UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO

Jorge Werner

UMA ABORDAGEM PARA ALOCAÇÃO DE MÁQUINAS VIRTUAIS EM AMBIENTES DE COMPUTAÇÃO EM NUVEM VERDE

Florianópolis

2011

Jorge Werner

UMA ABORDAGEM PARA ALOCAÇÃO DE MÁQUINAS VIRTUAIS EM AMBIENTES DE COMPUTAÇÃO EM NUVEM VERDE

Dissertação submetida ao Curso de Pós-Graduação em Ciências da Computação para a obtenção do Grau de Mestre em Ciências da Computação.
Orientador: Prof. Dr. Carlos Becker Westphall

Florianópolis

2011

Catalogação na fonte elaborada pela Biblioteca Universitária
da
Universidade Federal de Santa Catarina

W492a Werner, Jorge
Uma abordagem para alocação de máquinas virtuais em ambientes de computação em nuvem verde [dissertação] / Jorge Werner ; orientador, Carlos Becker Westphall. - Florianópolis, SC, 2011.
137 p.: il., grafs., tabs.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Ciência da Computação.

Inclui referências
1. Ciência da computação. 2. Sistema de computação virtual. I. Westphall, Carlos Becker. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Ciência da Computação. III. Título.
CDU 681

Jorge Werner

UMA ABORDAGEM PARA ALOCAÇÃO DE MÁQUINAS VIRTUAIS EM AMBIENTES DE COMPUTAÇÃO EM NUVEM VERDE

Esta Dissertação foi julgada aprovada para a obtenção do Título de "Mestre em Ciências da Computação", e aprovada em sua forma final pelo Curso de Pós-Graduação em Ciências da Computação.

Florianópolis, 25 de fevereiro 2011.

Prof. Coordenador, Dr. Mário Antônio Ribeiro Dantas
Departamento de Informática e Estatística - UFSC

Prof. Dr. Carlos Becker Westphall
Orientador

Banca Examinadora:

Prof. Dr. João Bosco Mangueira Sobral
Departamento de Informática e Estatística - UFSC

Prof. Dra. Carla Merkle Westphall
Departamento de Informática e Estatística - UFSC

Prof. Dr. Rômulo Silva de Oliveira
Departamento de Automação e Sistemas - UFSC

Dedico esta dissertação à toda minha família, em especial minha esposa Keity, minha filha Anna Luiza e aos meus pais Pedro e Marlene Werner.

AGRADECIMENTOS

Agradeço principalmente ao professor Dr. Carlos Becker Westphall pela confiança depositada, que mesmo sem conheçer me chamou para trabalhar nesta grande equipe do LRG - Laboratório de Redes e Gerênca, e ainda destinando a bolsa de Mestrado.

Não poderia esquecer do Dr. Fernando Luis Koch pelas grandes contribuições e sugestões a proposta de pesquisa, ao Msc. Carlos Oberdan Rolim pelos comentários.

Gostaria também de agradecer ao mestrando Guilherme Arthur Geronimo pelos dias trabalhados e ao Rafael Freitas pela contribuição nas simulações.

Nunca desista de seus sonhos

autor desconhecido

RESUMO

Este trabalho propõe uma solução para o controle integrado de computação e os elementos do ambiente em nuvens verdes. A abordagem funciona com base em modelos de organização que regulam o comportamento dos componentes autônomos (agentes), que veêm os elementos ambientais como prestadores de serviços, por exemplo, servidores de processamento, carga de serviço de distribuição, processador de tarefa, serviço de redução de temperatura, entre outros. Argumenta-se que esta arquitetura pode suportar até 70% em relação a otimização energética dos centros de dados que utilizam um modelo de *uma infraestrutura por aplicação* e até 40% de otimização em relação a modelos de computação baseados em nuvem. O desafio é oferecer uma solução integrada de gestão do sistema que controla e regula as configurações internas, em resposta à flutuação dinâmica das variáveis externas proporcionando um sistema de informação escalável, flexível e de qualidade. Entende-se assim que a idéia para resolver as questões deva passar por uma estratégia para alocação dinâmica de máquinas virtuais em ambientes de computação em nuvem, a qual se baseia na migração da carga de trabalho de um servidor físico para outro e na realocação de recursos conforme a demanda por capacidade de processamento possa variar, avaliando o cenário de alocação das máquinas virtuais ao término (isto é, *on line*) da execução de cada tarefa, com o intuito de identificar um hospedeiro ocioso ou subutilizado. O trabalho introduz um modelo de gestão do sistema, e um modelo de alocação e distribuição de máquinas virtuais, analisando o comportamento do sistema, são descritos os princípios de funcionamento, e por fim é apresentado um cenário de caso de uso.

Palavras-chave: Computação em Nuvem, Computação Verde, Máquinas Virtuais.

ABSTRACT

This proposes an integrated solution for controlling and computing aspects of the environment in green clouds. The approach works based on models of organization that govern the behavior of autonomous components (agents) who see the environmental elements such as service providers, for example, processing servers, load distribution service processor, task, service reduction temperature, among others. It is argued that this architecture can support up to 70% on energy optimization of data centers that use a model of *an infrastructure for application* and 40% compared to optimization models based computing cloud. The challenge is to offer integrated solution for managing the system that controls and regulates the internal settings in response to fluctuating dynamics of external variables provide a robust, scalable information systems, flexible and quality. We understand well that the idea for resolving the issues should go through a strategy for dynamic allocation of virtual machines in cloud computing environments, which is based on the migration of workload from one physical server to another and the reallocation of resources as demand for processing capacity varies by assessing the stage of allocation of virtual machines at the end (*i.e.* online) the implementation of each task, with the aim of identifying a host idle or underused. We introduce a model management system and a model of allocation and distribution of virtual machines, we analyze the behavior of the system, describe the operating principles, and present a use case scenario.

Keywords: Cloud Computing, Green Computing, Virtual Machines.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

SUMÁRIO

1 INTRODUÇÃO

Os centros de dados modernos, compostos por milhares de servidores, processam e tratam diferentes aplicações, assim existe uma busca constante de novas soluções para a redução de custos e um melhor aproveitamento de sua infraestrutura. Nesta busca surgiram os conceitos de computação em nuvem e computação verde, para simplificar o gerenciamento e fazer melhor uso dos recursos dos milhares de servidores. Esta dissertação discute os desafios enfrentados por esses *data centers*, e apresenta como a computação em nuvem aliada a computação verde pode oferecer soluções inovadoras e eficientes.

1.1 CONTEXTO

Computação em Nuvem ou simplesmente do inglês *Cloud Computing*, é um conceito bastante novo, é basicamente um novo modelo de computação emergente, nem mesmo existe um consenso sobre uma definição exata sobre o tema. O Modelo destaca-se, no entanto como sendo um novo paradigma que surgiu com a evolução tecnológica, da necessidade de redução de custos e da necessidade de escalabilidade para a infraestrutura. Esse conceito envolve aplicações, infraestrutura e plataforma computacional, providos sob demanda, de forma que os recursos sejam dinamicamente adequados às necessidades locais, atribuindo assim todo o poder computacional para a alocação dinâmica de recursos conforme as necessidades das aplicações.

Embora o modelo esteja fundamentado em conceitos conhecidos como os de virtualização, computação distribuída, computação em grade, computação utilitária e até mesmo o conceito de Internet, existem diversos desafios a serem solucionados, como a segurança, a interoperabilidade e o gerenciamento de forma eficaz e acessível, garantindo que os serviços providos atendam as expectativas.

A necessidade de redução de custos, mudanças tecnológicas muito rápidas, demandas de alto processamento e armazenamento, requerem investimentos altos de TI - Tecnologia da Informação. Ainda na busca de novas tecnologias a computação em nuvem se tornou uma grande oportunidade para a obtenção de recursos computacionais dinâmicos, de maneira rápida e fácil, sem se preocupar com novos investimentos em *hardware* e *software*.

(BUYYA; YEO; VENUGOPAL, 2008), define Computação em Nuvem como um tipo de sistema paralelo e distribuído que consiste de uma coleção de computadores interligados e virtualizados que são dinamicamente

provisionados e apresentados como um ou mais recursos de computação unificados baseados em acordos de nível de serviço estabelecidos através de negociação entre os prestadores de serviços e os consumidores.

A enorme quantidade de informação gerada pela sociedade e a procura de soluções para oferecer serviços de informação relacionados que tratem esse volume de tarefas com um menor custo, impulsiona pesquisas na área da computação. Uma alternativa a essa necessidade é o uso de computação em nuvem. Ambientes de computação em nuvem visam proporcionar a integração de recursos de armazenamento escalável, com grande poder de processamento e pelo uso de máquinas virtuais.

Na busca por soluções que possibilitem o uso eficiente dos recursos computacionais, que traga o equilíbrio nos pilares econômico, social e ambiental, para o desenvolvimento sustentável da tecnologia da informação, surgiu o conceito de computação verde (SHULZ, 2009).

A computação autonômica (BALEN et al., 2009) agrega em todo esse contexto propondo melhorias na gerência de sistemas computacionais para melhorar o aproveitamento de recursos, tanto de operadores humanos, quanto de equipamentos, introduzindo conceitos de auto-gerenciamento, para configuração, proteção, otimização e regeneração.

Neste trabalho são investigados os principais conceitos apresentados pela comunidade científica, bem como algumas aplicações de suas características num ambiente de computação em nuvem, para a busca de um modelo capaz de se "auto-gerenciar"garantindo um melhor aproveitamento dos recursos ociosos, considerando a dinamicidade da carga, para a busca um modelo para uso eficiente de recursos tecnológicos disponíveis em organizações, racionalizando tais recursos de forma a se obter um aproveitamento melhor deles, diminuindo desperdícios.

1.2 MOTIVAÇÃO

O estudo de um modelo para alocação e distribuição de máquinas virtuais em ambiente de Computação em Nuvem Verde, deve desenvolver uma relação custo-benefício para os ambientes de TI, onde a computação em nuvem tem se tornado uma realidade.

A tecnologia de computação em nuvem vem crescendo rapidamente, por sua inerente condição em agregar recursos e fazer com que estes sejam transparentes ao usuário. Os recursos seriam disponibilizados como um serviço de telefonia, ou de energia, onde são conectados os aparelhos, e tais recursos são utilizados, consumidos e ao final do mês o usuário paga a conta sem se preocupar com o que esta por trás (isto é, quais equipamentos, re-

des são necessários) para a disponibilidade do serviço. O uso da tecnologia de computação em nuvem traz alguns benefícios, pois quando utilizada em um ambiente pode permitir uma melhor utilização de recursos computacionais, para um melhor aproveitamento de capacidades computacionais. No entanto, as enormes quantidades de recursos de computacionais necessários para processar diferentes aplicações, de diferentes empresas, utilizados na computação em nuvem, podem trazer um aumento no consumo de energia na implantação de novos centros de dados (GREENPEACE, 2010).

Segundo (SHULZ, 2009), em 2006 os centros de dados de TI consumiam cerca de 61 bilhões de quilowatts-hora (kWh), ou 61 bilhões vezes 1000 watts-hora de eletricidade, com um custo aproximado de cerca de U$ 4,5 bilhões de dólares. Também foi relatado que os centros de dados de TI, em média, consomem 15 a 25 vezes (ou mais) de energia por metro quadrado em comparação com um edifício de escritórios típico. Sem mudanças no consumo de energia elétrica e melhoria da eficiência, o consumo estimado dos centros de dados será superior a 100 bilhões de kWh até 2011.

Assim as necessidades de redução no consumo de energia, em conjunto com a computação em nuvem, se tornam desafiadores e interessantes devido aos diferentes problemas de gestão dos recursos, de mecanismos e políticas para a alocação dinâmica de carga de trabalhos entre os diferentes centros de dados. Os sistemas baseados na computação em nuvem necessitam determinar a localização, e a distribuição ideal para processamento de serviços, atingindo níveis razoáveis de qualidade de serviço - QoS (do inglês, *Quality of Service*) e minimizando os gastos energéticos.

Para tratar de tais questões a Computação em Nuvem Verde tem sido defendida como uma alternativa. Computação em Nuvem Verde é similar a Computação em Nuvem convencional com carga para as aplicações distribuídas através de conjunto de máquinas virtuais em execução em servidores físicos agrupados em *clusters*. No entanto, a Computação em Nuvem Verde pode desligar ou colocar os recursos ociosos de um modo latente (isto é, modo de baixo consumo de energia) para economizar energia. Com esta abordagem, a Computação em Nuvem Verde propõe uma redução nos custos de energia dos centros de dados.

No entanto, há falta de um modelo de distribuição dos recursos que considera a relação de carga do sistema e consumo de energia. O argumento é que esse modelo precisa ser preditivo e deva considerar modelos alternativos de distribuição de recursos, avaliação e recomendação dos efeitos de configuração em relação a variações no ambiente operacional. Os modelos que são aplicados na computação em nuvem convencional (isto é, não verde) ignoram essa relação.

Desta forma, a integração dos elementos e tecnologias acima descri-

tos forma a base para o desenvolvimento desta pesquisa, já que, a proposta deste trabalho é utilizar técnicas de sistemas autônomos para gerenciamento de recursos, alocando de forma dinâmica às necessidades do cliente. Neste trabalho, a prova do conceito, será aplicada a arquitetura em um ambiente de simulação, simulando situações imprevisíveis em aplicações de Internet, hospedadas em centros de dados que utilizam o modelo de computação em nuvem em seu parque tecnológico, na busca de um modelo de eficiência energética nestes ambientes.

Consideramos um cenário simples demonstrado pela Figura 1.

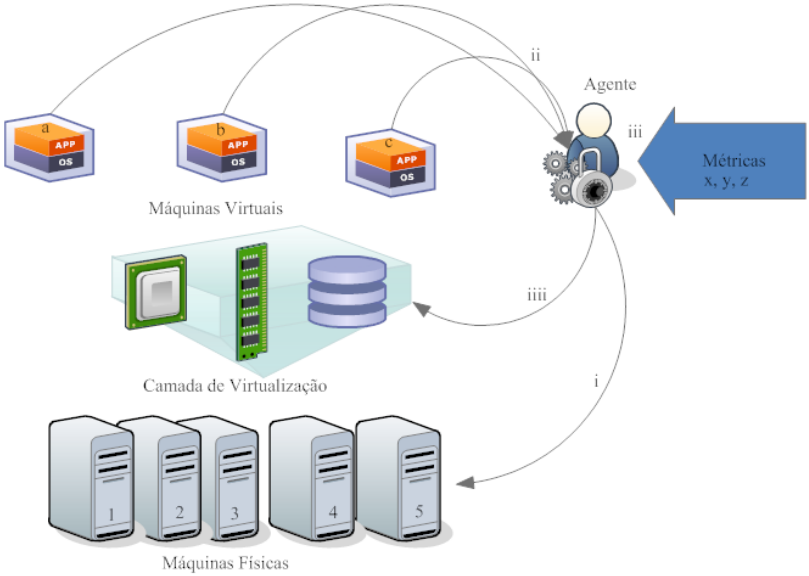

Figura 1 – Ambiente Dinâmico.

No exemplo da Figura 1, num ambiente de um centro de dados, que utiliza diversas máquinas físicas concentradas para armazenamento e processamento de diversas aplicações, de diversas áreas de uma empresa. Basicamente considera-se o centro de processamento de dados (isto é, o *data center*) de uma empresa, com máquinas físicas heterogêneas. No ambiente está sendo utilizado o modelo de Computação em Nuvem, ou seja, alocando e distribuindo seus recursos, de maneira elástica, através de máquinas virtuais, porém a empresa não consegue controlar eficientemente a utilização das máquinas físicas sendo possível o encontro de máquinas físicas ociosas, ao

mesmo tempo em que em momentos imprevisíveis, outras máquinas estão sobrecarregadas. Desta forma a proposta apresentada sugere um modelo de gerenciamento do modelo de Computação em Nuvem Verde, implantando o controle de recursos de máquinas virtuais dinâmico. Teríamos um agente que coleta informações (quantidade total e utilizada de máquinas, memória, armazenamento e processamento) das máquinas físicas em tempo real (i), ele também recebe informações (quais máquinas ativas, tarefas) das máquinas virtuais (ii), por outro lado recebe as métricas de negócio (iiii) estabelecidas no contrato entre o prestador e a empresa. Em seguida faz uma avaliação das máquinas que podem ser concentradas e quais tarefas estão subutilizadas para melhor distribuir ou realocar as máquinas virtuais, possibilitando uma melhor relação de custo e eficiência, aproveitando o menor número de máquinas físicas, que atenda os requisitos das aplicações, respeitando as regras requeridas (por exemplo, tempo de inatividade, garantia de QoS).

Assim na Figura 1, demonstramos que um agente deve conhecer as informações em tempo real das máquinas físicas (i), das aplicações (ii), das máquinas virtuais (iii) e dos conjuntos de métricas de negócio (iiii), de forma a promover uma auto-configuração das máquinas virtuais mantendo um uso eficiente dos recursos tecnológicos do centro de dados, utilizando um modelo de Computação em Nuvem Verde.

A abordagem é preencher esta lacuna e fornecer um modelo teórico para a alocação e distribuição de recursos para Computação em Nuvem Verde. O modelo descrito neste trabalho considera a dinamicidade da carga e alocação de recursos intrínsecos às nuvens com aspectos ecológicos relacionados à eficiência energética da Nuvem Verde. O modelo poderá ser usado por administradores de sistema para avaliar as configurações alternativas de distribuição dos recursos e antecipar os benefícios e problemas associados com diferentes configurações, a fim de maximizar os custos operacionais - OPEX (do inglês, *Operational Expenditure*), as despesas de capital - CAPEX (do inglês, *Capital Expenditure*) e promover a Computação em Nuvem Verde no parque tecnológico de médias e grandes corporações.

1.3 SOLUÇÃO PROPOSTA

Dentro de ambientes de computação em nuvem as cargas de trabalho, executadas por diferentes aplicativos podem ser dinâmicas, imprevisíveis, ao ponto que o posicionamento inicial e as quotas de determinado recurso para uma máquina virtual podem se tornar insuficiente ou ao mesmo tempo excedentes para as oscilações de demanda.

Considerando um modelo de Computação em Nuvem Verde, onde há

uma grande variação de demanda e uma grande velocidade na mudança das cargas de trabalho, é necessário utilizar técnicas de gestão automatizada de recursos para promover o uso eficiente de recursos. Acredita-se que uma estratégia eficaz de análise, alocação e distribuição automática dos recursos em diferentes centros de dados, pode trazer economia de recursos contribuindo para o uso eficiente de recursos computacionais, provendo a elasticidade necessária para a economia de recursos tecnológicos em equipamentos subutilizados ou ociosos.

A proposta visa fornecer através da metodologia da teoria da organização, um modelo teórico para a alocação e distribuição de recursos em Computação em Nuvem Verde, que atenda as métricas de negócio, bem como tratar as variações bruscas na demanda de serviços, aproveitando desta forma todos os recursos tecnológicos possivelmente ociosos ou subutilizados, nos centros de dados.

Busca-se um modelo que atenda os requisitos de sistemas autonômicos, ou seja, sistemas que podem se auto-adaptar aos níveis de serviço previamente estabelecidos em métricas de QoS, acordados entre os clientes e os fornecedores, e até mesmo adaptar-se a ambientes onde é possível uma mudança de comportamento para atender determinadas demandas imprevisíveis.

Esta dissertação, apresenta um modelo teórico para ajustar dinamicamente os recursos, migrando e ajustando máquinas virtuais, a fim de balancear a carga de trabalho de maneira eficiente. São propostas técnicas simuladas para detectar automaticamente a formação dos centros de dados, encontrando máquinas ociosas e sobrecarregadas, a fim de determinar um ajuste no balanceamento de carga, alocando e distribuindo recursos para atender um modelo eficiente de trabalho.

1.4 OBJETIVOS DA PESQUISA

Este trabalho tem como objetivo geral propor uma nova estratégia para alocação e distribuição de máquinas virtuais em ambientes de Computação em Nuvem Verde.

Considerando que a entrada de carga do ambiente é estimada (por valores históricos e projeção de aumento), mas inerentemente imprevisível, o problema está relacionado ao comportamento da saída em relação a configuração do sistema quando a carga extravasar a estimativa acima de uma margem pré-configurada. O trabalho verifica se é possível garantir a performance de saída do sistema pela reconfiguração *on line* da distribuição e alocação de máquinas virtuais (isto é, configuração interna).

A solução intuitiva de alocar máquinas virtuais com mais recursos para

resolver problemas de gargalos de processamento precisa considerar os efeitos colaterais do processo de reconfiguração (por exemplo, atrasos, interrupção de serviços, alocação de recursos físicos, entre outros) a fim de que o impacto inicial do ato de reconfiguração não impossibilite o ganho de longo prazo do processo de reorganização.

Assim chega-se ao problema principal da pesquisa que deve ser respondido ao final do trabalho: Qual o modelo de alocação e distribuição de máquinas virtuais em ambientes de Computação em Nuvem Verde (isto é, configuração interna) que garante níveis de performance de serviço contínuos e aceitáveis em resposta a variações de carga imprevisíveis?

1.4.1 Objetivos Específicos

Como objetivos específicos deste trabalho, podemos citar:

- Demonstrar e explicar o impacto de variações imprevisíveis da carga do sistema sobre as diversas configurações de distribuição e alocação de máquinas virtuais;
- Explicar as técnicas associadas à monitoração de performance na saída do sistema;
- Explicar os problemas práticos associados à reconfiguração *on line* de distribuição e alocação de máquinas virtuais;
- Pesquisar o equilíbrio entre distribuição e alocação de máquinas virtuais, variações de carga e qualidade de serviço;
- Demonstrar em um número de cenários;
- Propor um método estimativo para o "ponto de equilíbrio".

1.4.2 Resultados Esperados

Como resultados esperados deste trabalho, podemos citar:

1. Um modelo de configuração e estimativa de distribuição e alocação de máquinas virtuais em ambientes de Computação Verde, visando o equilíbrio entre recursos e qualidade de serviço na saída do sistema;
2. Simulação de várias configurações de ambiente, recursos disponíveis e configurações internas que demonstrem o impacto na resposta e nos custos de operação associados;

3. Validação desse modelo através da implementação de uma prova de conceito num ambiente controlado e/ou comparação de resultados em uma configuração já existente.

1.5 ORGANIZAÇÃO DO TRABALHO

Este trabalho está dividido da seguinte forma:

- O primeiro capítulo apresenta uma introdução ao tema, contexto e problema da pesquisa, citando suas questões e objetivos, demonstrando desta forma a caracterização do problema, os objetivos do trabalho, além da sua justificativa, mencionado as perguntas da pesquisa e finalizando com a organização do estudo.

- O segundo capítulo apresenta a contextualização. Neste capítulo são apresentadas as principais definições necessárias ao desenvolvimento da pesquisa, como os conceitos de Computação em Nuvem, Computação Verde, Computação em Nuvem Verde, suas diferenças e mencionando sobre o estado da arte, bem como conceitos de Teoria da Organização e Computação Autonômica.

- O terceiro capítulo é referente a esta proposta de pesquisa utilizando a Teoria da Organização como método e a caracterização de seu objeto de estudo tendo em vista as soluções propostas, em termos de sistemas mais abrangentes do modelo de alocação e distribuição de máquinas virtuais;

- O quarto capítulo apresenta a solução desenvolvida e os resultados obtidos que introduzem a idéia de composição do método da Teoria da Organização para o desenvolvimento do modelo proposto;

- O quinto capítulo aborda as considerações finais, mostrando como e onde foram alcançados os objetivos, descreve as limitações da pesquisa e propõe novas direções de estudo em função da complementação do problema enfocado.

2 CONCEITOS E TRABALHOS RELACIONADOS

Neste capítulo são descritos os conceitos relacionados ao trabalho e essenciais para o seu desenvolvimento, assim como os principais trabalhos relacionados à pesquisa realizada.

Para entender melhor a computação em nuvem, primeiramente é necessário entender as três principais estruturas nas quais estas podem ser implementadas: a Arquitetura Convencional, a Computação em Nuvem e a Computação em Nuvem Verde.

A Figura 2 apresenta uma visão geral do funcionamento destas estruturas, três arquiteturas possíveis para a formação de um centro de dados, onde são relacionados os serviços e os cenários.

Nas arquiteturas visualizadas, a carga de entrada do ambiente pode ser formada por requisições de Serviços *Web*, sítios da *Internet*, aplicações de *intranet*, banco de dados, Serviços de *Backup*, aplicações de *backup* e Serviço de *Boot* Remoto, sistemas operacionais, assim como todos os serviços disponíveis na empresa que utilizam o processamento e armazenamento pela rede. Estes foram escolhidos por serem serviços comuns no ambiente de TI e por apresentarem peculiaridades na distribuição das suas utilizações. Desta forma num centro de dados podemos ter diversas aplicações gerando cargas de diferentes tipos e variações imprevisíveis, variações bruscas que podem fugir ao controle do processamento do ambiente, sendo um ambiente muito heterogêneo, sem possibilidade de controle histórico.

