o fato de um dos mecanismos ou corretores estar presente em uma categoria não o impossibilita de estar em outra.

Tabela 3: Classificação dos mecanismos, soluções e corretores para nuvens IaaS seguindo a taxonomia de acordo com os mecanismos de virtualização.

tom of motamismos de virtualização.					
Categoria		Corretor/Mecanismo			
	Paravirtualização	VMWare [71] [66], Xen Server [6]			
Mecanismos de Virtualização	Virtualização Completa	VirtualBox [72], KVM [40], VMWare [71] [66]			
	Virtualização Aninhada	SuperCloud [36]			
	Contêineres	OpenVZ [56], Docker [49], Kangoroo [63], Xen-Blanket [73]			

A Tabela 4 apresenta uma classificação de acordo com os mecanismos de migração de MVs implementados pelas soluções e/ou corretores. Novamente, tem-se o mecanismo VMWare em mais de uma categoria, implementando as técnicas de migração por Pré-cópia e *Live Migration*. Já Xen, nativamente, trabalha com *Live Migration* enquanto KVM com a técnica de pós-cópia.

Tabela 4: Classificação dos mecanismos, soluções e corretores para nuvens IaaS seguindo a taxonomia de acordo com os mecanismos de migração de MVs.

		8
Categoria	Corretor/Mecanismo	
	Pré-cópia	VMWare [71] [66]
Mecanismos de Migração de MVs	Live Migration	Xen [14], VMWare [71] [66], Hyper-V [68]
	Pós-cópia	KVM [40]

Na Tabela 5, observa-se que para as categorias Gerenciado Indiretamente e Pró-ativo não foram identificados mecanismos ou corretores. Nestes casos, constatou-se que pode haver essa característica nos corretores ou nas soluções, entretanto na pesquisa realizada, nenhuma referência foi identificada para classificação, contemplando estas características.

Tabela 5: Classificação dos mecanismos, soluções e corretores para nuvens IaaS seguindo a taxonomia de acordo com os mecanismos de corretagem

	Corretor/Mecanismo					
	Interação com o Provedor Gerenciado Indiretamente		-			
Mecanismos de Corretagem Baseado em SLA	interação com o i rovedor	Gerenciado Diretamente		JClouds [35]		
		Combinação Exata Chef [11], P		Chef [11], Puppet [60]		
	Rasaada am SLA	Valor em Conju	nto	Chef [11], Puppet [60], STRATOS [58]		
	Bascado em SLA	Baseado em Limeares	Reativo	CloudFormation [15]		
		Baseado em Emicares	Pró-ativo	-		

A Tabela 6 apresenta algumas soluções classificadas de acordo com o tipo de serviço oferecido. O CompatibleOne [24] é um corretor de código aberto independente de qualquer provedor de serviço de nuvem e o Boomi [22] é uma plataforma de privada voltada para SaaS.

	Categoria		Corretor/Mecanismo	
	Agreg		Boomi [22], CompatibleOne [74], BlueWolf [7]	
	Personalização	Arbitragem	CompatibleOne [74]	
Tipo de Serviço	Integração	no do Sarvigo		Cordys [18], RightScale [64]
Tipo de Sei viço		Integração	Mediação	CompatibleOne [74]
		Transformação	CloudSherpas [17], Appirio [5]	
	Governança		Gravitant [28], Vordel [69]	

Tabela 6: Classificação dos mecanismos, soluções e corretores para nuvens IaaS seguindo a taxonomia de acordo com os mecanismos de corretagem

A Tabela 7 apresenta exemplos dos mecanismos classificados de acordo com a sua interface de interação. De uma maneira geral, pode-se citar como exemplo de agentes autônomos a interface BOTO, o *framework* Chef e a ferramenta Puppet. O BOTO é uma interface desenvolvida na linguagem Python e fornece uma interface para a *Amazon Web Services* (AWS) [8]. Por outro lado, o Chef é um *framework* de integração desenvolvido com o objetivo de auxiliar o usuário na configuração e no gerenciamento de toda a infraestrutura [11]. Por fim, o Puppet é uma ferramenta que automatiza os processos administrativos, *e.g.*, instalações de pacotes e atualização das configurações dos servidores [60].

Tabela 7: Classificação dos mecanismos, soluções e corretores para nuvens IaaS seguindo a taxonomia de acordo com a interface de interação fornecida.