As estruturas podem ser comparadas, visualizados de forma geral e de maneira hierárquica, da mesma forma que é modelado um ambiente de uma organização.

Inicialmente tem-se o modelo de computação convencional, onde cada aplicação roda em um servidor ou banco de servidores específicos, não importando sua carga de processamento.

Em segundo lugar é visualizada uma estrutura para formação de um centro de dados no modelo de computação em nuvem, onde a carga para o conjunto de aplicações é distribuída entre MVs - Máquinas Virtuais em execução em servidores físicos compartilhados por todas as aplicações.

No terceiro modelo de arquitetura, o modelo de Computação em Nuvem Verde, semelhante ao modelo de computação em nuvem, onde a carga para o conjunto de aplicações é distribuída entre MVs em execução em servidores físicos compartilhados agrupados em *clusters*, porém com o argumento que os *clusters* ociosos, podem ser auto-gerenciados, ou seja, ocorrer uma melhor migração, alocação e distribuição de máquinas virtuais, para poupar-em recursos e gerar uma melhor eficiência energética.

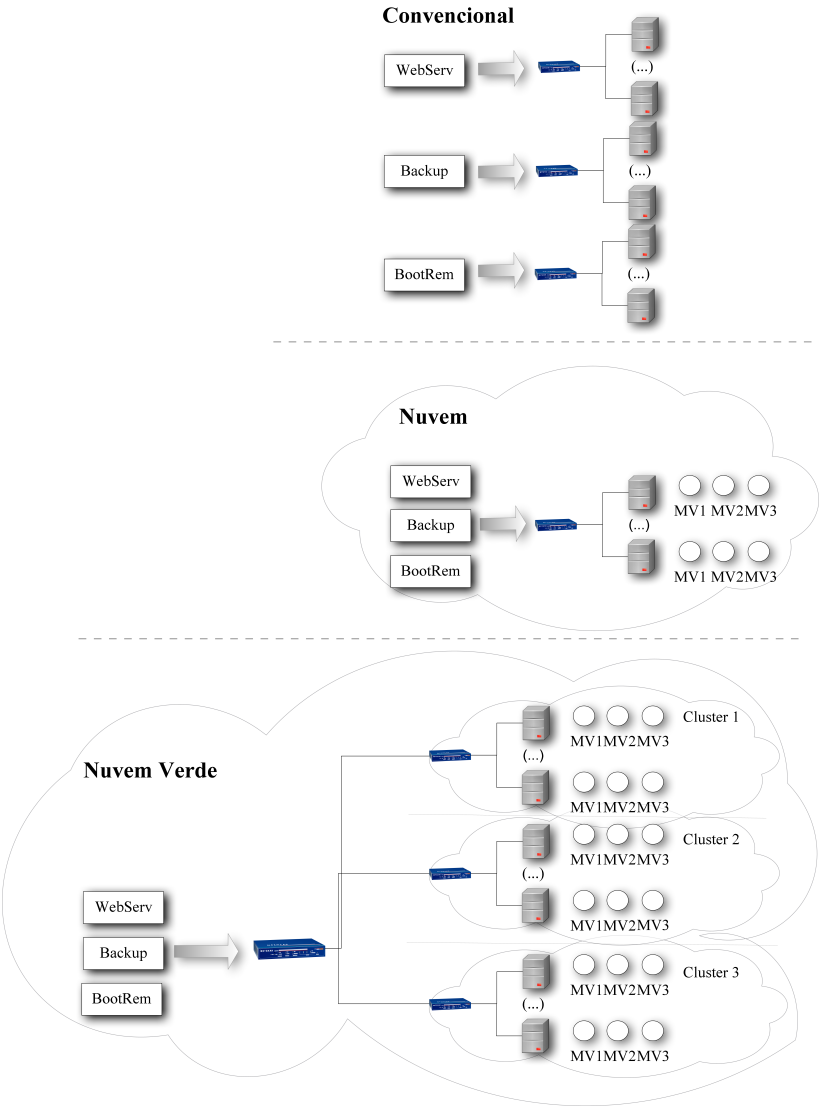

Figura 2 – Modelos de Centro de Dados.

Através dos modelos são definidas de maneira ordenada algumas características importantes dos sistemas, para a modelagem de um ambiente

genérico, buscando o ponto de equilíbrio para uma melhor performance de um ambiente real, considerando na ordem, a entrada do ambiente, o processamento proposto, ou seja uma configuração interna e uma configuração externa, e a saída ótima do sistema.

Abaixo são definidos alguns escopos para avaliação e comparação entre as três diferentes estruturas citadas anteriormente:

Flexibilidade: Refere-se a alocação de recursos e serviços, numa orquestração física e orquestração de serviços onde existe a possibilidade de variação de configuração das máquinas que hospedam os serviços e a variação da configuração dos serviços providos pela infraestrutura;

Disponibilidade: Refere-se ao comportamento da estrutura para com a distribuição da carga dos serviços (balanceamento de carga) e aos mecanismos da estrutura para garantir a operação dos serviços sem interrupções (alta disponibilidade);

Custo: Refere-se aos gastos (CAPEX) de capital (inicial e ocasional) para a existência da estrutura e aos gastos operacionais (OPEX) mensais para que a estrutura persista;

Sustentabilidade: Refere-se aos esforços para redução de consumo, assim como ao poder que a estrutura tem sob o ambiente (sistema de refrigeração, de energia) a sua volta.

2.1 DEFININDO A ARQUITETURA CONVENCIONAL

Um sistema distribuído, definido neste trabalho como uma Arquitetura Convencional, onde cada computador ou conjunto de computadores agrupados como um sistema único, independente para compartilhar recursos.

Abaixo são listados alguns conceitos, relacionados durante a pesquisa sobre a Computação Distribuída, a Arquitetura Convencional:

> Uma coleção de computadores independentes que se apresenta ao usuário como um sistema único e consistente (TANENBAUM, 2003).
> Uma coleção de computadores autônomos interligados através de uma rede de computadores e equipados com *software* que permita o compartilhamento dos recursos do sistema: *hardware*, *software* e dados (COULOURIS; DOLLIMORE; KINDBERG, 2007).
> O termo "sistemas de computação distribuída"reflete uma grande classe de sistemas de computação. A palavra "distribuídos"refere-se ao fato de que a lógica

> de processamento, funções, dados, controle, ou uma combinação desses do sistema de computação são distribuídos para um determinado ponto (YAU; YANG; SHATZ, 1981).

O suporte completo de um sistema de banco de dados distribuídos, por exemplo, implica que uma única aplicação seja capaz de operar de modo transparente sobre dados dispersos em uma variedade de banco de dados diferentes, gerenciados por vários SGBD (Sistema de Gerenciamento de Banco de Dados) diferentes, em execução em uma variedade de máquinas diferentes que podem estar rodando em diversas plataformas heterogêneas e uma variedade de sistemas operacionais. Onde o modo transparente diz respeito à aplicação operar sob um ponto de vista lógico como se os dados fossem gerenciados por um único SGBD, funcionando em uma única máquina com apenas um sistema operacional (TANENBAUM; STEEN, 2006).

Assim, a computação distribuída consiste em adicionar o poder computacional de diversos computadores interligados por uma rede de computadores ou mais de um processador trabalhando em conjunto no mesmo computador, para processar colaborativamente determinada tarefa de forma coerente e transparente, ou seja, como se apenas um único e centralizado computador estivesse executando a tarefa. A união desses diversos computadores com o objetivo de compartilhar a execução de tarefas, é conhecida como sistema distribuído (TANENBAUM; STEEN, 2006).

Um sistema de computação distribuída possui algumas características, como por exemplo, o sistema tem um número de processadores, existem canais de comunicação entre os processadores, há uma série de componentes funcionais em cada processador e existem interações entre os componentes funcionais. Estas características são essenciais para o compartilhamento de recursos, recursos de *hardware* e *software*, além da sincronização de execução dos componentes (YAU; YANG; SHATZ, 1981).

O projeto de sistemas distribuídos passa ainda pela divisão de responsabilidade entre os componentes do sistema (isto é, aplicação, servidor e outros processos) e a localização dos componentes nos computadores na rede. Visando assim garantir a performance, a confiabilidade e a segurança do sistema resultante (TANENBAUM, 2003). Os principais modelos de arquitetura nos quais esta distribuição de responsabilidades é baseada são: o modelo cliente-servidor, o modelo P2P (do inglês, *Peer-to-peer*) e o modelo objetos distribuídos (TANENBAUM; STEEN, 2006).

Basicamente os modelos de computação distribuída, tratam de distribuir o processamento de uma aplicação por diversas máquinas físicas, resolvendo problemas da computação centralizada (por exemplo, sistema inacessível devido uma máquina estar inacessível). Com isso, não há a preocupação

em redução de custos, redução de máquinas e equipamentos, contudo garante a disponibilidade do sistema.

Um exemplo de sistema distribuído é o programa Folding@home da universidade de Stanford. O objetivo é o estudo da estrutura das proteínas relacionadas com a cura de doenças como Alzheimer, Câncer, Parkinson entre outras. No página do projeto na *Internet*, qualquer pessoa pode ajudar no processamento dos dados do estudo, efetuando o *download* da aplicação que será instalada como um protetor de tela no seu computador. Este programa recebe os dados pela *Internet*, faz o processamento enquanto seu computador está ligado mas não está em uso, enviando as informações processadas para a origem ao final do processo (COULOURIS; DOLLIMORE; KINDBERG, 2007).

2.1.1 Funcionamento da Arquitetura Convencional

Nesta estrutura cada serviço possui um ou mais servidores alocado para sua utilização. Tendo suas aplicações concentradas em apenas um servidor ou distribuíveis por N servidores físicos (*clusters*).

Abaixo classifica-se os quatro escopos de acordo com as características da Arquitetura Convencional:

Flexibilidade: Alterações no ambiente demandam desligamento das máquinas físicas e adição de componentes físicos. Aprimoramentos da estrutura exigem compra de equipamentos. No aspecto da orquestração de serviços, limita-se a aspectos internos do serviço, como trocas de mensagens (por exemplo, *OpenMPI*, *RMI*, *WebService*);

Disponibilidade: A carga é distribuída como requisição, se o serviço de suporte permitir. Existe a redundância a falhas implementadas ao nível do Serviço ou Rede, com equipamentos redundantes ou pela exclusão de *proxy*;

Custo: Requer muitas máquinas físicas, relação de 1x1 entre Serviços X Máquinas Físicas (GRUBER, 2009). Redundância aumenta ainda mais a estrutura. Grande demanda de equipamentos de refrigeração. Grandes números de máquinas físicas e heterogêneas demandam mais manutenção, conseqüentemente mais homens/hora;

Sustentabilidade: Limita-se a redução de ciclo de relógio de CPU (por exemplo, abordagem DFS e DVFS) e desligamento de periféricos (por exemplo, HD (do inglês, *Hard Disk*) e Monitor);

Assim no modelo proposto neste trabalho de pesquisa, é considerada uma Arquitetura Convencional, como ilustrado na Figura 3, onde cada serviço é executado em um *pool* de servidores físicos, sem qualquer método de virtualização, que pudesse economizar recursos físicos para uma melhor otimização do ambiente. A Figura 3 ilustra três aplicações (isto é, *WebServ*, *Backup* e *BootRem*), sendo que cada uma delas esta relacionada a um grupo de equipamentos específico (isto é, equipamentos de rede, máquinas físicas), uma estrutura que atende os picos de carga, sem mobilidade.

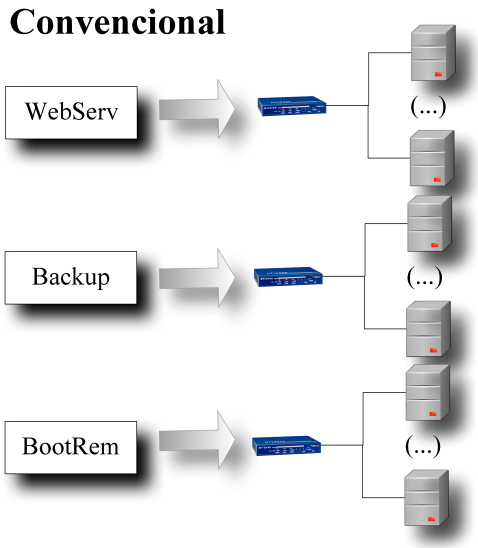

Figura 3 – Modelo de Arquitetura Convencional.

2.1.2 Trabalhos Relacionados de Arquitetura Convencional

A pesquisa analisou os principais trabalhos relacionados à computação distribuída, envolvendo a parte de gerenciamento de recursos e a eficiência no consumo de energia.

Em (PINHEIRO et al., 2001) foi proposto uma técnica para gerenciar um *cluster* de máquinas físicas, com o objetivo de minimizar o consumo de energia, mantendo os níveis de QoS. A principal técnica para minimizar o consumo de energia é a concentração de carga, ou balanceamento, mudando

de computação ociosa a nós desligados. A abordagem requer lidar com o desempenho de energia *trade-off* (ou seja, medidas estratégicas), como o desempenho de aplicações que podem ser degradadas, devido à consolidação da carga de trabalho. Os autores usam a taxa de transferência e tempo de execução de aplicações como as restrições para garantir a QoS. O algoritmo monitora periodicamente a carga e decide quais nós devem ser ligados ou desligados para minimizar o consumo de energia pelo sistema. Na avaliação do desempenho, os autores aplicam uma noção da demanda por recursos, que incluem recursos de CPU, disco e interface de rede. Esta noção é utilizada para prever a degradação do desempenho e *throughput* devido à carga de trabalho de migração baseadas em dados históricos. No entanto, a estimativa de demanda é estática - a previsão não considera possíveis mudanças na demanda ao longo do tempo. Sendo assim um modelo ultrapassado, pois não economizaria em plenitude os recursos do ambiente, ou seja, não é um modelo eficiente.

2.2 DEFININDO A COMPUTAÇÃO EM NUVEM

Nem mesmo a tradução da palavra do inglês "*Cloud Computing*"tem um consenso no Português, alguns chamam de Computação na Nuvem, ou Computação em Nuvem, ou ainda Nuvem Computacional. Sobre o conceito Computação na Nuvem, não é encontrada uma definição padrão, não existindo consenso sobre o tema. A Computação em Nuvem utiliza muitos conceitos já trabalhados no mundo acadêmico e empresarial, e é um novo paradigma que tem se mostrado muito desafiador e revolucionário principalmente para aplicações que demandam muito poder de processamento, memória e armazenamento. Abaixo são listados alguns conceitos traduzidos, relacionados durante a pesquisa de Computação em Nuvem:

> Computação em nuvem refere-se tanto a aplicações entregues como serviços pela *Internet* e os sistemas de *hardware* e *software* em *data centers* que oferecem esses serviços. Os próprios serviços têm sido referidos como *Software* como Serviço (SaaS). O *hardware* e *software* do *data center* é o que chama-se uma nuvem. (ARMBRUST et al., 2009).
>
> *Cloud computing* é o próximo passo natural na evolução da oferta de serviços em tecnologia da informação e produtos. Para uma grande extensão *cloud computing* será baseada em recursos virtualizados. (...) A computação em nuvem abraça infraestrutura cibernética e baseia-se em décadas de pesquisa em virtualiza-

> ção, computação distribuída, computação em grade, *utility computing*, e mais recentemente em rede, *Internet* e serviços de *software* (VOUK, 2008).
> Um paradigma da computação distribuída em grande escala que é conduzido por economias de escala, em que um conjunto de abstração virtualizada, dinamicamente escalável, gerência do poder da computação, armazenamento, plataformas e serviços são fornecidos sob demanda para clientes externos através da *Internet* (FOSTER et al., 2008).
> (...) A computação em nuvem é um negócio emergente e do conceito de tecnologia com significados diferentes para pessoas diferentes. Para aplicações e usuários de TI, é a TI como um serviço (ITaaS) - isto é, a entrega da computação, armazenamento e aplicações através da *Internet* a partir de centros de dados centralizados. Para os desenvolvedores de aplicativos da *Internet*, é uma *Internet* escalável, uma plataforma de desenvolvimento de *software* e ambiente de execução. Para os fornecedores de infraestrutura e administradores, as infraestruturas dos centro de dados distribuídos estão conectadas por redes IP (LIN et al., 2009).
> Computação em nuvem é um modelo que possibilita acesso, de modo conveniente e sob demanda, a um conjunto de recursos computacionais configuráveis que podem ser rapidamente adquiridos e liberados com mínimo esforço gerencial ou interação com o provedor de serviços (NIST, 2009).
> Uma nuvem é um tipo de sistema paralelo e distribuído que consiste de uma coleção de computadores inter-ligados e virtualizados que são dinamicamente provisionados e apresentados como um ou mais recursos de computação unificado baseado em acordos de nível de serviço estabelecido através de negociação entre o prestador de serviços e consumidores (BUYYA; YEO; VENUGOPAL, 2008).

Definições para a computação em nuvem podem variar. Do ponto de vista prático: Computação em Nuvem é o acesso sob demanda, virtualizando os recursos de TI que estão armazenados fora do seu centro de dados, compartilhado por outros, simples de usar, pago através de subscrição, e acessado através da *Internet*. Do ponto de vista acadêmico: A computação em nuvem refere-se a ambos os pedidos, entregues como serviços pela *Internet* e os sistemas de *hardware* e *software*, estão nos centros de dados que fornecem

estes serviços. O centro de dados *hardware* e *software* é o que chamamos de uma nuvem. Quando uma nuvem está disponibilizada em um modelo pague-pelo-uso para o público, esta é uma nuvem pública, o serviço que está sendo vendido é a computação utilitária, ou seja modelo sob demanda. Ambas as definições implícita ou explicitamente usam o modelo "utilitário"que incorpora a lógica do abastecimento de água, redes elétricas ou sistemas de esgoto. Este modelo é onipresente. O modelo de computação em nuvem possui muitos pontos fortes, porém não é ainda um modelo consolidado, livre de problemas (BRYNJOLFSSON; HOFMANN; JORDAN, 2010).

Com base nessas definições, são visualizadas algumas características comuns, ou seja, é um novo paradigma de computação distribuída para oferecer todo o poder de computação, *software* e armazenamento, até mesmo uma infraestrutura de centro de dados distribuídos sob demanda. A computação em nuvem faz uso das tecnologias existentes, tais como virtualização, computação distribuída, computação em grade, a utilidade de computação e *Internet*.

Esse paradigma tem como objetivo disponibilizar estruturas confiáveis, seguras e auto sustentáveis para aplicações e serviços na *Internet*, baseado no compartilhamento de dados e recursos computacionais na rede. Estas aplicações e serviços podem ser elásticos de acordo com a necessidade no usuário, sem que ele se preocupe com isso, seguindo o conceito de utilidade da computação (do inglês, *utility computing*) (LLORENTE et al., 2006) como um modelo de serviço de provisionamento que prevê acesso aos recursos computacionais com adaptabilidade, flexibilidade e simplicidade, permitindo um modelo de computação "pague-por-uso"(*pay-per-use*) semelhante aos tradicionais serviços públicos como o de água ou de eletricidade. Este modelo fornece suporte inerente às idéias de terceirização e disponibilidade no ato (*on demand*) de recursos, o que é desejável para a criação de serviços compostos em ambientes dinâmicos.

Com o desenvolvimento tecnológico, e a busca de um melhor relação custo/benefício na utilização de infraestruturas de TI, o conceito de Computação em Nuvem tornou-se uma inovação tecnológica fundamental. Essa inovação tecnológica tem propiciado a implementação de soluções criativas para desafios de economia de energia elétrica, recursos de *hardware* e espaço, não impactando em funcionalidades de equipamentos.

Dentre os benefícios da utilização de sistemas na Nuvem Computacional, estão a redução do espaço físico local, devido à redução da infraestrutura de TI, tanto de servidores e *softwares*, quanto de equipamentos de suporte, de energia, de climatização e de recursos humanos. Neste ponto a Computação em Nuvem, contribui até mesmo para a Computação Verde (isto é, *Green Computing*), tendência que vem surgindo nos últimos anos devido à necessi-

dade cada vez mais crescente de economia dos recursos naturais e controle do clima mundial (KURP, 2008), reduzindo o impacto de TI ao meio ambiente. A tendência seria o aumento de empresas com foco no fornecimento de todos os recursos de infraestrutura necessários para o ambiente computacional, desde *hardware* (por exemplo, processamento, memória, armazenamento) até aplicações (por exemplo, sistemas operacionais, serviços, ambientes de desenvolvimento), de forma que os usuários possam utilizar toda a tecnologia sem se preocupar com plataforma, aplicativos, serviços computacionais, atualizações de *hardware* ou *softwares*, disponibilidade, etc. A Computação em Nuvem ainda permite a flexibilização, a otimização no gerenciamento de TI, a garantia de disponibilidade e a segurança para o ambiente.

2.2.1 Tipos de Serviços na Computação em Nuvem

Os serviços disponíveis na Nuvem Computacional seguem uma classificação de acordo com modelos, abaixo são descritos os tipos de serviços mais comuns:

Infraestrutura como Serviço (IaaS) - proporcionam um "computador completo", toda uma infraestrutura de *hardware* com todos os seus recursos necessários através da *Internet*;

Plataforma como Serviço (PaaS) - neste caso, é oferecida toda ou uma parte de um ambiente, sistema operacional, inclusive no caso para desenvolvimento, de forma que os usuários podem acessar *on line*, individualmente ou em grupo;

***Software* como Serviço** (SaaS) - é disponibilizada uma aplicação completa pela *Internet*, para que o usuário possa utilizar de acordo com sua necessidade, sem precisar se preocupar com custos de licença e atualização.

2.2.2 Definições de Estrutura de Computação em Nuvem

Quanto à estrutura disponível para os ambientes de computação em nuvem, a literatura divide quatro diferentes tipos, tratados como de modelos de implantação, conforme definido em (NIST, 2009), privado, público, comunidade e híbrido. Podemos dizer que esta classificação trata da restrição de acesso do negócio, assim os requerimentos de segurança nos diferentes cenários podem e devem ser diferenciados.

Privado - neste modelo temos uma infraestrutura de nuvem que é utilizada apenas para uma organização, sendo esta nuvem local ou remota e administrada pela própria empresa. Um exemplo deste modelo poderia ser o cenário de uma empresa particular e seus diversos departamentos. A empresa poderia ter diversos servidores de arquivos, por exemplo, espalhados pelos departamentos, porém quando um determinado setor necessitar de um processamento maior, por exemplo, para gerar a folha de pagamento, todos os equipamentos poderiam se ajudar entre si, contribuindo para a eficiência da tarefa.

Público - neste caso de implantação público, a infraestrutura da computação em nuvem é disponibilizada para o público em geral, sendo acessado por qualquer usuário. Um exemplo seriam as aplicações do Google, Gmail, Google Docs, etc.

Comunidade - uma estrutura de computação em nuvem é considerada como comunidade, quando ocorre o compartilhamento por diversas organizações de uma nuvem. Um exemplo pode ser um grupo de universidades que cooperam entre si, para pesquisa de assuntos que demandam determinado processamento e compartilham suas estruturas para melhorar a velocidade das pesquisas.

Híbrido - no caso do modelo de estrutura híbrido, ocorre uma composição de duas ou mais nuvens, de qualquer tipo, que devem ser interligadas por uma tecnologia padronizada que possibilite a portabilidade de dados e aplicações, observando suas políticas de segurança.

2.2.3 Exemplos de Aplicações de Computação em Nuvem

As aplicações comerciais na nuvem computacional são muitas, de diversos tipos e características diferentes, existem diversos provedores, fornecendo diferentes produtos, temos como exemplo, *softwares* de colaboração - Box.net, Dropbox, *softwares* de desenvolvimento - Google Apps, Zoho, Parallels, *softwares* de rede social - Ning, Amitive (ALLIANCE-CSA, 2009).

2.2.4 Funcionamento da Computação em Nuvem

Esta seção descreve a estrutura mais comum de implementação de nuvem, baseia-se na funcionalidade de virtualização de servidores, onde há uma camada que abstrai os recursos dos servidores físicos e os apresenta como um conjunto (de recursos) a serem compartilhados pelas máquinas virtuais.

Estas, por sua vez, processam os serviços hospedados podendo compartilhar recursos em comum.