Categoria			Corretor/Mecanismo		
		Baseada em Agentes	BOTO [8], Chef [11], Puppet [60]		
Interface de Interação		Interface de Linha de Comando (CLI)	gsutil [31], CLI da AWS [2], Aeolus Project [1]		
Interrace de Interação Integração		Interface Web	AWS [3]		
		Linguagens de Especificação	VXDL [41], TOSCA [43], CloudML [26], OCCI [53]		

Verificando a Tabela 7, percebe-se exemplos de agentes autônomos baseados em CLI, como a ferramenta gsutil Tool do Google e a interface da linha de comando da Amazon AWS. A gsutil é uma ferramenta desenvolvida pela empresa Google e fornece meios dos usuários acessarem a Goolgle Cloud Storage através de CLI [31]. Já a interface CLI da AWS fornece, de uma maneira unificada, meios para o usuário gerenciar seus serviços da AWS [2]. Na categoria de interações através de uma Interface Web, os corretores disponibilizam seus serviços ou ferramentas para serem acessados através de portais hospedados na Internet. As ferramentas OpenStack [55], RackSpace [61] e AWS-EC2 [3] são exemplos desta categoria de agentes. Alguns corretores também utilizam Linguagens de Especificação para a realização desta iteração, e.g., Topology and Orchestration Specification for Cloud Applications (TOSCA) [43], Open Cloud Computing Interface (OCCI) [53] e Virtual Infrastructure Description Language (VXDL) [41]. Através destas linguagens, o usuário pode definir desde características mais básicas (como quantidade de CPU, memória e largura de banda) até a topologia de rede e os atributos temporais de cada recurso.

Referente ao quesito de confidencialidade do canal, a Tabela 8 apresenta algumas soluções. SuperCloud e Kangoroo não fornecem, por padrão, nenhum tipo de confidencialidade já o Xen Server utiliza SSH. As soluções OpenStack e serviços AWS tem a opção de configuração para empregar TLS na comunicação interna entre os

componentes da Nuvem.

Tabela 8: Classificação dos mecanismos, soluções e corretores para nuvens IaaS seguindo a taxonomia de acordo com a confidencialidade do canal implementada.

Categor	Corretor/Mecanismo		
- Carregor	Sem Confidencialidade		SuperCloud [36], Kangoroo
Confidencialidade do Canal	SSH		Xen Server [6]
	VPN	Padronizado	OpenStack [55], AWS [3]
	Proprietários Proprietários		-

Por fim, a Tabela 9 apresenta a classificação de acordo com o modelo de corretagem. O modelo de corretagem foi proposto inicialmente pela ODCA [54] e contém três categorias. Pode-se citar como um modelo de corretagem linear o modelo utilizado pela Amazon AWS e Google Compute Engine.

Tabela 9: Classificação dos mecanismos, soluções e corretores para nuvens IaaS seguindo a taxonomia de acordo com o modelo de corretagem.

Categor	ia	Corretor/Mecanismo	
	Modelo Linear	Amazon AWS [3], Google Computing Engine [25], Cloudify [16]	
Modelo de Corretagem	Modelo Matricial	-	
	Entre Parceiros	Eucalyptus [52]	

6 Considerações

Os corretores de nuvens possuem um papel essencial para usuários migrando para a nuvem ou até mesmo entre os próprios provedores de nuvem. Neste contexto, a padronização é o caminho mais sensato em direção a serviços de nuvens completamente interoperáveis [57]. Neste trabalho são descritos os principais conceitos de corretagem em nuvens computacionais, os seus principais aspectos e os objetivos de um corretor ao operar em um determinado ramo. Além disso, são apresentadas soluções tanto de código aberto quanto privadas e a contribuição mais relevante: uma taxonomia para a classificação destes corretores. A taxonomia proposta fornece meios para classificar os corretores de nuvens de acordo com o tipo de serviço prestado e a sua área de atuação no ambiente de nuvem computacional. Adicionalmente, a taxonomia também possui como objetivo auxiliar na padronização dos corretores e na correta identificação destes pelos usuários de nuvem, fazendo assim com que estes possuam mais informações ao seu dispor no momento de optar por um corretor ou ao migrar sua aplicação para a nuvem. Uma limitação da taxonomia apresentada é que a mesma não relaciona os aspectos de comunicação interna e externa no que tange a redes e endereçamento. Quanto a trabalhos futuros, a taxonomia foi um motivador para o desenvolvimento de um mecanismo para migração de IVs enter provedores de nuvem IaaS.