Abaixo são descritas suas características, conforme as definições apresentadas na Seção 2:

Flexibilidade: Possibilita rápida reconfiguração lógica das MV com configurações antes proibitivas. O aprimoramento da estrutura exige compra de novos equipamentos físicos. No processo de MVs os serviços podem ser agrupados por contextos, a fim de melhorar alguns aspectos, como a melhor rede ou recursos de E/S (isto é, Entrada / Saída);

Disponibilidade: Tem a mesma funcionalidade da computação verde, acrescido do balanceamento de carga que são estendidas para o nível do servidor. Se necessário, as MVs podem ser movidas e processadas em máquinas físicas exclusivas dentro do centro de dados, encapsulando e garantindo alguns recursos. Estratégias de redundância a falhas podem ser implementadas em um nível mais baixo, no nível do servidor. Pode-se fazer um espaço instantâneo no *status* do sistema, para posterior restauração. Ou, na situação de *backup*, se faz uma cópia da MV para o propósito de armazenamento. Em um cenário de armazenamento, a redundância, alta disponibilidade, pode ser implementada para ligar um servidor, que tem uma máquina virtual desativada (para qualquer finalidade), em outro servidor;

Custo: O desafio é reduzir, para reduzir o número de máquinas físicas necessárias. Neste novo cenário com poucas máquinas físicas, o desafio é ter robustez de equipamentos, especificamente recursos de E/S e de rede. Com isso, torna-se necessário a existência de profissionais capazes de fazê-lo. No entanto, pode-se reduzir a necessidade de refrigeração. Entende-se também que com a redução do número de máquinas físicas e a facilidade de gerenciamento se necessitem menos homens/hora, porém capacitados;

Sustentabilidade: A redução na quantidade de máquinas físicas impacta na redução de equipamentos de refrigeração e reduz duplamente o consumo de energia, pelos servidores e pelo sistema de refrigeração, consumindo menos BTU/hora (do inglês, *British Thermal Unit*).

Este trabalho considera um modelo de computação em nuvem, como sendo um modelo onde a arquitetura esta toda em um *cluster* somente, ilustrado na Figura 4, onde todos os serviços são processados por máquinas virtuais dentro dessa estrutura, sem a preocupação com otimização do ambiente para economia de energia. Na Figura 4 são ilustradas as três aplicações (isto

é, *WebServ*, *Backup* e *BootRem*), sendo todas relacionadas um mesmo grupo de equipamentos (isto é, equipamentos de rede, máquinas físicas), sendo que sua carga é processada por máquinas virtuais, reduzindo assim o número de equipamentos necessários ao ambiente.

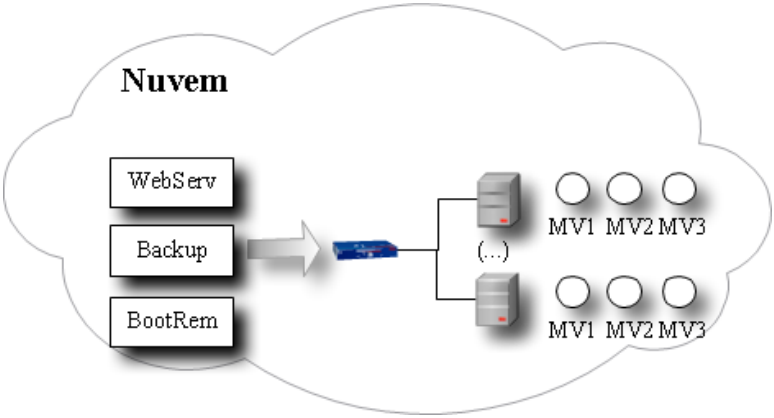

Figura 4 – Modelo de Computação em Nuvem.

2.2.5 Trabalhos Relacionados de Computação em Nuvem

Nesta Seção são descritos os principais trabalhos relacionados à computação em nuvem, analisando de acordo com a proposta desta pesquisa.

O trabalho descrito por (BUYYA; RANJAN; CALHEIROS, 2010) sugere a criação de um ambiente federado de computação em nuvem (InterCloud), formando um ambiente de computação que suporta expansão ou contração dinâmica das capacidades (máquinas virtuais, serviços, armazenamento e banco de dados) para o tratamento de variações bruscas na demanda de serviços. O estudo se baseia ao ponto de que os atuais fornecedores de computação em nuvem tem vários centros de configuração de dados em diferentes localizações geográficas pela *Internet* e que os sistemas existentes não suportam mecanismos e políticas para a dinâmica de coordenação de distribuição de carga entre os diferentes centros de dados baseados em nuvem, a fim de determinar a localização ideal para serviços de hospedagem de aplicação para atingir níveis razoáveis de QoS.

(CALHEIROS et al., 2009) desenvolveu um *framework* para simulação de Computação em Nuvem, é extensível, permite a modelagem de novas

infraestruturas e a gestão de serviços. Possui quatro características importantes: (i) apoio à modelagem e instanciação de grandes infraestruturas de computação em nuvem, incluindo centros de dados em um único nó de computação física e *Java Virtual Machine*, (ii) uma plataforma independente para modelagem de centros de dados, flexibilidade de tamanho e prioridade de serviços, agendamento e políticas, (iii) a disponibilidade do motor de virtualização, o que ajuda na criação e gestão de serviços virtualizados em um nó central de dados, e (iv) para alternar entre o espaço compartilhado e tempo compartilhado na atribuição de núcleos de processamento a serviços virtualizados.

No trabalho de (CHANG; CHOU; RAMAKRISHNAN, 2009) propõe-se a aplicação da computação em nuvem para atender à demanda de um vasto conjunto de cenários de serviços de saúde, em particular, o conceito de infraestrutura compartilhada e serviços fornece a fundação para a defesa dos ecossistemas de serviços de saúde. Busca conceitos para uma abordagem do ecossistema de modo a identificar requisitos de alto nível na tecnologia de computação em nuvem para proporcionar ambientes de hospedagem para ecossistemas de saúde sustentável.

Em (ELMROTH; LARSSON, 2009) são propostas interfaces de tecnologia neural e complementos de arquitetura para a manipulação de colocação, migração, e monitoramento de máquinas virtuais em ambientes de nuvem federado. O trabalho descreve o processo de migração, destacando a requisição de migração, a migração forçada, a inicialização da transferência do contexto e a verificação da transferência. Para eles, a migração deve ocorrer motivada pelos parâmetros de execução do cliente, do seu acordo de nível de serviço - SLA. Assim, embora a MV que executa as tarefas esteja em sua plenitude, é necessário observar as outras MVs do servidor para que a execução das tarefas sejam sempre feitas com a melhor combinação de recursos de acordo com o SLA do usuário.

Em (VOORSLUYS et al., 2009), considerando a necessidade de equilibrar a carga em servidores que se encontram sobrecarregados e a necessidade de trazer os servidores para manutenção depois de migrar sua carga de trabalho para outros servidores, é feita uma avaliação do impacto dos efeitos da migração de máquinas virtuais sobre o desempenho de aplicações que funcionam dentro das MVs. Os autores fazem a avaliação baseada em uma carga de trabalho composta por modernas aplicações da *Internet*. Sugerindo que o trabalho pode ser usado pela indústria para decidir se a migração de MVs se aplica para esse tipo de aplicação.

O artigo de (D'AURIOL et al., 2009), considera a visualização de grandes informações, heterogênea e integração complexa, especialmente para as implantações tempo real facilitando a compreensão rápida levando a to-

mada de decisão. Cita conceitos de computação em nuvem para uma arquitetura que promova a integração e colaboração dinâmica de fornecedores, para aumentar o desempenho da comunicação, *on demand*.

Em (SIDDIQI et al., 2009), sugere-se um ambiente de computação móvel para monitoramento de pacientes, tipo cadastro único, disponibilizando informações em dispositivos móveis, para facilitar e economizar, não tendo papéis, exames somente *on line*.

Sandpiper (WOOD et al., 2009) propõe duas abordagens para alocar máquinas virtuais dinamicamente entre máquinas físicas: uma abordagem caixa preta que se baseia em nível de sistema apenas através de métricas e uma abordagem caixa cinza que leva em conta indicadores à nível de aplicativo, juntamente com um modelo de filas. O sistema trabalha com uma heurística iterativa que coloca a MV mais carregada na MF de menor carga.

2.3 DEFININDO A COMPUTAÇÃO VERDE

Green Computing também conhecido como *Green IT*, traduzido para o português como Computação Verde, ou ainda TI verde, se refere ao uso eficiente de recursos computacionais minimizando o impacto ambiental, maximizando sua viabilidade econômica e assegurando os deveres sociais. Através dela pode-se dizer que no futuro as ações tecnológicas irão prejudicar o mínimo possível o meio ambiente e serão mais sustentáveis. Quando discutido sobre TI verde o consumo de energia sempre vem em primeiro lugar, pois é o fator que mais gera despesas dentre os tópicos pesquisados nesta área (SHULZ, 2009). Por isso existe a preocupação para achar técnicas de se fazer as mesmas coisas gastando menos energia como, por exemplo, a virtualização de servidores ou até mesmo a troca do parque tecnológico.

Os objetivos da computação verde são semelhantes aos da química verde, para reduzir o uso de materiais perigosos, maximizam a eficiência da energia durante a vida do produto, e promove a reciclagem ou biodegradabilidade de produtos e resíduos da fabricação. As pesquisas continuam em áreas chave, para a utilização de computadores com o mínimo gasto energético, com a concepção de algoritmos e outras tecnologias relacionadas com a eficiência (MURUGESAN, 2008). Um aspecto importante do sucesso ou fracasso tecnologia renovável é a tecnologia de armazenamento e sua eficiência (MCLAUCHLAN; MEHRUBEOGLU, 2010).

Abaixo são listados alguns conceitos, relacionados durante a pesquisa sobre a Computação Verde:

> TI Verde é o estudo e a prática do projeto, da fabricação, da utilização e do descarte de computadores, ser-

> vidores e subsistemas associados (por exemplo, monitores, impressoras, dispositivos de armazenamento e de rede e sistemas de comunicações), tratando cada um destes passos de forma eficiente e eficaz com um mínimo ou nenhum impacto sobre o meio ambiente (MURUGESAN, 2008).
> Computação Verde refere-se ao uso ambientalmente responsável de computadores e recursos relacionados. Essas práticas incluem a implementação da eficiência energética nas CPU's, servidores e periféricos, bem como o consumo reduzido de recursos e destinação adequada de lixo eletrônico (HARRIS, 2008).
> Computação Verde refere-se a práticas sustentáveis (LEONHARD; MURRAY, 2009).

Empresas, governos e sociedade em geral têm uma tarefa importante, abordar as questões ambientais e adotar práticas ambientalmente saudáveis. Ao longo dos anos, o uso de TI explodiu em diversas áreas, melhorando a nossa vida, o trabalho e oferecendo conveniência, juntamente com vários outros benefícios. No entanto, esta adoção em massa de TI, tem contribuído para o aumento dos problemas ambientais, embora a maioria das pessoas não perceba. Computadores e outras infraestruturas de TI consomem quantidades significativas de energia elétrica, onerando bastante as nossas redes elétricas e contribuindo com o efeito estufa com as emissões de gases. Além disso, as partes físicas, *hardware*, representam graves problemas ambientais, tanto durante a sua produção e na sua eliminação. Esta é uma parte significativa e crescente dos problemas ambientais que enfrentamos hoje. Somos obrigados a minimizar ou eliminar o possível impacto ambiental de TI para ajudar a criar um ambiente mais sustentável (MURUGESAN, 2010). Para reduzir os problemas do meio ambiente e criar um ambiente sustentável, o setor de TI, assim como cada usuário do computador deve buscar soluções eficientes para seus sistemas de TI. A sociedade esta legalmente, eticamente e socialmente precisando de produtos de TI verde, tanto de aplicativos, como serviços e práticas. TI verde beneficia o ambiente, melhorando a eficiência energética, reduzindo as emissões de gases de efeito estufa, usando materiais menos nocivos, e incentivando a reutilização e a reciclagem. Fatores como a legislação ambiental, o aumento do custo de eliminação de resíduos, imagens corporativas/públicas e a percepção dão um novo impulso a iniciativa de TI verde (SHULZ, 2009).

O acúmulo crescente de gases de efeito estufa está alterando o clima do mundo e os padrões climáticos, criando em alguns países secas e inundações, entre outros desequilíbrios ambientais. Lentamente também vem empurrando as temperaturas globais, tornando-as mais elevadas, causando problemas sé-

rios para o mundo. Por exemplo, 2005 foi o ano mais quente já registrado, e os 10 anos mais quentes ocorreram todos após 1980. Dados globais mostram que as tempestades, secas e outros desastres relacionados ao clima estão cada vez mais graves e mais freqüentes. Para parar a acumulação de gases na atmosfera, e reduzir o efeito estufa, as emissões globais terão de parar de crescer (MURUGESAN, 2008). A eletricidade é uma das principais causas da mudança climática, porque o carvão ou o petróleo, que também ajudam a gerar a eletricidade, liberam dióxido de carbono e de enxofre na atmosfera. Estas emissões podem causar desde doenças respiratórias, poluição atmosférica, chuva ácida, e até causar mudanças climáticas. Reduzir o consumo de energia elétrica é fundamental para reduzir as emissões de dióxido de carbono e seu impacto sobre o meio ambiente e o aquecimento global (HARRIS, 2008).

O impacto de TI afeta o ambiente de várias maneiras diferentes. Cada fase da vida de um computador, desde a sua produção, ao longo de sua utilização e em sua disposição, apresenta problemas ambientais. A produção de computadores, assim como de diversos equipamentos eletrônicos, consome energia elétrica, matérias-primas, produtos químicos, água, assim como geram resíduos perigosos. Todos direta ou indiretamente contribuem para o aumento das emissões de dióxido de carbono e o impacto do meio ambiente (LEONHARD; MURRAY, 2009). O consumo total de energia elétrica por servidores, computadores, monitores, equipamentos de comunicações de dados e sistemas de refrigeração dos centros de dados está em constante crescimento. Este aumento no consumo de energia resulta em aumento das emissões de gases, com efeito, de estufa. Cada computador pessoal em uso gera cerca de uma tonelada de dióxido de carbono a cada ano. Os componentes eletrônicos do computador contem materiais tóxicos (HARRIS, 2008). Cada vez mais, os consumidores se desfazem de um grande número de computadores velhos, monitores e outros equipamentos eletrônicos, isso de dois a três anos após a compra, e mais isso termina em aterros sanitários, poluindo a terra e contaminando a água. O aumento do número de computadores e sua utilização, juntamente com suas substituições freqüentes, fazem do impacto ambiental de TI uma grande preocupação. Conseqüentemente há uma crescente pressão sobre nós, a indústria de TI, empresas e indivíduos a fim de tornar o ambiente de TI em todo o seu ciclo de vida, do nascimento à morte. É de todos a responsabilidade social de proteger o nosso meio ambiente (MURUGESAN, 2008).

TI verde refere-se ao ambiente de TI. TI verde se esforça para alcançar a viabilidade econômica e melhorar o desempenho do sistema e o uso, respeitando as responsabilidades sociais e éticas. Assim, a TI verde inclui as dimensões da sustentabilidade ambiental, da eficiência energética e do custo total de

propriedade, que inclui o custo de eliminação e reciclagem (HARRIS, 2008). TI verde abrange um grande número de áreas e suas atividades, incluindo a concepção da sustentabilidade ambiental, da computação com eficiência energética, da gestão de energia, da modelagem do centro de dados, e da localização, virtualização de servidores, eliminação responsável e reciclagem, observância da regulamentação, métricas verdes, ferramentas de avaliação e metodologia, redução dos riscos relacionados com o ambiente, a utilização de fontes de energia renovável e eco-rotulagem de produtos de TI (UNHELKAR, 2011). Um número crescente de fornecedores de TI e os usuários estão se movendo em direção a TI verde, principalmente devido a aumento de impostos e/ou questões legais, e assim, ajudam na construção de uma sociedade e uma economia verde. No entanto, para construir um ambiente mais limpo, deve-se modificar ou suprimir muitas maneiras velhas e familiares de fazer as coisas e descobrir novos métodos (LEONHARD; MURRAY, 2009). Felizmente, a indústria de TI está interessada nas questões ambientais buscando novas oportunidades. Inovações no ambiente sustentável é a chave para o sucesso futuro (MURUGESAN, 2008).

O cenário competitivo das empresas, e o impacto de questões ambientais na TI das empresas, levam as empresas à busca de novas maneiras de conduzir a tecnologia e a visão de oferecer produtos e serviços, abordando questões ambientais. Por exemplo, quando da aquisição, locação ou terceirização de decisões, muitos clientes agora consideram os registros dos prestadores de serviços e iniciativas ambientais (MCLAUCHLAN; MEHRUBEOGLU, 2010). As empresas enfrentam custos mais elevados de energia, e eles também podem incorrer em taxas adicionais do governo se não abordarem as implicações ambientais em suas práticas. Os investidores e os consumidores estão começando a exigir a divulgação dos dados das empresas no que diz respeito a sua emissão de carbono, bem como suas iniciativas ambientais e realizações, e eles começaram a descontar os preços das ações das empresas que tratam mal os problemas ambientais que criam. Como resultado, muitas empresas começaram a mostrar as suas credenciais ambientais. Por exemplo, o *Carbon Disclosure Project* (www.cdproject.net) é uma iniciativa recente de petição de empresas globais para divulgar as suas emissões de carbono (UNHELKAR, 2011). Adotar práticas de TI verde oferece às empresas e indivíduos benefícios financeiros e vantagens competitivas. Operações de TI alcançam uma maior eficiência energética por meio de iniciativas verdes, que financeiramente pode beneficiá-los, especialmente quando a energia elétrica é um prêmio e ainda os preços da energia podem aumentar. As pessoas começaram a valorizar os atributos do ambiente de TI, e nos próximos cinco anos, TI verde irá tornar-se uma característica comum (LEONHARD; MURRAY, 2009). As empresas vão oferecer uma gama de novos produtos e serviços

verdes, e novas oportunidades de negócios surgirão (MURUGESAN, 2008).

2.4 FUNCIONAMENTO DA COMPUTAÇÃO VERDE

A busca por estratégias de computação verde em conjunto com a computação em nuvem surgiu a busca do conceito de Computação em Nuvem Verde.

A Nuvem Verde (*Green Cloud*) não difere muito da Computação em Nuvem (*Cloud Computing*), porém ela infere uma preocupação a mais sobre a estrutura, consumir menos energia (LIU et al., 2009) sem interferir na performance, garantindo a sustentabilidade (BUYYA; BELOGLAZOV; ABAWAJY, 2010). Paralelamente extrapola as barreiras do centro de dados, utilizando-se de recursos alheios sob demanda.

No iníco da Seção refcap2, foram definidos os escopos, abaixo classifica-se de acordo com as características da computação em nuvem verde:

Flexibilidade: Oferece as mesmas características de reconfiguração da computação em nuvem, além de poder gerenciar o *status* das máquinas físicas (ligando/desligando) quando necessário, assim como possibilita o agrupamento dos recursos, e possibilita a mobilidade da estrutura;

Disponibilidade: Oferece as mesmas características da computação em nuvem, e poderia gerenciar as máquinas físicas (hibernando) e remover máquinas virtuais ociosas;

Custo: Com a funcionalidade de movimentação de máquinas virtuais extra nuvem, centros de dados adotam uma configuração minimalista, impactando também no aumento da vida útil dos equipamentos. Já a estrutura diminui os seus custos mensais, reduzindo o consumo de energia derivada das políticas de "desligamento e hibernação";

Sustentabilidade: Com a funcionalidade de movimentação de máquinas virtuais, tem-se a possibilidade de, em períodos de baixa demanda, concentrar o processamento das máquinas virtuais em poucos servidores físicos permitindo o desligamento dos equipamentos ociosos. O sistema trabalha em conjunto ao ambiente, inferindo a estratégia de trabalho sob demanda também aos equipamentos externos (por exemplo, refrigeração e rede), assim como leva em consideração os fatores externos, como agir proativamente em caso de desastre (por exemplo, Incêndio).

Conforme ilustrado na Figura 5, nesta pesquisa considera-se um modelo de computação em nuvem verde como sendo diversos *clusters*, onde

estão agrupados diversos servidores físicos e seus equipamentos relacionados (equipamentos de rede, sistema de refrigeração, sistema de energia, entre outros equipamentos de suporte), e eles possibilitam a gerência de máquinas virtuais para o processamento de aplicações e serviços sob demanda. Neste modelo trabalha-se com a adaptação do ambiente de um modo sustentável, ou seja, com a preocupação de economizar, reutilizando recursos de uma melhor forma.

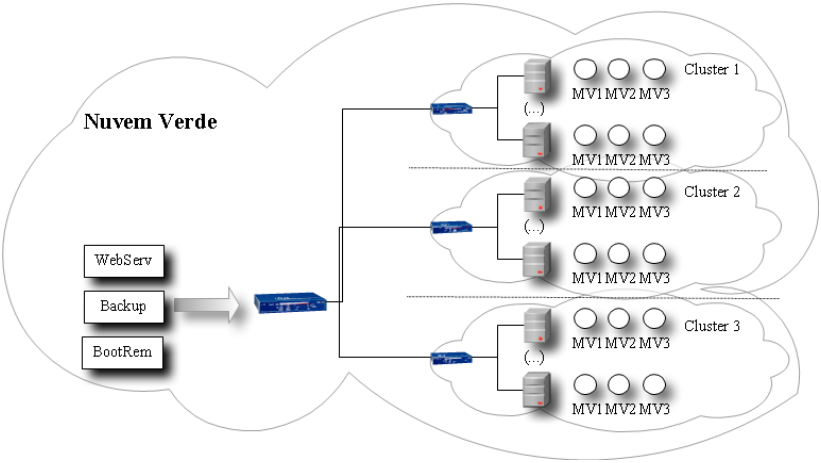

Figura 5 – Modelo de Computação em Nuvem Verde.

2.4.1 Trabalhos Relacionados com Computação Verde

Nesta seção foram analisadas algumas pesquisas realizadas envolvendo alocação de recursos para computação verde, usando diferentes conceitos.

Em (BUYYA; BELOGLAZOV; ABAWAJY, 2010), propõe que a computação em nuvem está oferecendo um modelo utilitário, orientado a serviços de TI para usuários do mundo inteiro. Baseado em um modelo "pague-por-uso", que permite a hospedagem de aplicações pervasivas tanto do consumidor, como científica e dos domínios de negócios. No entanto, os centros de dados que hospedam aplicações consomem enorme quantidade de energia, contribuindo para a alta dos custos operacionais e as emissões de carbono para o ambiente. Neste trabalho os autores criaram uma arquitetura de computação em nuvem verde, onde é feita a negociação entre as máquinas físicas e máquinas virtuais, aplicando alguns algoritmos para conseguir uma melhor

eficiência energética. Nas propostas são realizados alguns testes com três abordagens, inicialmente sem nenhuma política implementada, depois implementando as políticas de tensão dinâmica e escala de freqüência - DVFS (do inglês, *Dynamic Voltage and Frequency Scaling*), na seqüência utilizando um limite para utilização de CPU (do inglês, *Central Processing Unit*) e por fim utilizando um política de minimização de migrações. A análise foi feita comparando os resultados da saída do sistema, resultados de consumo de energia (kWh), de violações de acordo de nível de serviço - SLA (do inglês, *Service Level Agreement*) e do número de migrações de MVs. Ao final são propostas outras abordagens que ainda precisam ser estudadas, como por exemplo, otimização devido a problemas da topologia de rede virtual, basicamente tempo de resposta para a migração de MV devido a largura de banda entre os servidores ou máquinas virtuais, no caso de não estarem no mesmo centro de dados.

Em (KIM; BELOGLAZOV; BUYYA, 2009) é investigado o gerenciamento de energia, através do provisionamento de máquinas virtuais, para serviços em tempo real em ambientes de computação em nuvem. Este trabalho investiga o problema de aprovisionamento de recursos para os serviços de computação em nuvem em tempo real, a fim de minimizar o consumo de energia pela modelagem de máquina virtual em tempo real e utilizando regimes de tensão dinâmica e escala de freqüência. O trabalho propõe três regimes para o provisionamento de MV, buscando o menor custo para o usuário final e, conseqüentemente, menor consumo de energia, sendo eles, um regime com menor escala de freqüência - DFS (do inglês, *Dynamic Frequency Scaling*), um regime de DFS avançado e um regime de DFS adaptativo. Os resultados do trabalho demonstram um melhora no consumo de energia, no último regime proposto, contudo eles utilizam uma abordagem de DVFS que nem sempre pode ser adotada, pois depende da arquitetura dos servidores.

No artigo descrito por (BRANDIC, 2009), ele discute o auto gerenciamento do serviço de nuvem, alegando que em caso de falhas, mudanças ambientais e similares, os serviços devam gerir-se automaticamente, seguindo os princípios da computação autonômica e garantir os níveis de SLA pré-estabelecidos. A abordagem obteve a taxonomia da apresentação de recursos, com base no ciclo de vida de um serviço de nuvem auto-gerenciável. Além disso, apresenta uma arquitetura para a implementação dos serviços de nuvem auto-gerenciáveis, com base na mediação e negociação entre os usuários e os serviços.

Em (LIU et al., 2009) é apresentada uma arquitetura de *Green Cloud*, que visa reduzir o consumo de energia no centro de dados, que garante o desempenho, a partir da perspectiva dos usuários. A arquitetura apresentada permite acompanhamento *on line* e abrangente, realizando a migração e a

otimização de máquinas virtuais. Para verificar a eficiência e a eficácia da arquitetura proposta, foi utilizado um jogo *on line* (Tremulous), como uma aplicação de MV. Os resultados da avaliação mostram que pode-se economizar até 27% da energia ao aplicar arquitetura de Nuvem Verde. Este um dos trabalhos motivadores desta pesquisa, porém a proposta não consegue atender de forma simplificada e autonômica a arquitetura de Nuvem Verde.

Por último em (KIRSCHNICK et al., 2010) é descrita uma arquitetura que permite a implantação e gerenciamento automatizado da infraestrutura e de serviços implantados na nuvem. A arquitetura tem uma descrição do modelo de um serviço, que encapsula os requisitos, as opções, bem como o comportamento de um conjunto de recursos e orquestra o provisionamento do serviço em um recém-criado conjunto de recursos virtuais. O modelo é usado para integrar o comportamento de implantação e reconfiguração de um serviço no qual os componentes lógicos são descritos junto com opções para escalá-los adequadamente e mudar a sua configuração. Os serviços são descritos através de um conjunto de componentes, que podem ser facilmente mapeados e re-mapeados para recursos criados dinamicamente, permitindo aproveitar ao máximo os recursos da nuvem.

2.5 DEFININDO A TEORIA DAS ORGANIZAÇÕES

O conceito de combinar a Teoria das Organizações e complexos ambientes de computação distribuída não é nova, (FOSTER; KESSELMAN; TUECKE, 2001) propôs a idéia de organizações virtuais (VOs) como um conjunto de indivíduos e / ou instituições definidas por regras de interação. Neste trabalho (FOSTER; KESSELMAN; TUECKE, 2001), conclui-se que organizações virtuais têm o potencial de mudar radicalmente a forma como usamos computadores para resolver problemas, assim como a web mudou a forma de troca de informações.

O nível de organização visa desta forma fornecer as especificações de padrões organizacionais e regras, a fim de permitir uma adaptação estrutural (VÁZQUEZ-SALCEDA et al., 2010) de sistemas distribuídos ao longo do tempo. A idéia principal é que muitas das estratégias utilizadas hoje para organizar as interdependências muito complexas encontradas no comportamento humano, social, econômico será fundamental para estruturar o serviço de futuros sistemas. A fim de aplicar as técnicas e métodos existentes nos "humanos"nas sociedades, a abordagem adotada neste quadro é que um conjunto de serviços que interagem uns com os outros por algum motivo e/ou que habitam uma localidade específica pode ser considerada uma sociedade. Sociedades geralmente especificam mecanismos de ordem social, em termos

de normas e regras comuns que os membros devem aderir.

Uma organização pode ser definida como uma solução específica criada pelos intervenientes mais ou menos autônoma para atingir objetivos comuns e sub-objetivos. Ela fornece os meios para controlar a dinâmica das sociedades complexas, definindo e controlando as interações sociais entre as partes interessadas. Uma organização pode ser vista como um conjunto de entidades e suas interações, que são reguladas por mecanismos de ordem social (DIGNUM, 2004).

Uma organização é representada no nível organizacional (QUILLINAN et al., 2009), por meio do modelo de organização. Este modelo de pontos de vista de uma organização como um sistema social, descreve quais os objetivos e as preocupações que a organização possui com relação ao sistema social. Em (MOTTA, 2001) é destacado a flexibilidade da organização orgânica sob a mecanicista, bem como as definições de interação, as semelhanças e as tecnologias adotadas nos equipes de trabalho.

O modelo organizacional é especificado em termos das quatro estruturas:

- A estrutura social especifica os objetivos da sociedade, seus papéis e que tipo de coordenação rege o modelo;
- A estrutura de interação descreve momentos de interação, como *scripts* de cena, o que representa uma tarefa da sociedade que exige a ação coordenada de vários papéis, e dá uma ordenação parcial de *scripts* de cena, que especificam as interações entre os papéis destinados;
- A estrutura normativa expressa normas organizacionais (ALVAREZ-NAPAGAO et al., 2009) e regulamentos relativos aos papéis;
- A estrutura de comunicação especifica as ontologias para a descrição dos conceitos de domínio e da comunicação.

Nesta pesquisa são utilizados estes conceitos de Teoria da Organização para melhor gerenciar o ambiente, buscando um modelo que auto-gerencie automaticamente. Assim a Teoria da Organização (ALVAREZ-NAPAGAO et al., 2010; SCHMIDT et al., 2010; FOSTER; KESSELMAN; TUECKE, 2001; DIGNUM, 2004) visa fornecer os meios para descrever porque e como os elementos do ambiente de nuvem devem se comportar para atingir os objetivos do sistema global, que são (entre outros): um ótimo desempenho, redução dos custos de operação, compromisso de dependência, de acordos de nível de serviço e eficiência energética. As estruturas de organização permitem que os gerentes de rede, possam entender a interação entre os elementos dos ambientes da Nuvem, a sua influência no comportamento da organização, o impacto das ações nas estruturas macro/micro e vice-versa, e como os

processos de nível macro permitem restringir atividades a nível micro. Desta forma os modelos computacionais poderam classificar, compreender e prever essas interações e sua influência sobre o meio ambiente.

2.6 DEFININDO A COMPUTAÇÃO AUTONÔMICA

Conforme descrito em (BALEN et al., 2009), um sistema autonômico é um sistema capaz de regular seus próprios parâmetros funcionais sem atrapalhar os objetivos principais do sistema. Esse tipo de sistema é capaz de mostrar um perfeito funcionamento também em condições de estresse particulares ou de carga excessiva de trabalho. As vantagens de possuir uma gestão autonômica dos recursos e da configuração são notáveis e extremamente importantes. Essas são ainda mais evidentes em um ambiente heterogêneo e fortemente dinâmico como o ambiente de grades computacionais. Um ambiente autonômico consegue ter um melhor nível de otimização e então, desfrutar melhor os seus recursos.

Computação autonômica (HUEBSCHER; MCCANN, 2008) é uma abordagem sistemática para permitir que sistemas computacionais sejam capazes de gerenciar a si próprios sem intervenção humana. Um sistema autonômico tem quatro características básicas: auto-configuração, anula a obrigatoriedade de pessoas para ajustar propriedades no sistema de acordo com mudanças no próprio sistema ou ambiente; auto-otimização, trata a melhor configuração para utilização dos recursos; auto-cura, independe de pessoas para descobrir e recuperar ou prevenir falhas no sistema; auto-proteção, prove segurança ao sistema.

Tecnicamente, um sistema autonômico é um sistema computacional como outro qualquer, acrescido da habilidade autonômica. Os sistemas autonômicos podem ser divididos em duas partes que desempenham papéis diferentes: computação básica que utiliza o computador e a rede para resolução dos problemas dentro do domínio da aplicação; computação autonômica é responsável por fazer da computação básica a mais confiável, segura e eficiente possível. Ou seja, a computação autonômica coleta, mede, analisa os estados e comportamentos da computação básica e então decide quando e como os ajustes devem ser realizados (HUEBSCHER; MCCANN, 2008; BALEN et al., 2009).

A automatização do gerenciamento de redes e de servidores associados utilizando técnicas de computação autonômica, como ferramenta para diminuição do custo e complexidade do gerenciamento, pode prever problemas e diminuir/extinguir a indisponibilidade de serviços, além de possibilitar um melhor aproveitamento de recursos (humanos, energia, *hardware*, *software*).

Os requisitos necessários para atender a computação autonômica segundo (BALEN et al., 2009) são:

- Auto-Configuração (*Self-Configuration*): um sistema deve ser capaz de configurar-se de modo otimizado sem nenhuma intervenção externa (como a humana). A configuração deve ser dinâmica e feita de acordo com políticas de alto nível, que especificam quais características são necessárias, não o que deve ser feito para alcançá-las;

- Auto-Regeneração (*Self-Healing*): dá ao sistema a capacidade de efetuar diagnósticos sobre si mesmo sem intervenção externa, com o objetivo de encontrar eventuais erros ou sub-sistemas que não funcionam de modo correto;

- Auto-Otimização (*Self-Otimization*): o sistema deve ser capaz de otimizar-se em modo autonômico, permitindo sempre a melhor utilização possível dos recursos disponíveis. A auto-otimização permite que parâmetros funcionais do sistema sejam modificados sem intervenção externa, garantindo uma resposta positiva também no caso de situações de carga de trabalho imprevistas;

- Auto-Proteção (*Self-Protection*): o sistema deve ser capaz de antecipar e encontrar eventuais intrusões ou violações de alguns parâmetros do sistema, permitindo um funcionamento seguro e controlado de todas as próprias operações. Essa característica vai na direção da criação de sistemas sempre mais seguros e confiáveis;

- Auto-Conhecimento (*Self-Knowledged*): o sistema de ter habilidade para se conhecer, permitindo saber sobre a disponibilidade e estado dos recursos;

- Sensibilidade ao Contexto (*Context-Awareness*): o sistema deve estar consciente de suas atividades, podendo reagir a proposta de novas políticas, em acordo com seus vizinhos;

- Aberto (*Openness*): um sistema deve ser totalmente aberto, possibilitando um ambiente heterogêneo de plataformas e aplicações; e

- Antecipação (*Anticipatory*): um sistema deve ser capaz de prever e antecipar as necessidades às requisições dos usuários, otimizando a alocação de recursos.

Nesta pesquisa são utilizados estes conceitos de Computação Autonômica para melhor configurar (FOSTER; KESSELMAN; TUECKE, 2001; DIGNUM, 2004; BALEN et al., 2009), otimizar, conhecer e ter sensibilidade ao contexto ao ambiente, buscando um modelo que autonômica.

2.7 COMPARATIVO DOS TRABALHOS RELACIONADOS

Na literatura existente diversos autores buscam novas formas para gerenciamento de recursos de infraestrutura deste novo paradigma de Computação em Nuvem. Esta pesquisa vislumbra como oportunidade a adoção de conceitos da Teoria da Organização para a elaboração de um modelo que possa fazer a auto-configuração, auto-gestão, alocando e distribuindo recursos em máquinas virtuais para melhorar a eficiência e a eficácia do ambiente de computação em nuvem verde, reduzindo os custos de operação, e aumentando elasticidade, promovendo uma nova estratégia para a comunidade científica.

A Tabela 1 apresenta uma análise comparativa das características desejadas no ambiente proposto em relação aos trabalhos relacionados do (BUYYA; BELOGLAZOV; ABAWAJY, 2010) *Energy-Efficient*, do (KIM; BELOGLAZOV; BUYYA, 2009) *Power-aware*, do (BUYYA; RANJAN; CALHEIROS, 2010) *InterCloud*, do (WOOD et al., 2009) *Sandpiper*, do (PINHEIRO et al., 2001) *Load Balancing*, do (BRANDIC, 2009) *Self-Manageable*, do (LIU et al., 2009) *Green Cloud* e do (KIRSCHNICK et al., 2010) *Toward an Architecture* apresentados nas Seções 2.1.2, 2.2.5 e 2.4.1 .

	Flexibilidade	Disponibilidade	Custo	Sustentabilidade
Energy-Efficient	Parcial	Sim	Sim	Parcial
Power-aware	Não	Não	Sim	Não
InterCloud	Não	Não	Sim	Não
Sandpiper	Parcial	Não	Sim	Não
Load Balancing	Parcial	Parcial	Parcial	Não
Self-Manageable	Sim	Sim	Não	Não
Green Cloud	Parcial	Não	Sim	Sim
Toward an Architecture	Sim	Sim	Não	Não

Tabela 1 – Comparação Trabalhos Relacionados

Analisando as opções, esta pesquisa identificou uma falta na tecnologia em prover uma solução para o problema de controlabilidade, de ambientes distribuídos, provendo eficiência energética. Assim, existe uma oportunidade

de contribuir com uma abordagem diferenciada de migração e realocação dinâmicas de máquinas virtuais que resolve os problemas de controlabilidade, eficiência energética e alocação de máquinas virtuais identificados.

A pesquisa que está sendo desenvolvida nesse trabalho visa contribuir ao estado-da-arte em Computação em Nuvem Verde com essa tecnologia, e buscar uma contribuição para os itens discutidos, isto é, na flexibilidade, da disponibilidade, na redução de custo e na sustentabilidade.

2.7.1 Modelo de Computação em Nuvem Verde

O modelo *Green Cloud Computing* é uma estrutura hipotética, uma tendência desta área (DURKEE, 2010), é o objetivo de pesquisa.

Estes aspectos que são descritos abaixo são a referência para que o modelo tenha de cumprir. Na comparação de *Green Cloud*, é inferida a responsabilidade de consumir menos energia, o termo garante os acordos pré-acordados no SLA, e apresenta os dados:

Flexibilidade: Na orquestração física é onde se tem conhecimento sobre o estado de todos os equipamentos de TI. Agindo quando for necessário, não quando é preciso. Planejando suas ações com base nas informações da nuvem. Sendo capaz de prever e executar as mudanças necessárias no *hardware* de acordo com a demanda da nuvem. Como por exemplo, diminuindo ciclos de relógio da CPU quando em superaquecimento, desligando máquinas de acordo com a carga prevista, ou ativando um *backup* remoto em caso de incêndio. Assim como na orquestração de serviços o sistema é capaz de interagir automaticamente com nuvens públicas (BUYYA; RANJAN; CALHEIROS, 2010), migrando ou realocando recursos em tempo real, em nuvens remotas. Proporcionando um melhor suporte para os picos de carga de trabalho ou, ocasionalmente, os ataques de negação de serviço - DoS (do inglês, *Denial of Service*);

Disponibilidade: O sistema lida com o contexto de grupo automaticamente, sendo capaz de migrar esses grupos ou elementos de nuvens públicas, fazendo assim o balanceamento de carga. Além disso abrange um novo nível, sendo capaz de estendê-lo automaticamente, garantindo assim a alta disponibilidade;

Custo: Com a comunicação dentro da nuvem, adota-se uma configuração minimalista, garantindo o processamento local para a maioria do seu trabalho, deixando os picos de carga de trabalho para uma nuvem externa. Por ter um gerenciamento automatizado, baseado em experiên-

cias anteriores e os resultados, pode-se gerir com interferência humana mínima. E com um sistema de gestão 24/7, com o objetivo de proporcionar uma melhor utilização dos recursos, ele vai ampliar a vida útil de equipamentos, diminuindo o tempo de inatividade a partir de erros humanos e reduzir as despesas através da adoção de estratégias inteligentes para a utilização dos recursos;

Sustentabilidade: A estrutura tem a capacidade de adotar metas. Metas de SLA (99,999%), metas para o consumo de energia (kWh X média por dia) ou metas para a emissão de calor (média de Y BTU por dia). Além disso a estrutura pode reagir com os eventos do ambiente a fim de cumprir as metas pré-definidas. Eventos como o estado da sistema de energia - UPS (do inglês, *Uninterruptible Power Supply*) para baixo, sensores de temperatura acusando altos graus ou alarmes de incêndio. Em paralelo, se adapta ao ambiente dinâmico, a fim de cumprir as metas. Adaptações, como diminuição do sistema de arrefecimento, quando é interessante ativar a UPS, ou bloquear o acesso ao centro de dados, quando necessário.

2.8 SUMÁRIO DO CAPÍTULO

Os tópicos analisados aperfeiçoam-se a cada evolução das estruturas. Apresentando uma maior modularização das áreas e expansão das funcionalidades.

Na orquestração física a tendência é a separação da configuração dos servidores físicos, da configuração das MVs, responsáveis pelos serviços, permitindo crescimento horizontal da nuvem e uma livre configuração de suas MVs.

Para a orquestração de serviços, a reorganização dinâmica dos recursos dentro da nuvem mostrou-se imprescindível para a otimização da utilização dos recursos, melhorando o desempenho de suas MVs e serviços.

O balanceamento de carga mostra-se extremamente dependente da mobilidade da estrutura. Este, primordialmente preso pela estaticidade da estrutura física, mostra agora tendências de adaptação contextual, variando seu local de processamento de acordo com o fluxo inferido pelo ambiente.

A alta disponibilidade mostra-se uma funcionalidade em expansão vertical e horizontal. Partindo inicialmente do nível de serviços, utilizando redundância de serviços, esta extendeu-se ao nível das MVs, utilizando redundância e adaptações de MV, e ao nível das máquinas físicas, utilizando estratégias sustentáveis como "*wake-on-fail*", ativando máquinas em caso de falha do sistema.

O quesito CAPEX demonstra amadurecimento, trocando quantidade por qualidade. Quanto aos equipamentos, mostra tendências a utilização de menos equipamentos, porém mais robustos e especializados, como sistema de armazenamento, redes banda infinita e processadores gráficos. Quanto à utilização dos mesmos, começa a prezar mais pela saúde dos mesmos, almejando aumentar sua vida útil com uma estratégia parcimoniosa de uso.

Os OPEX visam o limiar da subsistência. Procurando cada vez mais utilizar o mínimo (recurso energético) e demandar o mínimo (de calor). Paralelamente prevê a automatização da nuvem, e uma interferência humana é cada vez menor, mesmo exigindo uma mão-de-obra mais capacitada.

As tendências de responsabilidade social vêm agregando valores às empresas que adotam soluções sustentáveis em seu modo de trabalhar.

Assim a questão de sustentabilidade vem, junto com as questões do OPEX e do CAPEX, formar um triângulo o qual exige equilíbrio da parte gerencial.

Apesar de geralmente a queda das OPEX e CAPEX ser reflexo da adoção de soluções sustentáveis (como "desligamento e hibernação"e estruturas minimalistas), outras vezes a adoção da mesma tem conseqüências inversas, visto que produtos ecológicos (os quais agridem menos a natureza) tendem a ser mais caros.

A tendência, então, é que o equilíbrio entre estes três pontos seja controlado pela nuvem de forma automatizada.

Como por exemplo, sistemas de controle de energia (*nobreaks*) inteligentes que recarregam-se em horários de baixo custo e provem sua energia em horários de alto custo (VYTELINGUM et al., 2010).

Na parte de controle do ambiente a interação entre a nuvem e seu ambiente foi o ponto mais fraco encontrado. Talvez pela dificuldade, ou falta de padrões, de intercomunicação entre os elementos não computacionais, com sistemas de refrigeração (não reativos), *nobreaks* e alarmes. E até mesmo de elementos subjugados, como equipamentos de interconexão de redes. Porém, incentivados por algumas pesquisas (GOOGLE, 2010), está atitude esta mudando. Tendendo para nuvens que residem em ambientes que mudam para melhor comportá-la, utilizando sistemas de refrigeração proativos (baseados na utilização dos serviços), sistemas de energia inteligentes e segurança de dados baseado no acesso físico da nuvem.

3 MODELO PROPOSTO

Neste capítulo descreve-se a proposta da pesquisa, buscando a junção entre alguns conceitos, na busca de uma melhor eficiência no controle do ambiente, são eles: a teoria da organização, a virtualização e o gerenciamento de recursos autonômico.

Considerando que pesquisas descrevem para servidores raramente completamente ociosos e raramente operando perto de sua máxima utilização, ou seja, os servidores operaram a maior parte do tempo entre 10 e 50 por cento de seu nível máximo de aproveitamento (BARROSO; HöLZLE, 2007).

A ideia da pesquisa é um modelo para alocação e distribuição de máquinas virtuais, para um ambiente de computação em nuvem verde, onde através da simulação, mostra-se que não somente através de migrações de MV, mas também através do redimensionamento de MV. Com isto, são alcançados ganhos significativos no consumo de energia, e em não violar SLA. Considerando assim, ambientes de computação em Nuvem Verde, para diferentes cargas de trabalhos (isto é, homogêneas/heterogêneas), principalmente atendendo os casos imprevisíveis de picos de trabalho, em diferentes aplicações, com diferentes SLA. Atendendo a um modelo dinâmico de alocação de recursos, para uma otimização autônoma.

A pesquisa busca uma solução de gerenciamento que controle todos os elementos do ambiente (por exemplo, Servidores, MV, Elementos de Rede, A/C - Ar Condicionado, entre outros).

A solução teria assim uma representação do ambiente (isto é, modelo do mundo), com sensores para atualizar essa representação do ambiente. Um sistema de deliberação sobre as possíveis ações sobre circunstâncias no modelo de ambiente no momento (isto é, regras de gerenciamento) e um mecanismo de implementação de ações de gerenciamento (isto é, automático ou manual).

Tanto sistemas de gerenciamento centralizados como distribuídos podem implementar diferentes graus de complexidade. Porém, sistemas centralizados são inadequados para gerenciar sistemas altamente distribuídos e/ou para ambientes voláteis devido a falta de flexibilidade e custo de reconfiguração das regras centralizadas associadas aos vários elementos no sistema, ou seja, qualquer alteração na arquitetura ou configuração do sistema implica em atualização tanto da representação do ambiente quanto base de regras normais, tornando inviável a operação de grandes sistemas e/ou sistemas que alteram muitas vezes num intervalo de tempo.

Sendo assim, nesses ambientes, a pesquisa considera ter um sistema multicomponentes de gerenciamento especializados para monitorar, repre-

sentar, deliberar e atuar em elementos específicos, promovendo flexibilidade e auto-configuração da arquitetura do gerenciamento que reflita as mudanças no ambiente de rede. Nesse caso, à medida que os elementos do sistema alterem os elementos de gerenciamento serão alocados/alterados para refletir a nova configuração.

Idealmente, espera-se que esses multicomponentes tenham um grau de liberdade de decisão e interação locais. Esta característica impulsiona os processos de adaptação a situações inesperadas e configurações não previstas. Sendo essas características desejáveis em sistemas de gerenciamento (isto é, controle) em ambientes altamente distribuídos e com grande volatilidade.

O problema de ter multicomponentes com alto grau de independência de atuação (isto é, adaptação local) é a controlabilidade do sistema como um todo. Ou seja, se cada componente pode tomar a decisão que quiser como ter certeza que o conjunto de decisões favorecerá a execução do sistema como um todo? A solução proposta é delimitar o grau de liberdade individual, impondo regras contratuais/sociais de comportamento que busquem garantir os objetivos/comportamento do sistema como um todo. Estruturas Organizacionais impõem as regras de funcionamento e padrões de interação entre os componentes, garantindo o comportamento global do sistema.

A solução busca promover duas características necessárias em Sistemas Adaptativos Complexos:

- o balanço entre adaptatividade e robustez;
- o balanço entre decisões/componentes locais e com grau de liberdade e controlabilidade.

O estudo busca um modelo integrado, ou seja, um modelo diferenciado em nível de serviços que atenda todos os requisitos da proposta acima, respondendo a pergunta acima. O sistema controlaria de uma maneira organizada e eficiente os recursos, tratando a entrada e saída de novos recursos, de novos dispositivos em um ambiente distribuído. Desta forma chegamos a um modelo baseado em estruturas organizacionais, utilizando o modelo de *Teoria da Organização*, ou seja, da mesma forma que uma empresa se estrutura, dividindo tarefas, delegando funções e atribuições a cada componente da estrutura.

3.1 MODELO BASEADO EM ORGANIZAÇÕES

Propõe-se um modelo de gerência proativa da computação em nuvem baseado na distribuição de responsabilidades por papéis (Figura 6). A responsabilidade da gerência dos elementos da nuvem é distribuída entre vários

agentes, cada qual em uma área de atuação. Estes controlarão individualmente os elementos da nuvem que lhe competem. Agindo de forma orquestrada visando o cumprimento de normas (metas). Tal orquestração baseando-se no fato (1) do conhecimento sobre o estado da nuvem (como um todo) ser compartilhado por todos e (2) ao fato de existirem crenças, constantemente revistas, sobre o funcionamento interno da mesma.

Na pesquisa é definido como proativo, o que toma atitudes para resolver problemas antes que eles aconteçam; alguém ou alguma situação que antecede problemas, mudanças ou necessidades futuras, que seja capaz de antecipar e modificar uma ocorrência de forma hábil e competente.

Dado que toda a estrutura do centro de dados é dimensionada e utilizada para prover serviços, esta continua sendo apenas uma ferramenta para disponibilizar tais serviços. Geralmente acordos (SLA) são estabelecidos visando esclarecer as responsabilidades de cada parte (cliente / provedor). Estes acordos devem manter-se ao seu nível (o de serviço), tornando-se puramente regras comportamentais (ou seja, retardo, tolerância a falhas, entre outras) do serviço, excluindo exigências estruturais e físicas. Com a saída da configuração do ambiente de dentro do acordo, a nuvem pode tornar-se flexível.