Agradecimentos

Os autores agradecem a UDESC pelo auxílio financeiro e ao Laboratório de Processamento Paralelo e Distribuído (LabP2D) pela disponibilização dos recursos computacionais para a realização do trabalho.

Referências

- [1] Aeolus Project. Disponível em: http://www.aeolus-project.org/. Acesso em: 30 abr. 2017.
- [2] Amazon Command Line Interface (CLI). Disponível em: https://aws.amazon.com/cli/. Acesso em: 30 abr. 2017.
- [3] Amazon Web Services (AWS). Disponível em: https://aws.amazon.com/pt/?nc2=h_lg. Acesso em: 30 abr. 2017.
- [4] Fabienne Anhalt, Guilherme Koslovski, and Pascale Vicat-Blanc Primet. Specifying and provisioning virtual infrastructures with HIPerNET. *International Journal of Network Management*, 20(3):129–148, 2010.

- [5] Appirio Next-Generation Worker and Customer Experience. Disponível em: https://appirio.com. Acesso em: 30 abr. 2017.
- [6] Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, and Andrew Warfield. Xen and the art of virtualization. In *ACM SIGOPS Operating Systems Review*, volume 37, pages 164–177. ACM, 2003.
- [7] IBM BlueWolf. Disponível em: http://www.bluewolf.com/. Acesso em: 30 abr. 2017.
- [8] BOTO: A Python Interface to Amazon Web Services. Disponível em: https://github.com/boto/boto. Acesso em: 30 abr. 2017.
- [9] Rajkumar Buyya, Chee Shin Yeo, and Srikumar Venugopal. Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services As Computing Utilities. In *Proceedings of the 2008 10th IEEE International Conference on High Performance Computing and Communications*, HPCC '08, pages 5–13, Washington, DC, USA, 2008. IEEE Computer Society.
- [10] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic. Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing As the 5th Utility. *Future Gener. Comput. Syst.*, 25(6):599–616, jun 2009.
- [11] Chef: Automate your Infrastructure. Disponível em: https://github.com/chef/chef. Acesso em: 30 abr. 2017.
- [12] Shruti Chhabra and V. S. Dixit. Cloud Computing: State of the Art and Security Issues. *SIGSOFT Softw. Eng. Notes*, 40(2):1–11, abr 2015.
- [13] N.M. Mosharaf Kabir Chowdhury and Raouf Boutaba. A survey of network virtualization. *Computer Networks*, 54(5):862 876, 2010.
- [14] Christopher Clark, Keir Fraser, Steven Hand, Jacob Gorm Hansen, Eric Jul, Christian Limpach, Ian Pratt, and Andrew Warfield. Live Migration of Virtual Machines. In *Proceedings of the 2Nd Conference on Symposium on Networked Systems Design & Implementation Volume 2*, NSDI'05, pages 273–286, Berkeley, CA, USA, 2005. USENIX Association.
- [15] AWS CloudFormation. Disponível em: https://aws.amazon.com/cloudformation/. Acesso em: 30 abr. 2017.
- [16] Cloudify The new hybrid cloud stack. One tool. Disponível em: http://getcloudify.org. Acesso em: 30 abr. 2017.
- [17] Accenture CloudSherpas. Disponível em: https://www.cloudsherpas.com. Acesso em: 30 abr. 2017.
- [18] OpenText Cordys. Disponível em: http://www.opentext.com.br/file_source/OpenText/en_US/PDF/OTCordys_Cloud.pdf. Acesso em: 30 abr. 2017.
- [19] CSA. Security Guidance for Critical Areas of Focus in Cloud Computing V3.0. Technical report, Cloud Security Alliance, 2011.
- [20] Cuomo, Antonio and Modica, Giuseppe and Distefano, Salvatore and Puliafito, Antonio and Rak, Massimiliano and Tomarchio, Orazio and Venticinque, Salvatore and Villano, Umberto. An sla-based broker for cloud infrastructures. *J. Grid Comput.*, 11(1):1–25, mar 2013.
- [21] Rodrigo da Rosa Righi. Elasticidade em cloud computing: conceito, estado da arte e novos desafios. *Revista Brasileira de Computação Aplicada (RBCA)*, 5(2):2–17, 2013.
- [22] Dell Boomi Build a Connected Business. Disponível em: http://www.boomi.com/. Acesso em: 30 abr. 2017.
- [23] M. R. Desai and H. B. Patel. Efficient Virtual Machine Migration in Cloud Computing. In 2015 Fifth International Conference on Communication Systems and Network Technologies (CSNT), pages 1015–1019, abr 2015.