Com a liberdade de alteração de configuração da estrutura, a mesma pode se tornar dinâmica e extensível, permitindo abranger fatores externos aos acordos estabelecidos, porém ainda assim vitais (por exemplo, como consumo de energia, desgaste de *hardware*, entre outros).

Assim como as leis da física, a arquitetura da computação em nuvem também deve existir sob normas bem definidas. Estas normas expressam (1) as regras de comportamentos estabelecidos no SLA e (2) os objetivos (de interesse) internos da nuvem que devem ser levados em conta.

Para que os vários elementos da nuvem trabalhem de forma eficiente, buscando a efetivação destas normas, eles devem ser coordenados por agentes externos aos serviços por eles auditados, gerenciando, por exemplo:

- Ativação e Desativação de Máquinas Virtuais;
- Ativação e Desativação de Máquinas Físicas;
- Alteração na configuração de Máquinas Virtuais;
- Alteração no Funcionamento de Equipamentos do Centro de Dados;
- Alteração no Funcionamento de Ativos de Rede;
- Agregação e Separação de Nós de *Clusters*.

Visto a vasta gama de elementos a serem gerenciados a complexidade cresceria proporcionalmente com o tamanho da arquitetura do ambiente de computação em nuvem.

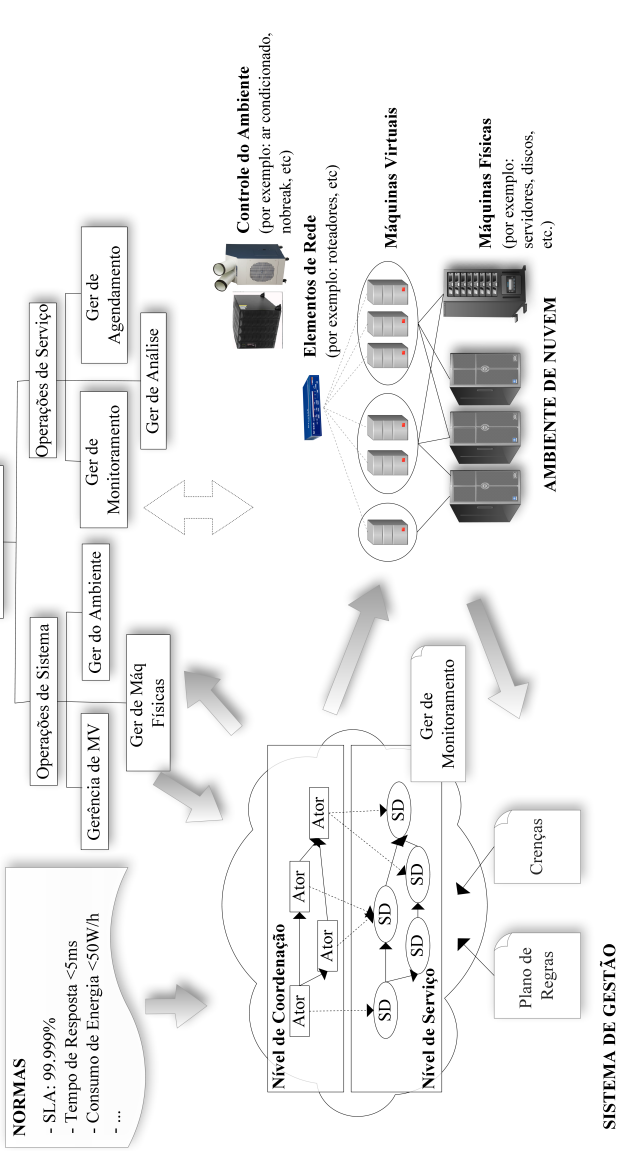

Figura 6 – Modelo de Gerenciamento.

3.1.1 Papéis

Para evitar tal complexidade infere-se uma hierarquia aos agentes-gerentes. Podendo fazer uma analogia a uma empresa de grande porte, onde há uma hierarquia a ser seguida e responsabilidades sendo delegadas. Onde se tem a Organização dono da empresa, visualizado na Figura 7, dividindo responsabilidades em Diretores e cada diretoria tem seus Gerentes que cuidam de áreas específicas, ou seja, processos por especialidades, de maneira a melhor controlar e administrar a organização.

Desta forma, deve existir um Gerente do Sistema, ver Figura 8, que manda em todo o ambiente, e seguindo a hierarquia temos os Coordenadores que dividem as operações entre suas equipes (DIGNUM et al., 2009), de maneira a facilitar a divisão de tarefas e responsabilidades entre suas equipes.

Baseando-se nas informações cedidas por seus agentes-monitores este deve tomar decisões em prol das normas a serem seguidas.

Dependendo da situação, as decisões gerarão Operações de Sistema ou Operações de Serviço, ou ambas. As Operações de Sistema podem ser divididas em Gerência de Máquinas Virtuais, Gerência de Máquinas Físicas e Gerência do Ambiente. As Operações de Serviço podem ser divididas em Gerência de Monitoração, Gerência de Agendamento e Gerência de Análise.

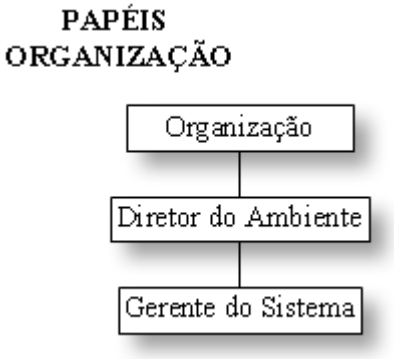

Figura 7 – Modelo de Organização.

A ação de cada papel reflete diretamente na configuração da estrutura como um todo.

As Operações de Sistema agirão na estrutura a qual os serviços estão sendo processados e no ambiente que esta estrutura se localiza. As Operações de Serviço agirão na camada dos serviços, adquirindo informações sobre ou

alterando a configuração a qual os serviços estão sendo executados.

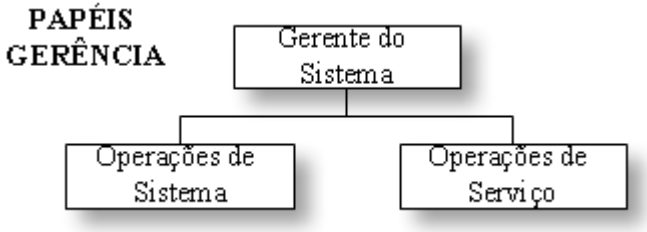

Figura 8 – Modelo de Gerência.

Os três papéis em que as Operações de Sistema podem ser classificadas são:

- Gerência de Máquinas Virtuais: responsáveis pelas ações inferidas nas máquinas virtuais. Tem papel de interface das máquinas virtuais ao modelo. Como por exemplo, criar ou destruir uma máquina virtual, alterar suas configurações e até movê-la de uma máquina física à outra (seja entre *clusters* locais ou remotos);

- Gerência de Máquinas Físicas: responsáveis pelas ações inferidas nas máquinas físicas. Tem papel de interface das máquinas físicas ao modelo. Como por exemplo, desligar e ligar as máquinas físicas, alterar configurações do sistema operacional hospedeiro (por exemplo, BIOS (do inglês, *Basic Input/Output System*) e SMART (do inglês, *Self-Monitoring, Analysis, and Reporting Technology*)) e configurações de *hardware* (por exemplo, Cooler e Acelerômetro), assim como de equipamentos finais (por exemplo, *Storages, Switches e Site Backups*);

- Gerência do Ambiente: responsável pelas ações fora da estrutura. Tem papel de interface do ambiente ao modelo. Como por exemplo, controle de temperatura do centro de dados, controle sob os sistemas de *backup* de energia (por exemplo, *Nobreaks* e Geradores), controle sobre a acessibilidade do centro de dados.

Os três papéis das Operações de Serviço podem ser classificadas são:

- Gerência de Monitoração: responsável pela coleta de informações da estrutura, de modo geral, e sua compreensão. Tem papel de manter o modelo ciente do estado da nuvem. Monitora os servidores, máquinas virtuais, tráfego de rede e etc. Baseado em parâmetros específicos

previamente configurados pelo Gerente do Sistema, tal como (1) a utilização de um recurso e seu limiar (*threashold*) de notificação, (2) a disponibilidade da banda de rede (dado binário) ou (3) a disponibilidade (*idleness*) de algum elemento da estrutura;

- Gerência de Agendamento: responsável pela agenda da nuvem. Tem um papel proativo dentro do modelo, planejando as ações a serem tomadas diante dos eventos agendados. Como por exemplo, manutenções planejadas geram o protocolo a ser seguido para a efetivação do mesmo. Em uma troca de máquinas físicas, por exemplo, gerará uma lista de passos que devem ser seguidos, prevendo: Teste de *Nobreak* Secundário, Ativação de Servidor Secundário, Teste dos *Hardwares* do Servidor Secundário, Testes de Vazão de Rede entre os servidores, Testes de Acesso ao Armazenamento, Migração das Máquinas Virtuais, Desligamento do Servidor Primário, Desativação de Alarmes do Centro de Dados;

- Gerência de Análise: responsável pelos testes comportamentais dos serviços e sua análise. Tem o papel de auditoria do serviço prestado pela estrutura e a compreensão do mesmo. Certifica-se se o serviço prestado está de acordo com as normas a serem seguidas, inferindo limiares pré-estabelecidos e alertando o Gerente do Sistema. Monitora a Qualidade de Serviço que é processado, e cruza com as variações da estrutura, procurando padrões entre o desempenho obtido e os elementos.

3.1.2 Regras de Planejamento

As Regras de Planejamento são as bases de conhecimento teórico que relacionam contextos com os seus objetivos. São utilizadas em momentos que decisões devem ser tomadas, durante o planejamento das ações. São conhecimentos primitivos inferidos pela experiência dos administradores. Pode-se tomar como exemplo de regras de planejamento as seguintes noções:

- Se MV aumenta utilização de *Swap*. Para diminuir a utilização de *Swap*, aumentar memória *RAM*;

- Se Máquina Física apresenta alta carga. Para diminuir a carga, mova a MV com maior processamento para outra MF - Máquinas Físicas;

- Se um centro de dados apresenta alta carga. Para diminuir a carga geral, ligue mais Máquina Física.

3.1.3 Crenças

Para que estes perfis possam tomar as melhores decisões, prevê-se a necessidade de "bases de conhecimento", a qual está sendo chamada de Crenças, conjunto de conhecimentos empíricos, utilizados para aprimorar as decisões a serem tomadas. Tem-se assim a compreensão do funcionamento da Nuvem, através das bases de conhecimento, são relacionadas as variáveis dentro da estrutura, como resultado temos algumas ações prévias. As crenças expressam a junção do conhecimento teórico, das premissas vindo das normas e o conhecimento empírico, originado dos históricos e experiências passadas. As crenças devem ser revistas freqüentemente por todos os elementos do modelo, assim como o compartilhamento destas revisões. Pode-se tomar como exemplo de crença as seguintes noções:

- A ativação de um servidor tipo X representa o aumento de Y graus em Z minutos;
- A ativação de uma MV tipo A representa um aumento de consumo de B kWh;
- Uma MV tipo A suporta C requisições por segundo, e;
- A realocação de uma MV tipo A representa uma diminuição de B kWh.

3.2 MODELO DE ALOCAÇÃO E DISTRIBUIÇÃO DE MVS

Neste contexto, faz-se o mapeamento das Operações do Sistema, onde o Gerente de Máquinas Virtuais deve controlar o processo de alocar recursos dinamicamente no ambiente. O modelo faz um paralelo com a estrutura organizacional estruturando a infraestrutura de um ambiente de modo que cada componente pode ser visto como um serviço, visualizado na Figura 9.

O problema está na capacidade das Nuvens em oferecerem serviços para o usuário. A alocação de recursos é baseada em análise de tendências (análise de dados históricos) e não análise causal (isto é, que tipo de evento pode causar variações na entrada). Assim, o problema interno está na maleabilidade do sistema em poder se ajustar a situações inesperadas (por exemplo, picos de tráfego inesperados). A pergunta a ser respondida neste ponto é: Quais são os critérios de reconfiguração interna do sistema (visando exclusivamente reconfiguração da distribuição de MV, no caso) para garantir a melhor performance do sistema em relação a qualidade de serviços e considerando a disponibilidade de recursos?

Se houvessem recursos infinitos, então alocar-se-iam MVs de tamanho gigantescos. Mas isso não é realístico, pois consumiria-se muitos recursos, tendo um aumento nos gastos, provavelmente desproporcional ao produto consumido.

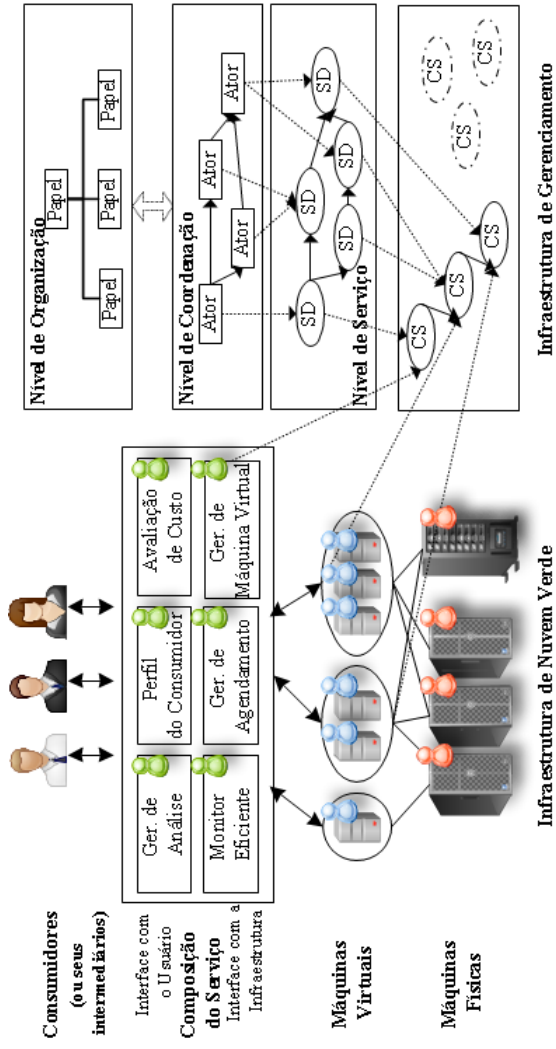

Figura 9 – Modelo de Alocação Baseado em Serviços.

Se houvesse apenas um serviço rodando no mesmo domínio de recursos, todo o recurso seria destinado aquele serviço. Mas isso não é computação em nuvem, tem-se dessa forma um ambiente de arquitetura convencional, ou seja, para cada servidor um serviço, o que não é viável, é ultrapassado nos moldes da computação distribuída atualmente.

Então, havendo limitação de recursos e sendo o princípio da computação em nuvem o compartilhamento de *hardware* entre múltiplos serviços (por exemplo, MV é uma ferramenta para isso), as sub-questões são:

- Como distribuir MV para os serviços no tempo (isto é, dado uma previsão de necessidade de carga)?

- Como ajustar o sistema caso determinados serviços extrapolem essa previsão (isto é, picos de carga de alguns serviços)?

- O que fazer nos casos extremos? (por exemplo, picos de cargas de mais de um serviço, picos de todos os serviços, etc).

Entende-se assim que a idéia para resolver as questões deva passar por uma estratégia para alocação dinâmica de máquinas virtuais em ambientes de computação em nuvem, a qual se baseia inicialmente na migração da carga de trabalho de um servidor físico para outro conforme a demanda por capacidade de processamento varia, avaliando o cenário de alocação das MV ao término (isto é, *on line*) da execução de cada tarefa, com o intuito de identificar um hospedeiro ocioso ou subutilizado. Caso o hospedeiro ocioso ou subutilizado seja identificado, a MV é então migrada.

Uma segunda alternativa seria ajustar os parâmetros da MV (WOOD et al., 2009) sem precisar migrar ou desligar, por exemplo, aumentando ou diminuindo os parâmetros de alocação de CPU e memória, caso exista a possibilidade no ambiente, ocorrendo, assim, um balanceamento na carga de trabalho, de acordo com a demanda por processamento naquele instante.

Algumas perguntas seriam respondidas: (1) Para que migrar em si? Porque não acrescentar mais uma MV *on demand*, simplesmente? (2) Qual o impacto da migração na operacionalização do sistema? Por exemplo, cria algum atraso no processamento dos serviços durante a migração (isto é, que seria inaceitável)? Abaixo é apresentado um exemplo de problema:

- Utilizando alguns parâmetros hipotéticos, dados de consumo, como se fosse de *benchmarks*, SPEC (do inglês, *Standard Performance Analysis Corporation*)(SPEC, 2010), para a comparação;

- Considerando que a demanda para processar uma tarefa J de um serviço SJ for (10 SPEC, 300Kb);

- Considerando ainda que se tem uma média de 500 vezes a tarefa J num tempo T1 e 1000 vezes a tarefa J num tempo T2, em T3 o sistema baixa para 500 vezes a tarefa J.

Então em um tempo T1 e T3 é necessário uma MV configurada para (5.000 SPEC, 150Mb) em um tempo T2 essa MV precisa ter (10.000 SPEC e 300Mb), desta maneira chega-se a três (3) possíveis soluções:

1. Solução (Redimensionamento de Máquinas Virtuais): no tempo T1 aloca-se uma MV de (5.000 SPEC, 150Mb), perto de T2 reconfigura-se para (10.000 SPEC e 300Mb) e depois em T3 à configuração volta para (5.000 SPEC, 150Mb), ver Tabela 2;

2. Solução (Migração de Máquinas Virtuais): num tempo T1 aloca-se uma MV1 com (5.000 SPEC, 150Mb) numa máquina MF1, perto de T2 aloca-se uma máquina MV2 com (10.000 SPEC e 300Mb) em uma máquina MF2 e é removido MV1, depois em um tempo T3 aloca-se uma MV1 com (5.000 SPEC, 150Mb) novamente numa máquina MF1, ver Tabela 3;

3. Solução (Sem Migração e Sem Redimensionamento): em um tempo T1 aloca-se MV1 com (5.000 SPEC, 150Mb) numa máquina MF1, perto de T2 aloca-se mais uma MV2 com (5.000 SPEC e 150Mb) em uma máquina MF2 e não é removido a MV1, reconfigurando o controlador para atuar entre as máquinas MF1 e MF2), depois de um tempo T3, é removida a MV2, ver Tabela 4.

Tempo	MF1 / MV1
T1	5.000 SPEC, 150Mb
T2	10.000 SPEC, 300Mb
T3	5.000 SPEC, 150Mb

Tabela 2 – Cenário Realocação

A idéia inicial sim é ajustar os parâmetros da MV em tempo real, *on line*, das tarefas, aplicações, por exemplo, quando o processamento estiver em 90% o sistema verifica as métricas de negócio da empresa (contratante/-contratada), tendo que garantir a qualidade, tempo de resposta, sem parar o sistema e com baixo custo, numa aplicação que cresça sua necessidade de processamento, o sistema aumentaria a MV, para 110% (isto é, realocação).

Num caso onde, por exemplo, quando o processamento estiver em 90% o sistema verifica as métricas de negócio da empresa (isto é, contratante/contratada), tendo que garantir a qualidade e tempo de resposta, e se o

Tempo	MF1 / MV1	MF2 / MV2
T1	5.000 SPEC, 150Mb	-
T2	-	10.000 SPEC, 300Mb
T3	5.000 SPEC, 150Mb	-

Tabela 3 – Cenário Migração

sistema fica ocioso num determinado tempo (isto é, tem um intervalo entre uma tarefa e outra) e tem baixo custo, numa aplicação que cresça sua necessidade de processamento, a aplicação seria migrada para outra MV, onde o processamento fica mais ajustável.

Tempo	MF1 / MV1	MF2 / MV2
T1	5.000 SPEC, 150Mb	-
T2	5.000 SPEC, 150Mb	5.000 SPEC, 150Mb
T3	5.000 SPEC, 150Mb	-

Tabela 4 – Cenário Sem Realocação e Sem Migração

Podem existir casos onde o sistema analisa a concentração de diferentes máquinas virtuais em um número menor de máquinas físicas para adequação da eficiência, porém antes de ajustar ou efetuar a migração, o sistema teria que observar as métricas e critérios contratados para não parar a aplicação, violando as regras de negócio. Defini-se abaixo alguns requerimentos, da estrutura:

- O processo tem que ser *on line*;
- Tem que vislumbrar métodos de proatividade; ou seja, considerar as métricas históricas e sugerir/realizar as alterações considerando para um período para frente de tempo, sem ter as informações completas e (então) considerando um período para trás de tempo;
- Tem que visualizar mecanismos de segurança e margem de erro para os casos de o tráfego ser muito acima do que o esperado e se houver um pico de tráfego inesperado num momento, etc.

O problema de alocação e distribuição de máquinas virtuais pode ser dividido em duas partes:

1. A admissão de novos pedidos com a alocação e distribuição de MV em diferentes máquinas físicas;

2. A adequação ou otimização das MV existentes nas máquinas físicas existentes.

A seguir a pesquisa analisa os diferentes tipos de cargas de trabalho, os diferentes acordos de nível de serviço, para adequar as duas partes citadas de acordo com os requerimentos estabelecidos, propondo assim a abordagem de migração e realocação de acordo com o modelo de Teoria da Organização.

3.2.1 Cargas de Trabalho

A cargas de trabalho podem variar de acordo com a necessidade de processamento, armazenamento, memória e até mesmo banda de rede, considerando assim a possibilidade de atrasos ou não na resposta do ambiente, picos de demanda, serviço previsíveis ou imprevisíveis, horários em que demandam mais ou menos recursos.

Diante destas variações classifica-se as cargas de trabalho, conforme mostrado no trabalho de (BELOGLAZOV et al., 2011), adaptado na Figura 10, uma taxomia de cargas de trabalho computacional como aplicações Arbitrárias, aplicações em Tempo Real e Aplicações de Alta Performance - HPC (do inglês, *High-performance computing*).

Aplicações Arbitrárias podem ser definidas como serviços que não tem uma particularidade, ou seja, não possuem uma característica bem definida, não demandam recursos de maneira constante. Alguns exemplos seriam Serviços *Web*, Serviços de *Backup* ou Serviço de *Boot* Remoto. Geralmente Serviços *Web* possuem uma utilização constante durante o dia, apresentando picos esporádicos, porém previsíveis (por exemplo, horário de almoço, pós-expediente, entre outros).

Serviços de *Backup* já são processados fora dos horários de trabalho, durante a noite e demandam muito recurso de E/S (Entrada e Saída), porém podem ser paralelizados.

Enquanto Serviço de *Boot* Remoto demonstra picos no início do expediente, e a utilização constante durante o dia, porém geralmente não é paralelizado.

Aplicações em Tempo Real podem ser definidas como serviços que demandam recursos de maneira constante, ou seja, enquanto ativas não podem sofrer perdas de conexão entre o cliente e o servidor. Alguns exemplos poderiam ser Serviços de Telefonia da *Internet*, Serviço de Tele-conferência pela *Internet*, Sistemas de Supervisão ou até mesmo Sistemas de Monitoramento de Pacientes em Hospitais.

Um sistema de tempo real é dito ser previsível, no domínio lógico e no domínio temporal quando, independentemente de variações ocorrendo em

nível de *hardware*, da carga e de falhas, o comportamento do sistema pode ser antecipado, antes de sua execução (FARINES; FRAGA; OLIVEIRA, 2000).

Aplicações de Alta Performance - HPC podem ser definidas como aplicações ou serviços, que focam em performance (isto é, velocidade). Como exemplo, temos aplicações que analisam grandes massas de dados e exige um grande processamento, um grande desempenho, geralmente utilizadas em pesquisas como no projeto Genoma.

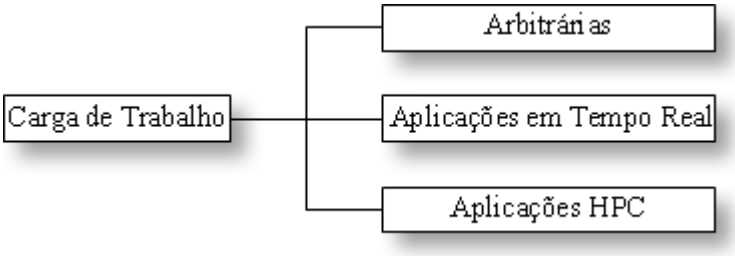

Figura 10 – Taxomia de Cargas de Trabalho adaptado de (BELOGLAZOV et al., 2011).