- [24] Frank Fowley, Claus Pahl, and Li Zhang. A Comparison Framework and Review of Service Brokerage Solutions for Cloud Architectures. In Alessio Lomuscio, Surya Nepal, Fabio Patrizi, Boualem Benatallah, and Ivona Brandic, editors, Service-Oriented Computing ICSOC 2013 Workshops CCSA, CSB, PASCEB, SWESE, WESOA, and PhD Symposium, Berlin, Germany, December 2-5, 2013. Revised Selected Papers, volume 8377 of Lecture Notes in Computer Science, pages 137–149. Springer, 2013.
- [25] Google Compute Engine (GCE). Disponível em: https://cloud.google.com/compute. Acesso em: 30 abr. 2017.
- [26] Glauco Goncalves, Patricia Endo, Marcelo Santos, Djamel Sadok, Judith Kelner, Bob Melander, and Jan-Erik Mangs. CloudML: An Integrated Language for Resource, Service and Request Description for D-Clouds. In *Proceedings of the 2011 IEEE Third International Conference on Cloud Computing Technology and Science*, CLOUDCOM '11, pages 399–406, Washington, DC, USA, 2011. IEEE Computer Society.
- [27] Nelson Mimura Gonzales. *MPSF: Cloud Scheduling Framework for Distributed Workflow Execution*. PhD thesis, Escola Politécnica da Universidade de São Paulo, 2016.
- [28] IBM Gravitant Experience analytics-based decision making in a multi-sourced environemnt. Disponível em: http://tools.gravitant.com/cloud-compare-for-cloud-applications/. Acesso em: 30 abr. 2017.
- [29] Laura Grit, David Irwin, Aydan Yumerefendi, and Jeff Chase. Virtual Machine Hosting for Networked Clusters: Building the Foundations for "Autonomic" Orchestration. In *Proceedings of the 2Nd International Workshop on Virtualization Technology in Distributed Computing*, VTDC '06, pages 7–, Washington, DC, USA, 2006. IEEE Computer Society.
- [30] Nikolay Grozev and Rajkumar Buyya. Inter-Cloud architectures and application brokering: taxonomy and survey. *Softw., Pract. Exper.*, 44(3):369–390, 2014.
- [31] Google gsutil Tool. Disponível em: https://cloud.google.com/storage/docs/gsutil?hl=en. Acesso em: 30 abr. 2017.
- [32] Matt Helsley. LXC: Linux container tools. IBM devloperWorks Technical Library, 2009.
- [33] Michael R. Hines, Umesh Deshpande, and Kartik Gopalan. Post-copy Live Migration of Virtual Machines. *SIGOPS Oper. Syst. Rev.*, 43(3):14–26, jul 2009.
- [34] Michael Hogan and Annie Sokol. NIST Cloud Computing Standards Roadmap, 2013.
- [35] Apache JClouds, the Java Multi-Cloud Toolkit. Disponível em: https://jclouds.apache.org/. Acesso em: 30 abr. 2017.
- [36] Qin Jia, Zhiming Shen, Weijia Song, Robbert van Renesse, and Hakim Weatherspoon. Supercloud: Opportunities and Challenges. *SIGOPS Oper. Syst. Rev.*, 49(1):137–141, jan 2015.
- [37] Foued Jrad, Jie Tao, and Achim Streit. SLA based Service Brokering in Intercloud Environments. In Frank Leymann, Ivan Ivanov, Marten van Sinderen, and Tony Shan, editors, *CLOSER*, pages 76–81. SciTePress, 2012.
- [38] Dongjae Kang, Sokho Son, and Jinmee Kim. Design of Cloud Service Brokerage System Intermediating Integrated Services in Multiple Cloud Environment. *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, 8(12):2035 2040, 2014.
- [39] Bernhard Kauer, Paulo Verissimo, and Alysson Bessani. Recursive virtual machines for advanced security mechanisms. In 2011 IEEE/IFIP 41st International Conference on Dependable Systems and Networks Workshops (DSN-W), pages 117–122. IEEE, 2011.
- [40] Avi Kivity, Yaniv Kamay, Dor Laor, Uri Lublin, and Anthony Liguori. KVM: the Linux virtual machine monitor. In *Proceedings of the Linux symposium*, volume 1, pages 225–230, 2007.
- [41] Guilherme Piegas Koslovski, Pascale Vicat-Blanc Primet, and Andrea Schwertner Charao. *VXDL: Virtual Resources and Interconnection Networks Description Language*, pages 138–154. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