3.2.2 SLA - *Service Level Agreement*

O Acordo de Nível de Serviço - SLA (do inglês, *Service Level Agreement*), é um acordo formal entre a parte contratante e o contratado, determinando a qualidade mínima de serviço exigida. No SLA negociado entre duas partes, é designada a qualidade, as prioridades e as responsabilidades no serviço prestado. Na Figura 11, apresenta-se algumas métricas que são utilizadas na negociação entre as partes envolvidas, são elas:

- Segurança: garantindo controles de acesso, controle contra invasão, roubo de dados, entre outros;
- Disponibilidade: visando garantir a disponibilidade do sistema;
- Atrados: mantendo o tempo de atrasos de acessos, a taxa de perda de pacotes;
- Taxa de Transferência: garantindo um tempo mínimo entra a troca de requisições;
- Perda de Pacotes: definindo qual limite de perda de pacotes na rede.

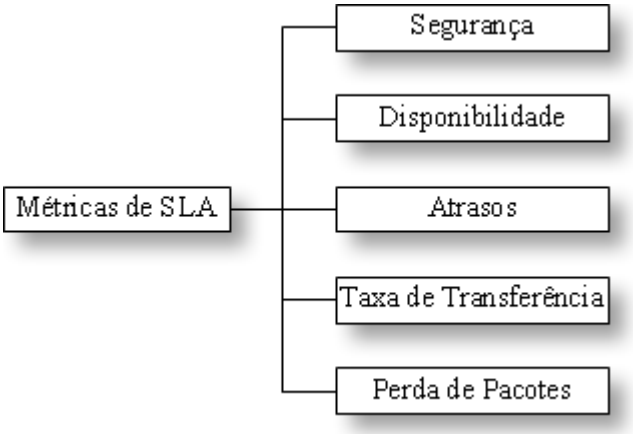

Figura 11 – Taxomia do SLA.

3.2.3 Provisionamento de Máquinas Virtuais

A estratégia para alocação e distribuição de máquinas virtuais deve inicialmente, conforme exposto na seção 3.2 trata da admissão de novos pedidos com o provisionamento das máquinas virtuais em máquinas físicas de acordo com os níveis de suas aplicações, assim como os níveis de seus SLAs.

Seguindo as funções hierárquicas do ambiente o conjunto de Operações de Serviço, através da monitoração e análise do ambiente, define-se as operações de provisionamento de recursos de maneira autonômica.

Com base nas funções determina-se que as regras de planejamento devam atribuir que as máquinas virtuais com maior prioridade devem ser submetidas seguindo as aplicações com maior prioridade, ou seja, Aplicações em Tempo Real e Aplicações HPC, para as máquinas físicas que possuam melhor qualidade, sejam mais eficientes e contenham o maior número de recursos disponíveis, evitando qualquer quebra de contrato, seguindo um limiar de de duas (2) MV de alta prioridade para uma (1) MF.

Já para as Aplicações Arbitrárias sem qualquer prioridade devem ser distribuídas nas máquinas físicas ocupadas, ou melhor, subutilizadas seguindo um limiar de cinco (5) MV para uma (1) MF.

3.2.4 Migração de Máquinas Virtuais

Para a adequação ou otimização das MVs ativas nas máquinas físicas existentes, ou seja, uma otimização da carga de trabalho, conforme tratado na Seção 3.2, é imprescindível uma estratégia eficaz de alocação de MVs. A estratégia deve inferir um plano de regras, com a monitoração do elementos e suas crenças, que seja capaz de migrar a carga de trabalho de acordo com a demanda em ambientes de computação em nuvem verde. O que é imprescindível para garantir que os serviços sejam capazes de atender as expectativas de QoS dos usuários, assim como garantir as estratégias de eficiência energética.

Percebe-se, desta forma, a necessidade de se adequar a capacidade de processamento a um ambiente inerentemente dinâmico. Nesse contexto, a estratégia para alocação dinâmica de máquinas virtuais em ambientes de Computação em Nuvem Verde deve tratar as infraestruturas do centro de dados a nível de *clusters* e avaliar o cenário de alocação das MVs ao término da execução de cada tarefa, com o intuito de identificar uma máquina física ociosa ou subutilizada.

Caso a Gerência de Monitoração, responsável pela coleta das informações da estrutura, ou a Gerência de Análise, responsável pelos testes comportamentais dos serviços e sua análise, identifiquem as máquinas físicas ou máquinas virtuais ociosas ou subutilizadas, os agentes avisam o Gerente do Sistema que atua junto a Gerência de Máquina Virtual, para migrar a MV. Em conjunto a Gerência de Máquinas Físicas e a Gerência do Ambiente atuem sobre a máquinas físicas e todos os equipamentos relacionados com aquele *cluster*, desligando os equipamentos obsoletos, ocorrendo assim um balanceamento na carga de trabalho, de acordo com a demanda por processamento naquele instante, possibilitando a economia de recursos assim como o uso efetivo dos recursos.

A ideia é que os agentes monitorem, analisem e atuem sobre os limites de utilização de processamento (CPU) e memória das máquinas físicas de tal forma que exista um limite superior e um limite inferior mínimos a serem seguidos em cada aplicação, preservando os recursos livres para impedir a violação de SLA. Sempre considerando concentrar máquinas virtuais no mínimo de máquinas físicas necessárias, porém considerando também migrar o menor número de MVs a fim de minimizar a sobrecarga de migração. Ou seja, se a utilização de processamento e memória de uma MF estiver abaixo do limite inferior, todas as MVs tem que ser migradas a partir desta MF e a MF tem de ser desligada, a fim de eliminar o consumo de energia ociosa. Se a utilização for acima do limite superior, algumas MVs tem que ser migradas para outra MF para reduzir o uso e evitar possível violação de SLA.

O pseudocódigo apresentado no Algoritmo 1 é uma simplificação do algoritmo utilizado na estratégia de migração de MV. O trecho de código apresenta a monitoração das máquinas físicas a nível de *cluster*, de forma a identificar o processamento ativo e faltante de cada tarefa, identificando as máquinas virtuais, as máquinas físicas ou *clusters* ativos, ociosos, subutilizados.

Algoritmo 1: Alocação e Distribuição.

```
para cluster faça
    retornaListaHosts();
    //monitora Host
    para host faça
        retornaListaVms();
        //monitora VM
        para vm faça
            retornaListaTasks();
            //monitora Workload ou Task
            para task faça
                tempoRestanteVm[k] = tempoRestanteVm[k] + +;
                //verificar o quanto de processamento a máquina
                virtual precisa.
            fim
            tempoRestanteHost[n] = tempoRestanteHost[n] +
            tempoRestanteVm[k];
            //verificar o quanto de livre/ocupado o host possui.
        fim
        tempoRestanteCluster[n] = tempoRestanteCluster[n] +
        tempoRestanteHost[k];
        //verificar o quanto de livre/ocupado o cluster possui.
    fim
fim
```

No Algoritmo 2, é mostrada a continuação do pseudocódigo onde efetivamente são checadas as condições para migração, realocação e ativação ou desativação de máquinas físicas. Conferidas as métricas e condições pré-estabelecidas, o Gerente de Análise pode decidir sobre a migração ou realocação, e atuar sobre os demais serviços para a efetiva alocação de recursos.

Apresenta-se na Figura 12, a proposta de migração, onde são visualizados dois *clusters* dentro de uma Nuvem, no primeiro *clusters* temos uma máquina física desligada e outras duas ligadas, porém no segundo *cluster*

tem-se somente uma máquina física sendo ocupada com poucas máquinas virtuais, desta forma essas máquinas virtuais são movidas para o outro *cluster*, ativando a máquina física desligada, e conseqüentemente desliga-se todo o segundo *cluster* e sua infraestrutura conectada, buscando dessa forma um modelo minimalista e eficiente.

3.2.5 Realocação de Máquinas Virtuais

Nessa abordagem, a proposta visa auxiliar o modelo em casos de variações rápidas e imprevisíveis de carga de trabalho, diminuir o número de violações de SLA e conseqüentemente reduzir o número de migrações de máquinas virtuais, aproveitando a disponibilidade de recursos dentro do domínio de processamento.

O sistema monitor de máquinas virtuais possibilita escalonar os seus limites de processamento. A técnica é desenvolvida em algumas pesquisas (SCHEDULER, 2010), para facilitar a implementação de ambientes virtualizados em grandes centros de dados, utilizando o monitor de máquina virtual de código aberto Xen.

Algoritmo 2: Migração e Realocação.

```
para cluster faça
    calculaDifTempoHosts();
    retornaListaVms();
    para host faça
        se checaCondições então
            ativaHost();
        senão
            desativaHost();
        fim
    fim
    para vm faça
        se checaCondições então
            migraVm();
        senão
            realocaVm();
        fim
    fim
fim
```

Baseado em que os monitores de máquinas virtuais, denominados *hypervisors*, como o Xen (HYPERVISOR, 2010), podem reservar recursos das máquinas físicas de maneira inadequada, deixando por vezes as máquinas virtuais ociosas ou subutilizadas, existe assim a possibilidade de realocação e distribuição das máquinas virtuais mais adequada, principalmente em casos de carga de trabalho dinâmicas.

O *hypervisor* do Xen utiliza como escalonador o *Credit*, que em seu modo padrão é configurado com pesos fixos (isto é, com limites máximos) de processador e memória para cada máquina virtual, porém ele pode ser configurado sem os limites máximos de processamento. Configurado dessa forma, sem limites máximos de processamento, o escalonador se ajusta dinamicamente às cargas das máquinas virtuais, sendo direcionada ao domínio principal todo restante de processamento. Nessa situação deve-se ter cuidado com a definição do acordo de nível de serviço entre os contratantes (HYPERVISOR, 2010).

Considerando que o tempo de provisionamento / desprovisionamento, para a migração de máquinas virtuais de uma máquina física para outra, para algumas cargas de trabalho, como por exemplo, em aplicação de tempo real ou de HPC, possam violar os acordos de SLA, é implementado dentro da

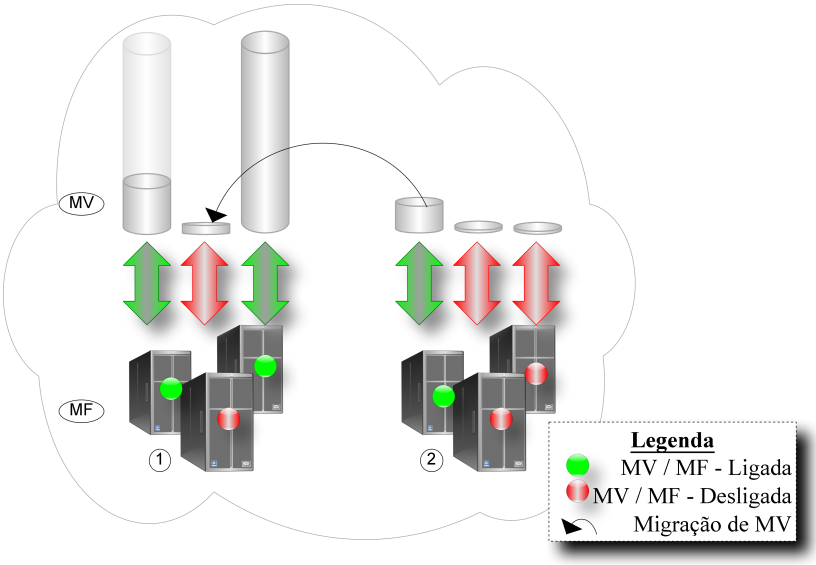

Figura 12 – Migração de Máquinas Virtuais.

proposta uma forma de realocação de máquinas virtuais.

A proposta de redimensionamento de máquinas virtuais deve seguir algumas condições, ou seja, analisar o impacto de diferentes cargas de trabalho e de diferentes SLAs, definindo quando migrar. Considera-se a disponibilidade das máquinas físicas em ter mais recursos, em alguns casos pode ser interessante o ajuste da máquina virtual, para a economia de recursos e uma menor violação de SLA. Em geral aplicações de HPC ou aplicações que demandem disponibilidade total do serviço, tem uma maior prioridade no ambiente, desta forma a migração de máquinas virtuais para um outro servidor, ocasiona a perda de um certo tempo, gerando violações de SLA, ou seja, multas contratuais.

A proposta desta seção visa fazer a realocação dos recursos para as máquinas virtuais, no caso haver picos imprevistos de carga de trabalho e existirem máquinas virtuais não utilizando todos os recursos disponibilizados a ela. Através da Gerência do Ambiente, o ambiente verifica qual máquina virtual precisa ultrapassar seus limites máximos de processamento, assim como quais máquinas virtuais estão obsoletas. O Gerente de Máquina Virtual irá modificar as configurações dos pesos de utilização de recursos (por exemplo memória e processador) somente se as outras máquinas virtuais não estiverem utilizando parte dos recursos que elas possuem disponíveis, proporcionando um melhor balanceamento de carga no ambiente.

Na Figura 13 é visualizada a proposta de realocação de máquinas virtuais. No primeiro *cluster*, num dado instante a primeira máquina virtual sofre um pico de carga de trabalho, necessitando mais recursos do que possui disponível, enquanto que a segunda máquina virtual, do mesmo *cluster* não está utilizando todos os recursos que lhe foram disponibilizados. O agente através da sua função proativa irá redistribuir a alocação de recursos, escalonando de forma diferenciada para atender a demanda do momento imprevisto.

Como verificado, de maneira autonômica, a proposta é que todas as máquinas tenham os recursos que necessitem nos momentos imprevisíveis (isto é, não perdendo muito tempo, evitando a violação de SLA e economizando energia) e que não existam recursos subutilizados.

Na situação de pico de carga de trabalho e não havendo recursos disponíveis naquelas máquinas físicas, o sistema consideraria as máquinas virtuais de maior SLA, ou melhor de maior prioridade e migraria para outra máquina física disponível com mais recursos disponíveis, de modo a ocorrer uma menor possível violação de SLA.

3.2.6 Monitoração de Performance

O Sistema de Gerência "imita"as operações de recursos físicos na Estrutura da Organização, classificando e antecipando os comportamentos dos elementos, planejando adaptações reativas e proativas dos parâmetros de operação. Ele ainda combina informações de Normas, Estrutura da Organização e Crenças nas estatísticas atuais do Ambiente de Nuvem, com as atuais estimativas de desempenho da organização e caso seja abaixo dos níveis aceitáveis, delibera sobre os ajustes de parâmetros de operação. Quando necessário, faz a reorganização e/ou realocação de elementos.

Com o intuito de gerenciar os elementos do ambiente da Nuvem Verde (*Green Cloud*), conforme esboçado na Figura 14, cada elemento do ambiente deve ser tratado como um serviço, garantindo um melhor gerenciamento, um gerenciamento distribuído e proativo, onde cada elemento é responsável por realizar a coleta das informações do equipamentos e aplicações.

Ainda na Figura 14 podemos visualizar quatro pontos importantes da infraestrutura de computação em Nuvem Verde, os consumidores (e/ou seus intermediários), a composição do serviço, as máquinas virtuais e finalmente

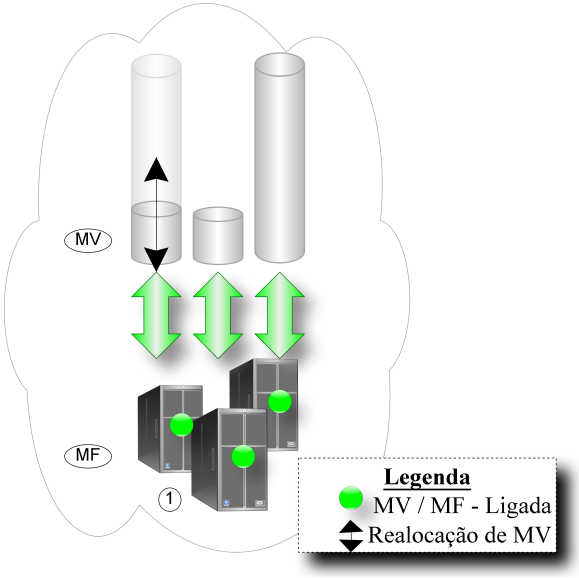

Figura 13 – Realocação de Máquinas Virtuais.

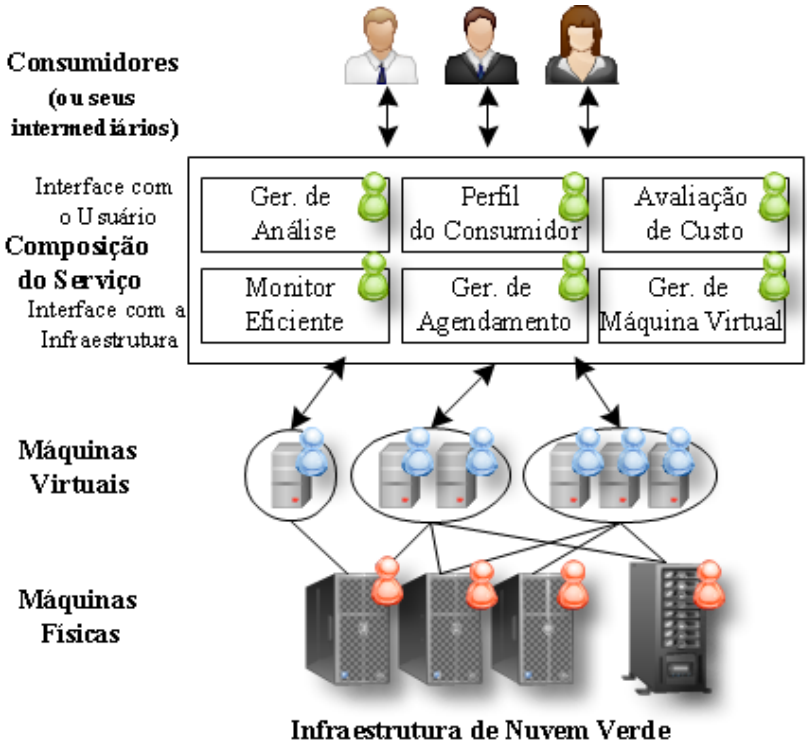

Figura 14 – Gerência da Infraestrutura.

as máquinas físicas. Consideremos que cada elemento dentro dessa infraestrutura possui agentes-gerentes que monitoram a entrada e a saída de qualquer componente do sistema. A composição do serviço é dividida em serviços de interação (isto é, monitoram, configuram, atuam) com o usuário (interface com o usuário) e com a infraestrutura (interface com a infraestrutura), onde cada serviço possui um papel ou função dentro da infraestrutura.

Como elementos essenciais na monitoração de performance, a interface do usuário possui:

- Gerência de Análise: conforme descrito na Seção 3.1.1, faz uma auditoria do serviço prestado pela estrutura, de maneira a manter a qualidade do serviço, cruzando dados de elementos e do desempenho;
- Perfil Consumidor: reúne as características específicas de consumido-

res, de modo que os consumidores importantes podem ser concedidos privilégios especiais e com prioridade sobre os demais consumidores, ou seja, quem tem um maior nível de SLA, tem preferência;

- Avaliação do Custo: decide como as solicitações de serviços são cobradas para gerir a oferta e a demanda de recursos computacionais e facilitar a atribuição de prioridade de serviços de forma eficaz.

Na interface da infraestrutura temos:

- Monitoração Eficiente: observa e determina quais máquinas físicas podem ser ligadas ou desligadas de acordos com as políticas de eficiência energética;
- Gerência de Agendamento: conforme descrito na seção 3.1.1, atuando de forma proativa, planejando as atividades a serem realizadas no ambiente;
- Gerência de Monitoramento: conforme descrito na seção 3.1.1, basicamente coleta os dados do estado da estrutura da nuvem.

Todos estes elementos continuamente atualizam as Crenças, refletindo as condições e comportamentos do ambiente de Nuvem.

3.2.7 Equilíbrio entre a Distribuição e Alocação

Baseado no provisionamento dos diferentes tipos de carga de trabalho no centro de dados o sistema deve de maneira proativa avaliar de acordo com suas crenças, a entrada de novas requisições e destinar, agrupar cargas de menos impacto em um grupo de servidores de mesmo *cluster*.

Potencialmente o sistema deve analisar suas métricas de QoS, dividindo as cargas com menor ou maior SLA, de acordo com as regras de planejamento do sistema.

Não obstante, verificando ao final de cada tarefa o sistema deve avaliar o plano de regras, imposto ao centro de dados, para ajustar a alocação e distribuição de carga em tempo real, diminuindo a quantidade de servidores ou até mesmo *clusters*, ociosos ou subutilizados, movendo ou realocando as máquinas virtuais, objetivando concentrar as MVs em um menor número de máquinas físicas.

Por exemplo, conforme visualizado na Figura 15, pode-se considerar a mitigação ao nível de consumo de energia e sustentabilidade do ambiente, em relação as políticas ou abordagens de migração e realocação,

descritas nas Seções 3.2.4 e 3.2.5, ou seja, teria-se um crescimento do nível de SLA, proporcional ao número de abordagens implementadas para garantir o nível do serviço, conseqüentemente um menor consumo de energia.

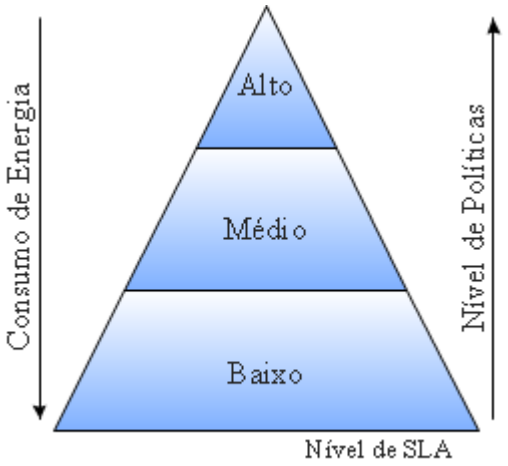

Figura 15 – Comparativo entre Nível de SLA, Consumo de Energia e Políticas.

Deve-se ainda analisar os serviços, aplicações do usuário de acordo com a prioridade ou complexidade, ou seja, quando maior a prioridade, maior o SLA previsto, assim o centro de dados terá no máximo uma aplicação de alta prioridade por servidor físico reduzindo assim o número de perda de requisições, chegando num ponto de equilíbrio do ambiente.

3.2.8 Cenários de Estudo

Como discutido anteriormente, a abordagem propõe o desenvolvimento de uma plataforma, uma infraestrutura baseada em serviços para suporte ao gerenciamento eficiente de energia, assim como alocação e distribuição de recursos do centro de dados na nuvem. A ideia dessa dissertação no entanto é realizar testes analisando os resultados e demonstrando que é possível a implementação de um sistema de agentes

baseado em serviços, onde cada elemento tem sua responsabilidade, para a orientação das ações e economia de recursos dentro do centro de dados na nuvem, atendendo as exigências dos provedores e seus clientes.

Em contrapartida, no Laboratório de Redes e Gerência - LRG, foi desenvolvido por alunos pesquisadores, uma arquitetura para monitoramento de nuvens privadas, denominada PCMONS - *Private Cloud MOnitoring Systems* (CHAVES; URIARTE; WESTPHALL, 2010). O trabalho foi desenvolvido para monitorar as máquinas virtuais e máquinas físicas de um ambiente de computação em nuvem, em especial nuvens privadas. O sistema de monitoramento é extensível e modular, é compatível com a infraestrutura como serviço (IaaS) de código aberto Eucalyptus, sendo monitorada pela ferramenta de rede Nagios. Através dela pode-se obter informações sobre o centro de dados, como por exemplo, o *status* das máquinas físicas e o *status* das máquinas virtuais. Entendemos que essa implementação de monitoramento do ambiente é um primeiro passo para evolução real desta proposta que visa a otimização da alocação e distribuição de recursos. Com base nos dados monitorados o sistema poderá atuar de maneira proativa, delegando funções aos elementos do ambiente, concentrando máquinas virtuais e desligando máquinas físicas ociosas.

3.3 SUMÁRIO DO CAPÍTULO

A abordagem proposta traz mais eficiência nos centro de dados, abrangendo todos os elementos que compõe o centro de dados (sistema de energia, sistema de refrigeração, servidores, equipamentos de rede, sistema de iluminação, entre outros), de maneira fácil e descomplicada. Outras abordagens realizadas em pesquisas nesta área não contemplam toda essa facilidade de gestão como serviços, também não focando no ponto de equilíbrio entre a carga de entrada, SLA e a real necessidade do cliente, em momentos de pico de carga.

As normas regulam o comportamento dos agentes de uma forma holística, com a compreensão de todos os elementos que integram o ambiente. Ou seja, elas impõem restrições de comportamento num sistema que supõe que os agentes tem comportamento livre e amplo. Assim, por exemplo, um agente (1) é responsável em coordenar a distribuição de carga e um agente (2) é responsável em regular a potência do ar condicionado.

Num sistema reativo, o agente (2) regula mais potência quando ele percebe que o ambiente esta esquentando. Num sistema proativo, ele pode começar a regular a potência para cima quando houver (ou existir a tendência histórica de haver) mais carga entrando no sistema, o que implica mais máquinas operando em maior capacidade que ira eventualmente gerar mais calor e aumentar a temperatura. Num sistema multi-agente, agente (1) notifica agente (2) da situação (mais carga entrando) que já pode considerar a contra-medida proativamente.