- [42] Wubin Li, Johan Tordsson, and Erik Elmroth. Modeling for Dynamic Cloud Scheduling Via Migration of Virtual Machines. In *Proceedings of the 2011 IEEE Third International Conference on Cloud Computing Technology and Science*, CLOUDCOM11, pages 163–171, Washington, DC, USA, 2011. IEEE Computer Society.
- [43] Paul Lipton and Simon Moser. Topology and Orchestration Specification for Cloud Applications Version 1.0, Novembro 2013.
- [44] Fang Liu, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee Badger, and Dawn Leaf. NIST Cloud Computing Reference Architecture: Recommendations of the National Institute of Standards and Technology (Special Publication 500-292). CreateSpace Independent Publishing Platform, USA, 2012.
- [45] Renaud Lottiaux, Pascal Gallard, Geoffroy Vallée, Christine Morin, and B. Boissinot. OpenMosix, OpenSSI and Kerrighed: a comparative study. In *CCGRID*, pages 1016–1023. IEEE Computer Society, 2005.
- [46] Meriam Mahjoub, Afef Mdhaffar, Riadh Ben Halima, and Mohamed Jmaiel. A Comparative Study of the Current Cloud Computing Technologies and Offers. In *Proceedings of the 2011 First International Symposium* on Network Cloud Computing and Applications, NCCA11, pages 131–134, Washington, DC, USA, 2011. IEEE Computer Society.
- [47] Ibrahim Mansour, Reza Sahandi, Kendra Cooper, and Adrian Warman. Interoperability in the Heterogeneous Cloud Environment: A Survey of Recent User-centric Approaches. In *Proceedings of the International Conference on Internet of Things and Cloud Computing*, ICC16, pages 62:1–62:7, New York, NY, USA, 2016. ACM.
- [48] Violeta Medina and Juan Manuel García. A Survey of Migration Mechanisms of Virtual Machines. *ACM Comput. Surv.*, 46(3):30:1–30:33, jan 2014.
- [49] Dirk Merkel. Docker: Lightweight linux containers for consistent development and deployment. *Linux J.*, 2014(239), mar 2014.
- [50] Charles Miers, Marcos Simplicio Jr, Tereza Carvalho, Guilherme Koslovski, Marco Rojas, Bruno Rodrigues, and Leonardo Horn Iwaya. Análise de Segurança para Soluções de Computação em Nuvem. In SBRC 2014 Minicursos. SBC, mai 2014.
- [51] Srijith K. Nair, Sakshi Porwal, Theo Dimitrakos, Ana Juan Ferrer, Johan Tordsson, Tabassum Sharif, Craig Sheridan, Muttukrishnan Rajarajan, and Afnan Ullah Khan. Towards Secure Cloud Bursting, Brokerage and Aggregation. In *Proceedings of the 2010 Eighth IEEE European Conference on Web Services*, ECOWS '10, pages 189–196, Washington, DC, USA, 2010. IEEE Computer Society.
- [52] Daniel Nurmi, Rich Wolski, Chris Grzegorczyk, Graziano Obertelli, Sunil Soman, Lamia Youseff, and Dmitrii Zagorodnov. The Eucalyptus open-source cloud-computing system. In *Cluster Computing and the Grid*, 2009. CCGRID'09. 9th IEEE/ACM International Symposium on, pages 124–131. IEEE, 2009.
- [53] OCCI Open Cloud Computing Interface. Disponível em: http://occi-wg.org/. Acesso em: 30 abr. 2017.
- [54] Open Data Center Alliance. Open Data Center Alliance Usage Model: Cloud Service Brokering Rev. 1.0. Technical report, Open Data Center Alliance, 2014.
- [55] OpenStack Open source software for creating private and public clouds. Disponível em: https://www.openstack.org. Acesso em: 30 abr. 2017.
- [56] OpenVZ Open Virtuozzo. Disponível em: https://openvz.org/Main_Page. Acesso em: 30 abr. 2017.
- [57] Claus Pahl, Li Zhang, and Frank Fowley. A look at cloud architecture interoperability through standards. In *In: The Fourth International Conference on Cloud Computing, Grids, and Virtualization, Mai 27- Junho 1, 2013, Valenica, Espanha. ISBN 978-1-61208-271-4*, Valenica, Spain, mai 2013. IARIA.
- [58] Przemyslaw Pawluk, Bradley Simmons, Michael Smit, Marin Litoiu, and Serge Mankovski. Introducing STRATOS: A Cloud Broker Service. In Rong Chang, editor, *IEEE CLOUD*, pages 891–898. IEEE, 2012.

- [59] Christy Pettey and Rob van der Meulen. Gartner Says Cloud Consumers Need Brokerages to Unlock the Potential of Cloud Services. Technical report, Gartner, 2009.
- [60] Puppet: the shortest path to better software IT automation for cloud, security and DevOps. Disponível em: https://github.com/puppetlabs/puppet. Acesso em: 30 abr. 2017.
- [61] RackSpace: Managed Dedicated and Cloud Computing Services. Disponível em: https://www.rackspace.com/pt-br. Acesso em: 30 abr. 2017.
- [62] S.Thamarai Selvi Rajkumar Buyya, Christian Vecchiola. *Mastering Cloud Computing Foundations and Applications Programming*. Morgan Kaufmann, 2013.
- [63] Kaveh Razavi, Ana Ion, Genc Tato, Kyuho Jeong, Renato Figueiredo, Guillaume Pierre, and Thilo Kielmann. Kangaroo: A tenant-centric software-defined cloud infrastructure. In *Cloud Engineering (IC2E)*, 2015 IEEE International Conference on, pages 106–115. IEEE, 2015.
- [64] Universal Cloud Management Platform at RightScale. Disponível em: http://www.rightscale.com. Acesso em: 30 abr. 2017.
- [65] Owen Rogers and Dave Cliff. A financial brokerage model for cloud computing. *Journal of Cloud Computing*, 1(1):1–12, 2012.
- [66] Mendel Rosenblum. VMware's virtual platformTM. In *Proceedings of hot chips*, volume 1999, pages 185–196, 1999.
- [67] Johan Tordsson, Rubén S. Montero, Rafael Moreno-Vozmediano, and Ignacio M. Llorente. Cloud Brokering Mechanisms for Optimized Placement of Virtual Machines Across Multiple Providers. *Future Gener. Comput. Syst.*, 28(2):358–367, fev 2012.
- [68] Anthony Velte and Toby Velte. Microsoft virtualization with Hyper-V. McGraw-Hill, Inc., 2009.
- [69] Axway Vordel. Disponível em: https://www.axway.com/en/enterprise-solutions/vordel-products. Acesso em: 30 abr. 2017.
- [70] Bimlesh Wadhwa, Aditi Jaitly, and Bharti Suri. Cloud Service Brokers: An Emerging Trend in Cloud Adoption and Migration. 2013 20th Asia-Pacific Software Engineering Conference (APSEC), 0:140–145, 2013.
- [71] Carl A Waldspurger. Memory resource management in VMware ESX server. *ACM SIGOPS Operating Systems Review*, 36(SI):181–194, 2002.
- [72] Jon Watson. Virtualbox: bits and bytes masquerading as machines. Linux Journal, 2008(166):1, 2008.
- [73] Dan Williams, Hani Jamjoom, and Hakim Weatherspoon. The Xen-Blanket: virtualize once, run everywhere. In *Proceedings of the 7th ACM european conference on Computer Systems*, pages 113–126. ACM, 2012.
- [74] Sami Yangui, Iain-James Marshall, Jean-Pierre Laisne, and Samir Tata. CompatibleOne: The open source cloud broker. *Journal of Grid Computing*, 12(1):93–109, 2014.
- [75] Tao Yu and Kwei-Jay Lin. The Design of QoS Broker Algorithms for QoS-Capable Web Services. *Int. J. Web Service Res.*, 1(4):33–50, 2004.
- [76] Jie Zhang, Xiaoyi Lu, and Dhabaleswar K Panda. Performance Characterization of Hypervisor-and Container-Based Virtualization for HPC on SR-IOV Enabled InfiniBand Clusters. In 2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), pages 1777–1784. IEEE, 2016.
- [77] Li Zhang, Frank Fowley, and Claus Pahl. A Template Description Framework for Services as a Utility for Cloud Brokerage. In *Proceedings of the 4th International Conference on Cloud Computing and Services Science*, pages 60–69, 2014.
- [78] Jun-Feng Zhao and Jian-Tao Zhou. Strategies and Methods for Cloud Migration. *International Journal of Automation and Computing*, 11(2):143–152, mar 2015.