Com mais elementos do ambiente nesse contexto multi-agente, tem-se uma solução multi-agente para computação em nuvem verde.

As regras dizem o que cada agente faz, o agente (1) coordena distribuição de carga e agente (2) regula o ar condicionado. Os padrões de interação são os que definem a comunicação entre os agentes, como eles conversam, sobre o que eles conversam, etc. As normas são as regras de alto nível para dar uma estabilidade ao sistema, ou seja, o agente (2) não pode sair regulando o ar condicionado para cima porque ele considera uma boa alternativa, isso implica em gasto maior de energia, ele esta fazendo pois existe uma norma de manter a temperatura do ambiente o mais perto possível de 25 graus, mesmo que para isso ele precise gastar mais energia, o que é aceitável. As normas é que determinam os balanços entre custo/beneficio das ações.

Para resolver questões da distribuição e alocação de máquinas virtuais, a pesquisa sugere a abordagem de migração de máquinas virtuais, juntamente com a realocação das mesmas considerando os diferentes tipos de aplicações, assim como diferentes níveis de QoS requeridos pelos clientes, juntamente com a monitoração proativa do ambiente trás melhorias significativas ao ambiente.

4 AMBIENTE E ESTUDO DE CASO

A validação da abordagem proposta nesta dissertação foi feita através da avaliação em estudo de casos, com a utilização do simulador *CloudSim* (CALHEIROS et al., 2009), utilizando valores reais, valores matemáticos e de *benchmarks*, onde foram seguidos alguns planos de execuções. Utilizando essas possibilidades, pode-se comparar o comportamento da carga de trabalho no ambiente sem considerar a preocupação de eficiência energética, um modelo de arquitetura de Computação em Nuvem simplesmente e com a utilização do modelo minimalista de alocação e distribuição de recursos, o modelo de Computação Verde.

Neste capítulo será abordado como foram feitas as análises dos resultados sobre a utilização da abordagem proposta, bem como ferramentas utilizadas nas medições, e os seus resultados.

4.1 AVALIAÇÃO DE DESEMPENHO

Basicamente existem três técnicas para a análise de desempenho e avaliação de sistemas: simulação, métodos analíticos e o monitoramento de ambientes reais.

A simulação possibilita a criação de um modelo com as mesmas características do ambiente real, para a análise do desempenho sem a necessidade de implantação do ambiente não gerando custos desnecessários. Considera-se assim uma abstração do ambiente real, um modelo lógico, que pode-se adaptar antes da real implantação, para um melhor ajuste (FILHO, 2008).

Os métodos analíticos basicamente são modelos matemáticos ou estatísticos, que simulam o sistema através de uma abstração numérica, com um pouco mais de erros mas também podem ser vistos como uma boa técnica de análise, se comparada com a aplicação direta em ambientes reais.

A monitoração de ambientes reais, pode ser considerada uma das melhores técnicas para avaliação de desempenho e qualidade do sistema, possibilita uma avaliação quantitativa de carga, sem erros, trazendo resultados mais próximos ao real. Porém essa técnica gera custos altos para a implantação de equipamentos e de aplicações (FILHO, 2008), assim como análise de diferente políticas, o que vai contra o estudo

realizado, que prega primeiro avaliar quais recursos serão necessários para posterior implantação.

Neste trabalho, utiliza-se inicialmente modelos analíticos para a avaliação inicial da proposta, considerando dados de equipamentos e cargas de trabalho obtidas de *benchmarks* (SPEC, 2010). Posteriormente buscou-se dados de cargas de trabalho de ambientes reais e foram realizadas diversas simulações através do *framework* de simulação denominado *CloudSim* (CALHEIROS et al., 2009).

4.2 ANÁLISE ANALÍTICA

Nesta análise foram inferidas duas cargas de diferentes aplicações, aplicações arbitrárias, com base em seus valores de *benchmarks*, extraídos do SPEC2006 (SPEC, 2010), considerando cargas de processamento de CPU, conforme visualizado na Figura 16, são elas: Serviços *Web* e Serviços de *Backup*.

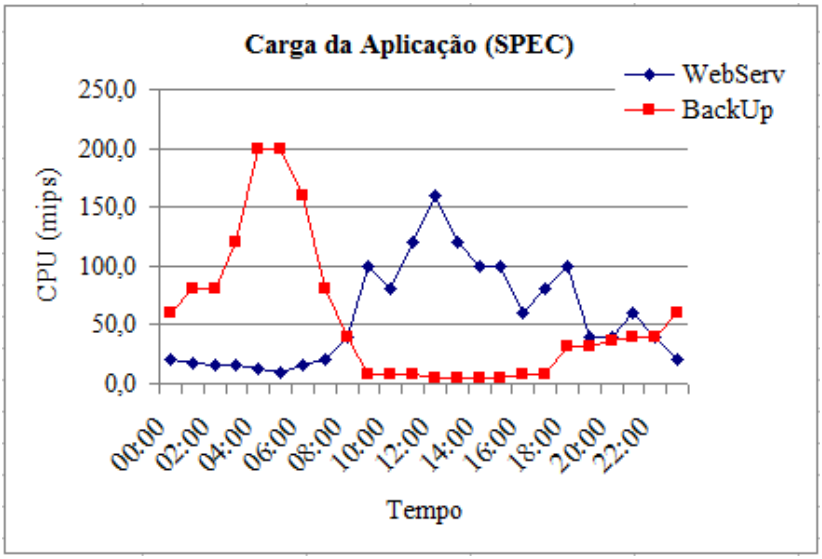

Figura 16 – Carga das Aplicações Num Dia.

As análises foram feitas dentro do período de um dia de trabalho, considerando as oscilações das cargas das aplicações.

Inicialmente para garantir a perfeita atividade, sem falhas, o modelo considera atender uma margem de 30%, além do pico máximo. O modelo matemático para Arquitetura Convencional analisa cada umas das três cargas de trabalho individualmente, considerando em cada uma delas seus pico máximo solicitado ao sistema. No modelo matemático para os ambientes de Computação em Nuvem e Computação em Nuvem Verde, o sistema analisa o valor total da soma das três cargas de trabalho, considerando o seu pico máximo total solicitado ao sistema.

Com isso determinamos o número de máquinas físicas nos ambientes de Arquitetura Convencional e Computação em Nuvem da mesma forma, conforme mostra a Equação 4.1.

No caso do ambiente de computação em Nuvem Verde, o modelo matemático considerou ainda determinar o número de *clusters* necessários para atender as cargas de trabalho, assim foi determinada uma divisão por 3 (três) *clusters*. A Equação 4.2 mostra a como é definida a quantidade de máquinas físicas na Nuvem Verde.

$$NumMF = \frac{PicoMax}{SpecServ} * 1 + Margem \tag{4.1}$$

$$NumMF = NumClus * \frac{PicoMax}{SpecServ} * 1 + Margem \tag{4.2}$$

Onde:

- NumMF = número de máquinas físicas;
- PicoMax = pico máximo da(s) carga(s) de trabalho;
- SpecServ = valor de processamento do servidor (SPEC);
- Margem = valor de margem do ambiente para garantir a qualidade;
- NumClus = número de *clusters* no ambiente.

Foi utilizada uma planilha eletrônica, desta forma na entrada do sistema foram inseridas as cargas, no processamento foram feitas as divisões de acordo com o modelo requerido e calculadas a eficiência energética prevista, ao final, a saída do sistema representa a variação do consumo energético dos três modelos.

Nesta etapa busca-se mostrar somente a variação do consumo de energia nas diferentes arquiteturas de centros de dados existentes, considerando os modelos descritos anteriormente (Arquitetura Convencional, Computação em Nuvem e Nuvem Verde).

Considerando a análise pela variação de carga de trabalho, onde na Arquitetura Convencional não existe alteração aa estrutura, basicamente tem uma máquina física ligada para cada aplicação de serviço independente da variação ou tipo de carga de trabalho, tendo um consumo de 100% dos recursos da infraestrutura.

Na arquitetura de Computação em Nuvem, é modelado o seu ambiente como um grande *cluster*, gerenciando todas as máquinas físicas através de máquinas virtuais, ou seja, tem uma gerência mais inteligente dos recursos, gerando uma redução no consumo de energia na ordem de 51% em relação a Arquitetura Convencional, forma distribuída.

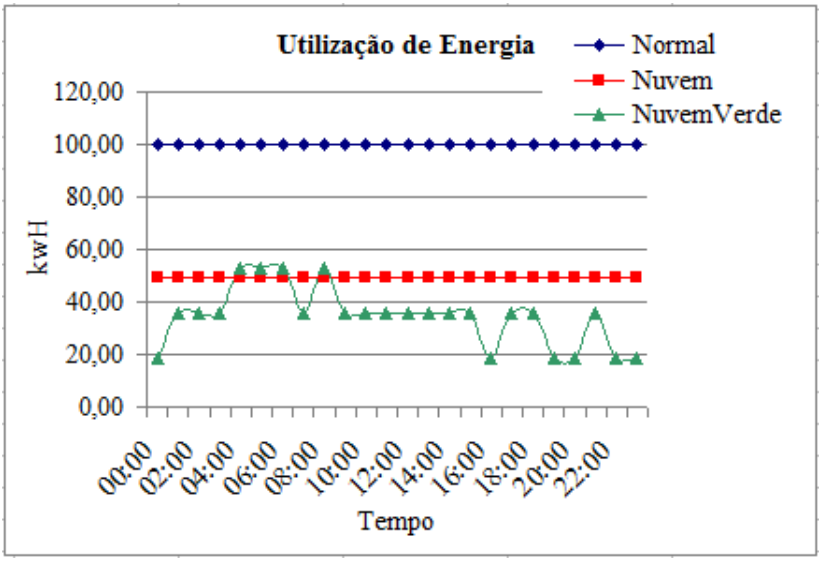

Figura 17 – Variação da Utilização de Energia Num Dia.

Já na arquitetura de Nuvem Verde, o modelo é dividido em *clusters*, ou seja, grupos de máquinas físicas e seus equipamentos de energia, de refrigeração, de rede, divididos em *clusters* e seus recursos tratados a nível de serviços e gerenciados autonomicamente, gerando uma redução no consumo de energia na ordem de 30%, em relação ao modelo de Computação em Nuvem. Neste modelo, conforme mostrado na Figura 17, a infraestrutura seria otimizada conforme a variação de carga de trabalho, havendo assim momentos em que a carga de processamento das aplicações é menor e, conseqüentemente, a redução no consumo de

energia é ainda maior.

Nesta etapa a pesquisa conseguiu mostrar de maneira bastante restrita que o consumo de energia dentro de um dia pode ser otimizado de acordo com as variações de processamento das aplicações, o que oportuniza uma melhor análise, mais detalhada, através da simulação pela qual pode-se analisar outros pontos críticos das arquiteturas de centros de dados.

4.3 SIMULAÇÃO

A simulação tem o intuito de projetar um modelo computacional de um sistema real e conduzir experimentos com este modelo com o propósito de entender seu comportamento e/ou avaliar estratégias para sua operação (PEGDEN; SADOWSKI; SHANNON, 1995).

Para a simulação da abordagem de teoria da organização e implementação das políticas de migração e realocação foram feitas modificações, melhorias no código do *framework* CloudSim, visualizadas no Anexo A. Com isso pode-se avaliar as abordagens propostas no Capítulo 3, além de possibilitar a reusabilidade dos modelos e o controle total do sistema (PEGDEN; SADOWSKI; SHANNON, 1995).

Na Tabela 5, foram definidas algumas características básicas dos elementos do ambiente, máquinas físicas e máquinas virtuais para as simulações.

Parâmetros	Valor
MV - Tamanho da Imagem	1000 MB
MV - Memória (RAM)	256 MB
MV - Largura de Banda	1 Mbps
MF - Arquitetura	x86
MF - Sistema Operacional	Linux
MF - VMM	Xen
MF - Memória (RAM)	8192 MB
MF - Velocidade do Processador	1000, 2000 e 3000 MIPS
MF - Armazenamento	1 TB
MF - Largura de Banda	100 Mbps
MF - Número de Processadores	2

Tabela 5 – Características do Cenário de Simulação Proposto

Os dados foram definidos de maneira a representar a realidade de um centro de dados, tendo como base o centro de dados em produção na universidade, que possuí basicamente máquinas físicas heterogêneas e diferentes necessidade de configurações de máquinas virtuais.

No simulador foram criados dois *clusters* com 100 máquinas físicas cada, de maneira a possibilitar a análise da abordagem proposta.

4.3.1 Simulador de Computação em Nuvem - *CloudSim*

De forma a modelar e simular aplicações que atendam os requisitos decorrentes do paradigma de computação em nuvem, foi disponibilizada a ferramenta de código aberto *CloudSim*, desenvolvida pela Universidade de Melbourne da Astrália. O simulador *CloudSim* suporta a criação e gerenciamento de recursos virtualizados entregues sob demanda. É uma ferramenta possibilita usar as variáveis associadas, as que são consideradas mais relevantes neste estudo de caso são: custo de processamento, custo para uso de memória, custo para uso de armazenamento e custo para utilização de banda da rede (CALHEIROS et al., 2009).

É necessário entender que cada recurso que é utilizado na Nuvem possui o seu preço. Quanto mais banda ou mais memória for necessário para realizar a tarefa do usuário, maior será o preço associado a este serviço. Desta forma, é necessário fazer com que a alocação destes recursos seja sempre ótima, aumentando o custo/benefício do cliente, bem como utilizando somente os recursos necessários disponíveis na Nuvem (CALHEIROS et al., 2009).

Estes custos estão associados às estruturas que controlam as políticas de alocação de cada recurso, os centros de dados. Além de definir a quantidade de cada recurso que será alocado, estas estruturas armazenam informações sobre as máquinas e a arquitetura relacionada a cada uma delas, bem como as propriedades que estas possuem (CALHEIROS et al., 2009).

Com o CloudSim é possível modelar vários aspectos do funcionamento de uma nuvem, como a política de alocação de máquinas virtuais, comportamento da rede, nuvens federadas, cargas de trabalhos dinâmicas, consumo de energia de um datacenter e criação dinâmica de entidades.

4.3.1.1 Arquitetura *CloudSim*

A Figura 18 mostra a arquitetura do *CloudSim*, para que seja possível entender melhor como a ferramenta funciona.

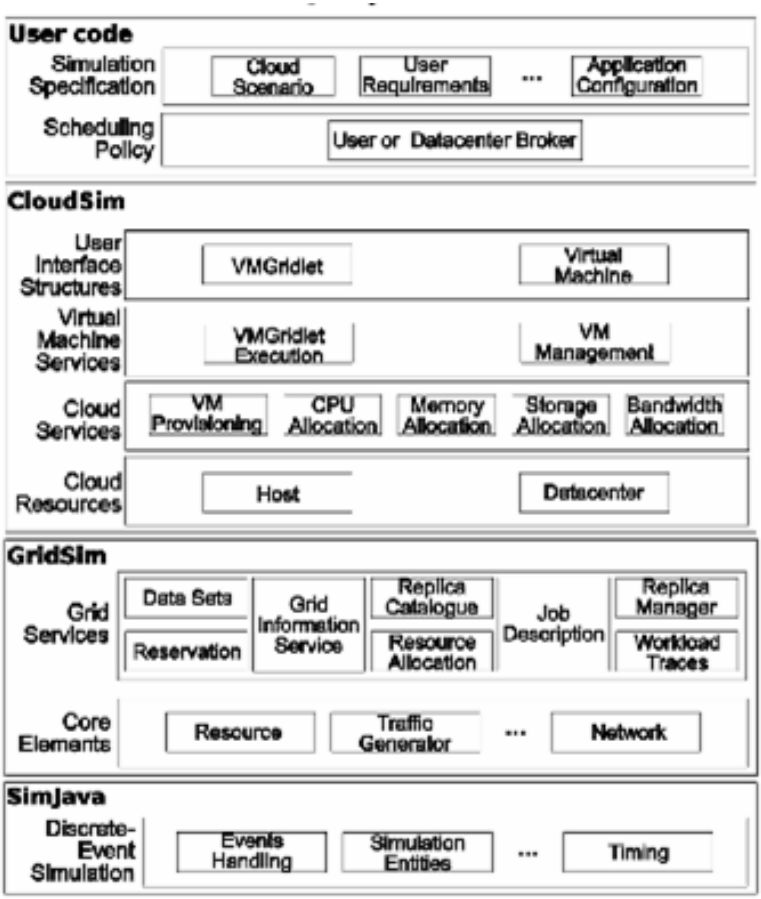

Figura 18 – Arquitetura *CloudSim* (CALHEIROS et al., 2009).

A arquitetura do *CloudSim* constitui-se em um núcleo, uma camada intermediária e uma camada superior. No núcleo é processada a simulação. Na camada intermediária estão as classes que simulam o ambi-

ente de computação em nuvem, como a máquina física, o *data center*, a máquina virtual, o elemento de processamento, além de outras classes que servem como apoio as principais, como a alocação e o escalonador de máquinas virtuais, o modelo de utilização (para memória, largura de banda e processamento), o escalonador de requisições, os modelos de consumo, as distribuições matemáticas, entre outras diversas classes. Por fim existe uma camada superior que é onde está o cenário e as configuração da simulação em si.

4.3.1.2 Recursos da *Cloud*

Para os recursos da Nuvem, cada centro de dados é composto por servidores, computadores físicos que possuem uma quantidade definida de capacidade de processamento, armazenamento, entre outros. Estes servidores são nós da Nuvem, pois é sobre eles que as máquinas virtuais serão geradas (CALHEIROS et al., 2009).

4.3.1.3 Serviços da *Cloud*

Nos serviços da Nuvem, cada Nuvem deverá possuir recursos a serem utilizados. Os mais importantes são CPU, memória, espaço para armazenamento e alocação de banda. Estes recursos são disponibilizados pela Nuvem com a criação de uma máquina virtual (CALHEIROS et al., 2009).

4.3.1.4 Serviços da Máquina Virtual

Nos serviços da máquina virtual, cada camada controla e gerência os recursos e o funcionamento das máquinas virtuais que são criadas para o processamento das aplicações do usuário (CALHEIROS et al., 2009).

4.3.1.5 Estrutura de Interface do Usuário

Nas estruturas da interface do usuário, o usuário necessita rodar sua aplicação na Nuvem, sobre uma ou mais máquinas virtuais. Para tanto,

é disponibilizada a estrutura ou uma parte destas máquinas para que a mesma possa realizar suas tarefas (CALHEIROS et al., 2009).

4.3.1.6 Comunicação das Entidades

Além de entender como que o *CloudSim* foi construído, é fundamental também ver como que suas entidades se comunicam. A Figura 19 mostra como este processo é feito desde o registro de uma nova máquina até a finalização de toda a simulação, o seu fluxo de comunicação(CALHEIROS et al., 2009).

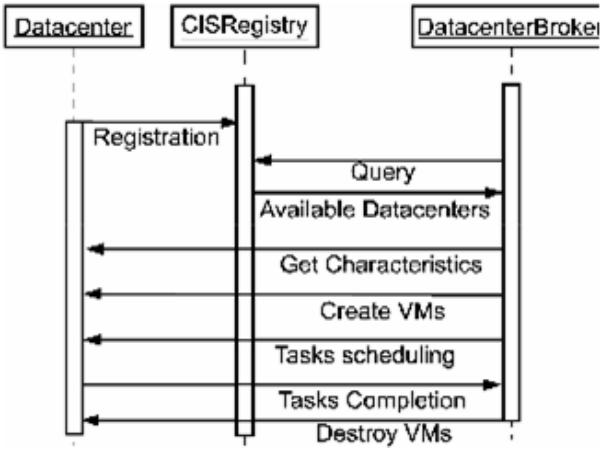

Figura 19 – Comunicação das Entidades do *CloudSim* (CALHEIROS et al., 2009).

Inicialemte os provedores registram seus centros de dados no CIS (*Cloud Information Service* - uma base de dados que mapeio os processos). Em seguida o usuário envia suas MVs e aplicações ao *Broker* para execução na infraestrutura oferecida pelo provedor. O *Broker* por sua vez consulta o CIS para identificar os provedores de serviços que atendam os requisitos de QoS. O CIS retorna os centros de dados disponíveis com suas características. Uma vez identificados, o *Broker* negocia a alocação de uma ou mais máquinas físicas no centro de dados para hospedar as MVs que irão executar as tarefas. Depois de realizada a

negociação, as MVs passam a executar as tarefas em uma ou mais MFs alocadas. A MF é responsável pela gestão do ciclo de vida das MVs e pode instanciar várias máquinas virtuais simultaneamente e atribuir seus núcleos de processamento de acordo com políticas de alocação de recursos predefinidas.

4.3.2 Implementações no CloudSim

Para a simulação no *CloudSim* das abordagens propostas no Capítulo 3 foram necessárias implementar algumas classes novas adaptando o *framework* do simulador de forma a contemplar as novas políticas, assim como o modelo de gerência dinâmico e pró-ativo.

A Figura 20 mostra as classes do *CloudSim* e as adaptações feitas para atender esse trabalho, são elas: *HostMonitor, VmMonitor, NewBroker, SensorGlobal, CloudletSchedulerSpaceShareByTimeout, VmAllocationPolicyExtended, VmSchedulerExtended, UtilizationModelFunction, CloudletWaiting e DatacenterExtended*.

As implementações do código do simulador *CloudSim* são descritas a seguir:

HostMonitor: controle de entrada e saída de máquinas físicas;

VmMonitor: controle de entrada e saída de máquinas virtuais;

NewBroker: controle do tamanho das requisições;

SensorGlobal: controle dos sensores;

CloudletSchedulerSpaceShareByTimeout: controle do tempo de agendamento das tarefas;

VmAllocationPolicyExtended: políticas de alocação das máquinas virtuais;

VmSchedulerExtended: controle da execução das máquinas virtuais;

UtilizationModelFunction: modelo matemático, função de utilização;

CloudletWaiting: controle do tempo de execução das tarefas;

DatacenterExtended: controle do centro de dados.

Foram feitas extensões em várias partes do simulador para atender os requisitos dos experimentos. Algumas classes só tiveram alteradas implementações de alguns métodos, mas outras tiveram que serem feitas outras alterações para que fossem atendidos os requisitos.

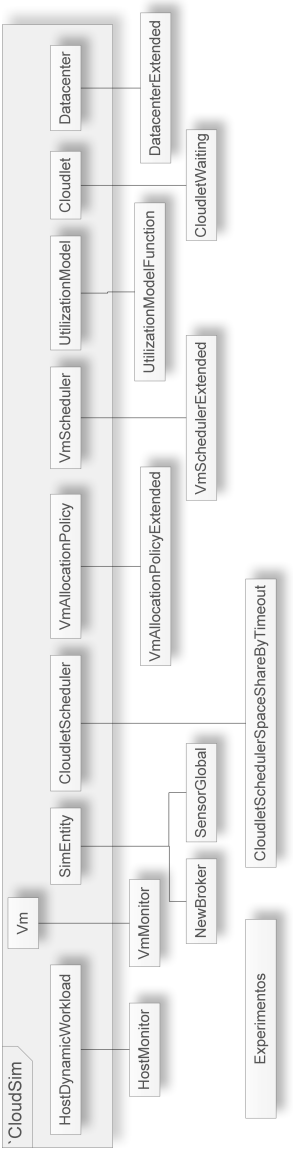

Figura 20 – Classes Implementadas no CloudSim

A parte de máquinas virtuais e máquinas físicas foram adicionados métodos que verificassem a carga em um certo instante de tempo, tais como a adição de limiares para alertar que a carga passou de um limiar mínimo ou máximo.

Foi feita a implementação de política de alocação de máquinas virtuais em máquinas físicas, que aloca as máquinas virtuais nas máquinas físicas com o maior número de máquinas virtuais alocadas mas que ainda possuem espaço para máquinas virtuais serem alocadas, ao contrário da implementação do simulador, que aloca na máquina física que possui menos máquinas virtuais alocadas.

A alteração na classe *Datacenter* foi realizada para alterar como é feita a leitura do consumo de energia do centro de dados.

Na classe *Cloudlet* foi criada uma extensão para que fosse adicionado um atributo que guardasse o tempo máximo em que a requisição pudesse ser executada, caso a execução ultrapasse esse tempo máximo, a requisição é contada como perdida e é contabilizada. Também foi refeito o método que atualiza o processamento das máquinas virtuais para que verificasse o tempo de execução de uma requisição.

Foi criada uma nova classe que faz a monitoração em intervalos de tempo junto com as classes das máquinas virtuais e máquinas físicas, executando certos procedimentos quando a carga ultrapassar certos limiares de acordo com o experimento em específico.

E por fim, foi necessário criar também um novo gerenciador de criação das máquinas virtuais e de envio de requisições para o centro de dados, onde também foram implementadas as duas cargas de envio de requisições citadas anteriormente.

4.3.3 Modelos de Carga Real

Nesta seção apresenta-se o modelo de carga de trabalho inferida nos experimentos para a realização dos testes.

Considerando os dados de carga de trabalho conforme Figura 22 e os dados levantados de processamento de CPU conforme Figura 21.

Foi utilizada uma distribuição derivada de medições em ambientes reais, durante o período de uma semana, de um servidor de serviços do centro de processamento de dados da universidade. Este servidor foi escolhido devido ao seu modelo de *hardware*, assim como pelos serviços que estava processando. O *hardware* deste servidor é padronizado e

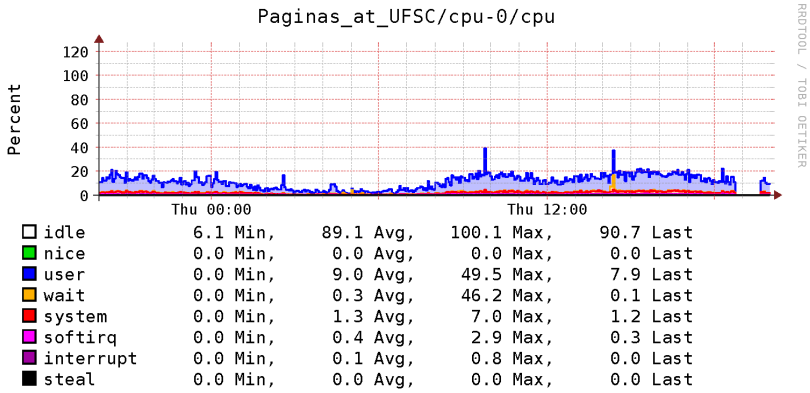

Figura 21 – Processamento CPU do Servidor Páginas

possui suas características de performance publicadas na *Internet*, através de seus *benchmarks*. Outra característica importante é que este servidor prove o serviço de hospedagem de centenas de páginas *web*, sendo difícil prever a demanda de pico de carga de trabalho, representando assim a realidade de muitos centro de dados.

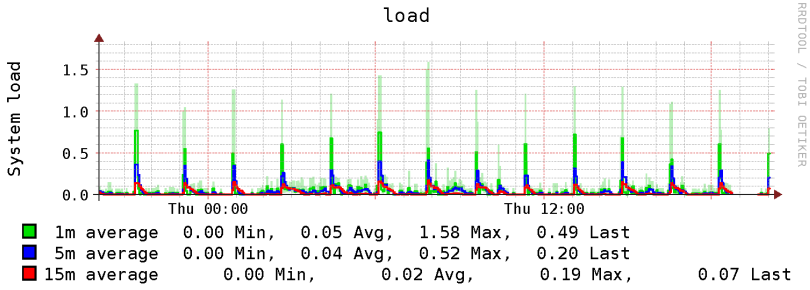

Figura 22 – Carga do Servidor Páginas

4.4 EXPERIMENTOS

Como citado na Seção 4.3.2 foram realizadas algumas modificações no código do *CloudSim* no intuito de realizar os experimentos. Quatro cenários foram simulados com o intuito de buscar a análise comparativa entre os métodos existentes e a abordagem proposta, são eles:

- Sem Políticas de Alocação de Recursos;
- Migrando as Máquinas Virtuais;
- Realocando as Máquinas Virtuais;
- Migrando e Realocando as Máquinas Virtuais.

Com a implementação e a análise dos experimentos, buscou-se basicamente os resultados de facilidade, flexibilidade do ambiente em administrar a entrada e saída de elementos, assim como a disponibilidade em poder se adaptar a diferentes picos de carga de trabalho, ambiente dinâmico.

Foram analisados ainda critérios para a sustentabilidade, ou seja, reduzir o consumo de energia como um todo, desligando máquinas físicas ociosas, porém sempre garantindo um número satisfatório de requisições perdidas de maneira a cumprir diferentes níveis de serviço, adequando-se aos SLAs dos usuários. Ainda neste foco, buscou-se a redução nas utilizações de máquinas virtuais, bem como redução no número de migrações, com a alegação de que o aumento no número de migrações possa comprometer a qualidade de serviço do ambiente.

4.4.1 Sem Políticas de Alocação de Recursos

Neste primeiro experimento, foi simulado um modelo sem políticas de provisionamento de recursos, para obter parâmetros base, como consumo, disponibilidade e violações de SLA para comparação. Utilizando a estratégia de provisionamento de recursos sob demanda, ou seja, não aplicando qualquer otimização de recursos e energia. Foi definida uma infraestrutura fixa na qual pudesse atender todos o momentos previsíveis das cargas inferidas no centro de dados, desta forma o centro de dados fica executando com 100% da utilização do processamento (CPU) dos servidores.

O experimento foi realizado conforme mostra a Figura 23, dois grupos de máquinas físicas com 100 máquinas cada, disponibilizando todos os

recursos da infraestrutura constantemente. Foram inferidas as cargas de trabalho de aplicações imprevisíveis, e foi monitorado o comportamento do ambiente.

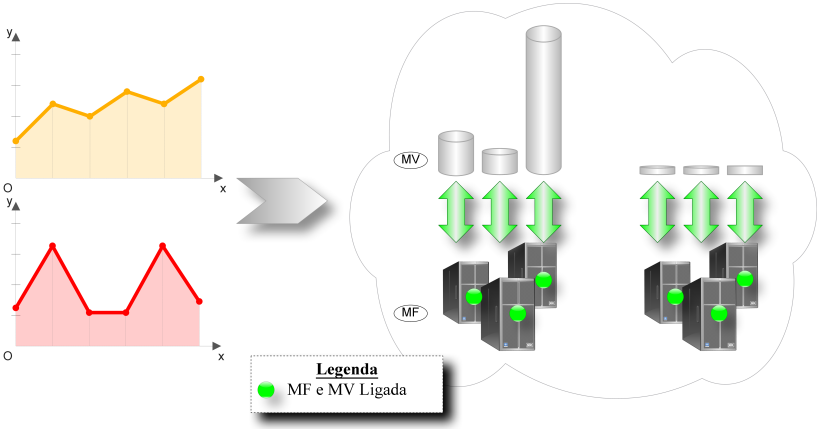

Figura 23 – Experimento Sem Polícicas de Alocação.

Um ambiente de Nuvem Verde presa por estratégias para economia de recursos, tratando questões de sustentabilidade. As pesquisas existentes tentam otimizar o provisionamento de recursos sob demanda, diminuindo ciclos de processamento, ou até mesmo reaproveitando equipamentos antigos. Contudo as pesquisas não garantem uma alta disponibilidade dos serviços, uma boa qualidade de serviço, caso haja um pico de *Workload* (do inglês, carga de trabalho).

Foram assim levantados dados de Consumo de Energia, Requisições Perdidas (Violações de SLA), Tempo de Processamento para comparação posterior. Utilizou-se as características de máquinas físicas e virtuais já pré estabelecidas na Tabela 5. Dado que esta é uma simulação para base de comparação, espera-se obter um alto consumo de energia, um número baixo de perda de requisições e nenhuma violação de SLA. O gerenciador deve rodar todo o centro de dados em plena execução sem qualquer política de provisionamento de recursos, atendendo todas as requisições sem restrição. A carga de trabalho deve ser variável, de modo que possamos avaliar os casos de pico de carga. O ambiente sempre será subutilizado, pois temos muito processamento para pouca carga, em exceção nos momentos de pico de carga da aplicação.

4.4.2 Migração de Máquinas Virtuais

Neste experimento utilizou-se a estratégia de Migração de Máquinas Virtuais, como recurso (*on line*), verificando, analisando e tratando o comportamento e as variações das carga de trabalho, nas máquinas físicas existentes, para prover economia substancial de recursos no centro de dados, incluindo energia, ar condicionado e dos próprios servidores. Aumentando a taxa de disponibilidade acima de 99,9% do tempo, assim como diminuir o número de violações de SLA e o número de máquinas físicas no centro de dados.

A Figura 24, ilustra dois *clusters* de 100 máquinas físicas, onde seriam executadas aplicações imprevisíveis, e conforme a demanda seriam criadas, alocadas máquinas virtuais. No decorrer das tarefas seriam avaliadas as tarefas e migradas as máquinas virtuais, concentrando num menor número de máquinas físicas, desligando assim máquinas físicas obsoletas.

Neste caso argumenta-se que a migração de máquinas virtuais traz grandes benefícios para a redução de máquinas físicas ligadas, e para diminuição do tempo de processamento, assim como redução no número de requisições perdidas, contribuindo para uma ambiente minimalista.

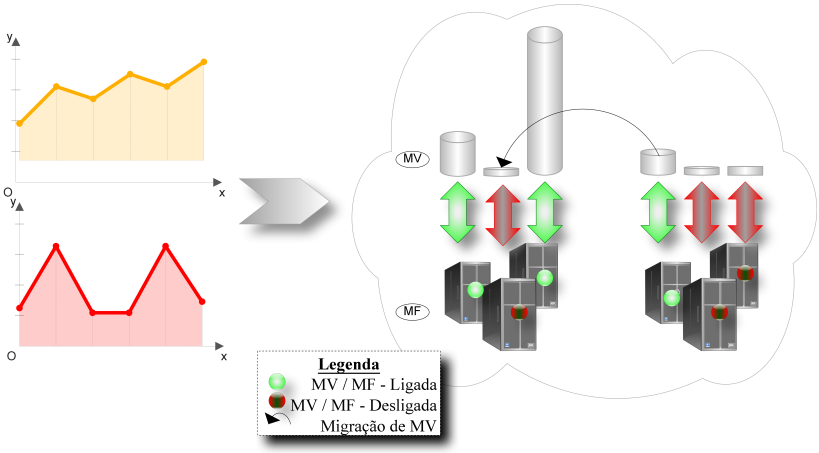

Figura 24 – Experimento de Migração.

A carga de trabalho deve ser variável, de modo que possa-se avaliar os casos de pico de carga. O gerenciador deve monitorar os recursos das máquinas físicas, assim como as máquinas virtuais em tempo real, no caso de uma máquina virtual com menos de 20% de utilização num dos centros de dados, esta deve ser migrada para o outro centro de dados e a máquina física anterior desligada.

4.4.3 Realocação de Máquinas Virtuais

Neste experimento é verificado se a estratégia de utilizar a "Realocação de Máquinas Virtuais", como recurso em tempo real (*on line*), verificando, analisando e tratando as variações de carga de trabalho, nas máquinas virtuais existentes, provê economia substancial de recursos do centro de dados, incluindo energia, ar condicionado e dos próprios servidores. Aumentando a taxa de disponibilidade acima de 99,9% do tempo, assim como diminuir o número de violações de SLA e o número de migrações devido ao aumento de requisições.

Neste caso argumenta-se que somente a migração de máquinas virtuais possa comprometer os níveis de qualidade de serviço, e até mesmo promover a perda de tempo para migração.

A Figura 25, ilustra dois *clusters* de 100 máquinas físicas, onde seriam executadas aplicações imprevisíveis, e conforme a demanda seriam criadas, alocadas máquinas virtuais. No decorrer das tarefas seriam avaliadas as tarefas e ajustadas as máquinas virtuais, de maneira que não fosse necessário alocar outra MF ou máquinas virtual, evitando ligar máquinas físicas.

A carga de trabalho deve ser variável, de modo que possa-se avaliar os casos de pico de carga. O gerenciador deve verificar nos momentos de pico de carga de trabalho, se a máquina virtual chegar a 90% de utilização deve aumentar de 10% em 10% na medida que começar a receber mais requisições.

4.4.4 Migração e Realocação de Máquinas Virtuais

Este experimento visa verificar se a estratégia de utilizar a "Realocação de Máquinas Virtuais"em conjunto com estratégia de "Migração de Máquinas Virtuais", como recurso em tempo real (*on line*), verificando,

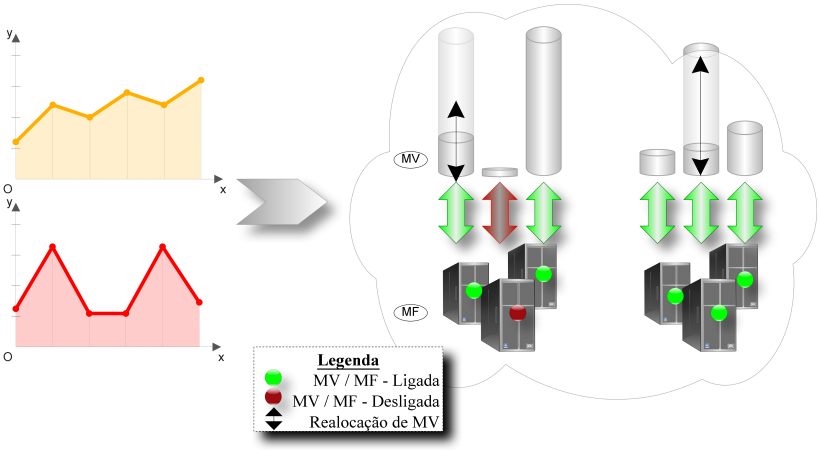

Figura 25 – Experimento de Realocação.

analisando e tratando as variações de carga de trabalho, nas máquinas virtuais e nas máquinas físicas, provê economia substancial de recursos do centro de dados, incluindo energia, ar condicionado e dos próprios servidores. Aumentando a taxa de disponibilidade acima de 99,9% do tempo, assim como diminuir o número de violações de SLA e o número de migrações devido ao aumento de requisições.

Neste caso argumenta-se que somente a migração de máquinas virtuais possa comprometer os níveis de qualidade de serviço, e até mesmo promover a perda de tempo para migração.

A Figura 26, ilustra dois *clusters* de 100 máquinas físicas, onde seriam executadas aplicações imprevisíveis, e conforme a demanda seriam criadas, alocadas máquinas virtuais. No decorrer das tarefas seriam avaliadas as tarefas e ajustadas ou migradas máquinas virtuais.

A carga de trabalho deve ser variável, de modo que possa-se avaliar os casos de pico de carga. O gerenciador deve verificar nos momentos de pico de carga de trabalho, se a máquina virtual chegar a 90% de utilização deve aumentar de 10% em 10% na medida que começar a receber mais requisições. O gerenciador deve ainda monitorar os recursos das máquinas físicas, assim como as máquinas virtuais em tempo real, no caso de uma máquina virtual com menos de 20% de utilização num dos centros de dados, esta deve ser migrada para o outro centro de dados e a máquina física anterior desligada.

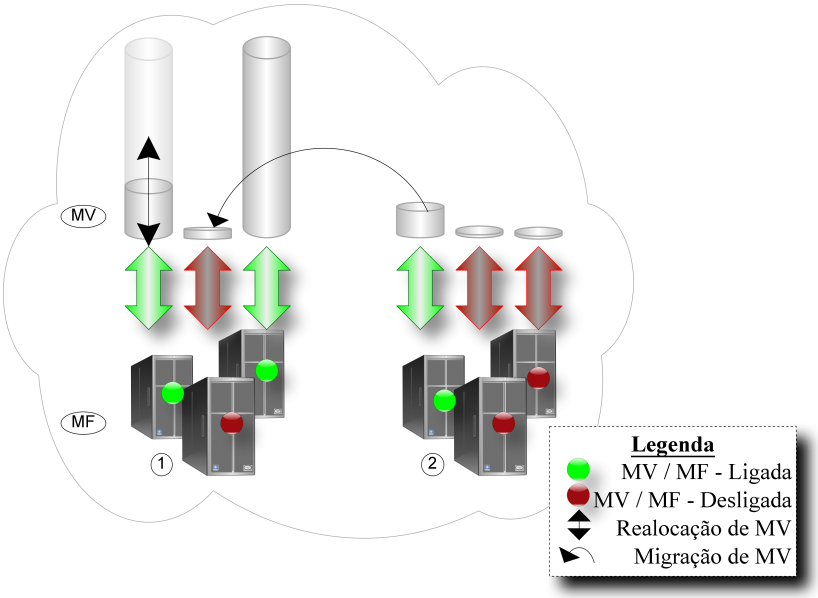

Figura 26 – Experimento de Migração e Realocação.

Neste experimento estão previstas ainda algumas possibilidades de variação nos limiares (inferior e superior) de utilização das máquinas físicas para a efetiva migração das máquinas virtuais, com o intuito de se obter um melhor ponto de equilíbrio, de acordo com os requisitos de QoS do usuário final.

4.5 DADOS E ESTATÍSTICAS

Nesta seção, são apresentados resultados de alguns testes quantitativos obtidos nas simulações dos experimentos citados na Seção 4.4. A implementação de toda a abordagem proposta no Capítulo 3 visa mostrar a eficiência energética do ambiente com o mínimo de requisições perdidas e ainda um mínimo de intervenções no ambiente, isso tudo em diferentes situações de uso e diferentes tipos de aplicações, ou seja, diferentes níveis de SLA.

4.5.1 Consumo de Energia

Pode-se perceber no gráfico apresentado na Figura 27, que o consumo de energia em ambientes de computação em nuvem sem políticas implementadas tem um consumo de energia regular durante todo o dia, ou seja, as máquinas virtuais, assim como as máquinas físicas estão sempre disponíveis durante todo o dia. Neste cenário a simulação de diferentes máquinas físicas, considerando um *cluster* com 100 máquinas físicas, teve um consumo de energia no valor 29.757 de kWatt/hora.

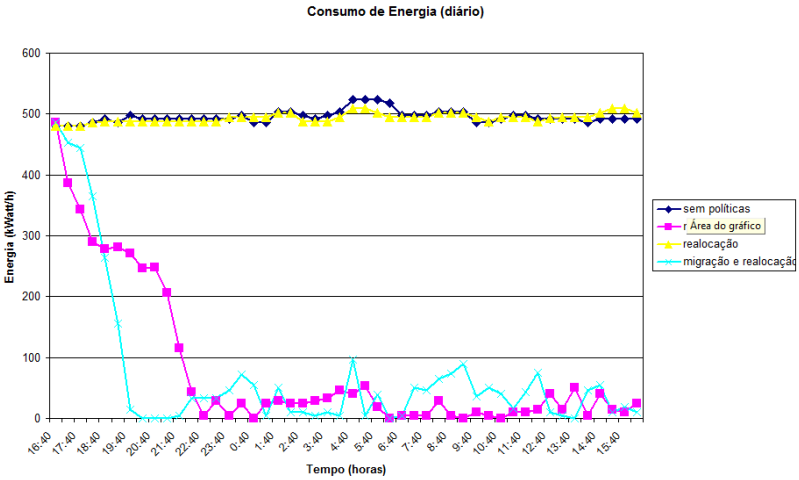

Figura 27 – Consumo Diário de Energia nos Experimentos.

No cenário da implementação da política de migração de máquinas virtuais, concentrando as máquinas virtuais em um mínimo possível de máquinas físicas, eliminando assim o consumo de energia de máquinas físicas cujo processamento esteja zerado, através do desligamento ou hibernação das máquinas físicas, coseguiu-se um ganho significativo como já previsto na analise analítica da Seção 4.2, ou seja, na simulação do cenário proposto pode-se vizualizar uma redução no consumo de energia em 84,33% de kWatt/hora, em relação ao cenário sem políticas implementadas.

Já no caso da implementação da política de realocação de máquinas virtuais, considerando que em alguns casos de aumento no processa-

mento, pico de carga de trabalho, não seria necessário a migração de máquinas virtuais para outras máquinas físicas, apenas seria feito o aumento de máquinas físicas disponíveis, readequando a disponibilidade de recursos no escalonador. No entanto, neste cenário de somente realocação, não foi observada uma redução significativa no consumo de energia, houve somente a redução 0,39% de kWatt/hora, em relação ao cenário sem políticas implementadas, não sendo assim eficaz.

Juntando as duas abordagens, conforme descrito no experimento da Seção 4.4.4, migrando e realocando as máquinas virtuais de acordo com a necessidade de carga de trabalha, o cenário mostrou-se eficaz com uma redução no consumo de energia na ordem de 87,18% de kWatt/hora, em relação ao cenário sem políticas implementadas.

A Figura 28 mostra a variação no consumo de energia durante uma semana de simulação considerando os picos de carga de trabalho de uma carga real. As abordagens propostas ajudam de maneira eficaz no tratamento das cargas imprevisíveis.

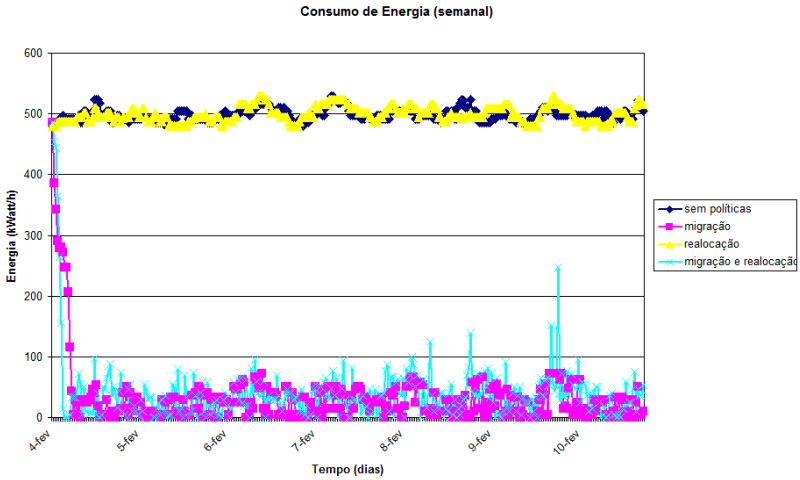

Figura 28 – Consumo Semanal de Energia nos Experimentos.

4.5.2 Violações de SLA

No gráfico apresentado na Figura 29, considerando um dia de simulação e monitoração, observa-se 1.171 requisições perdidas em ambientes de computação em nuvem sem políticas implementadas, ou seja, todos os equipamentos do centro de dados (isto é, *hardware*, sistema de energia, sistema de refrigeração) estão dimensionados para atender um média de de carga de trabalho, considerando sempre um SLA médio.

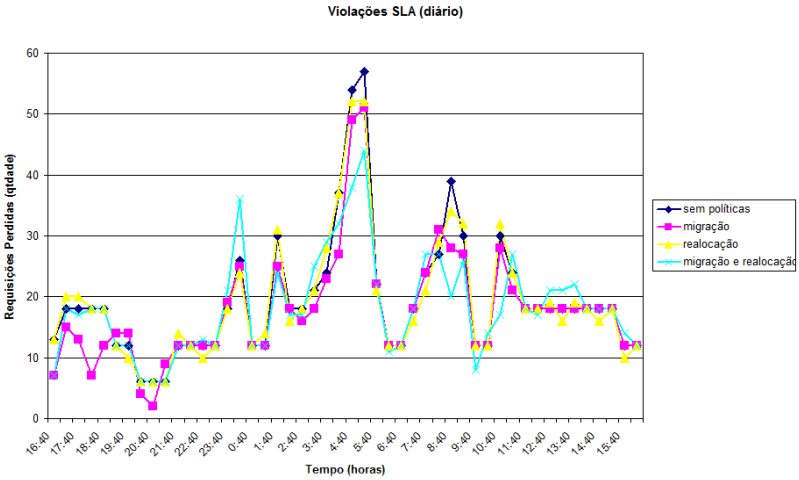

Figura 29 – Requisições Perdidas nos Experimentos num Dia.

No caso da implementação da política de migração de máquinas virtuais, concentrando as máquinas virtuais em um mínimo possível de máquinas físicas, e considerando que temos diferentes SLAs para cada cliente, foi reduzido o custo operacional num centro de dados, evitando perdas (isto é, multas, dados não processados, consumo de energia, etc.) por SLA. Com isso o *data center* com diferentes níveis de SLAs, teria perda de requisições em torno de 1.077 requisições perdidas, uma redução de 8% em relação ao centro de dados sem políticas implementadas, porém não afeta seus indicadores, pois perde requisições em SLAs menos criteriosos, definidos em conjunto com o cliente.

Já na implementação da política de realocação de máquinas virtuais, considerando que em alguns casos de aumento no processamento, pico

de carga de trabalho, não seria necessário a migração de máquinas virtuais para outras máquinas físicas, realocando as máquinas virtuais que tiverem um nível de SLA mais alto. A realocação não mostrou-se eficaz nestes casos não reduzindo o número de requisições perdidas no centro de dados de computação em nuvem verde.

Da mesma forma quando implementadas as duas abordagens juntas, migração e realocação de máquinas virtuais, a simulação de uma semana de carga de trabalho com diferentes momentos de picos de carga, conforme Figura 30, a redução no número de requisições perdidas, foi na ordem de 7,34%, em relação as cenário sem políticas implementadas, não sendo assim eficaz fazer a realocação de máquinas virtuais.

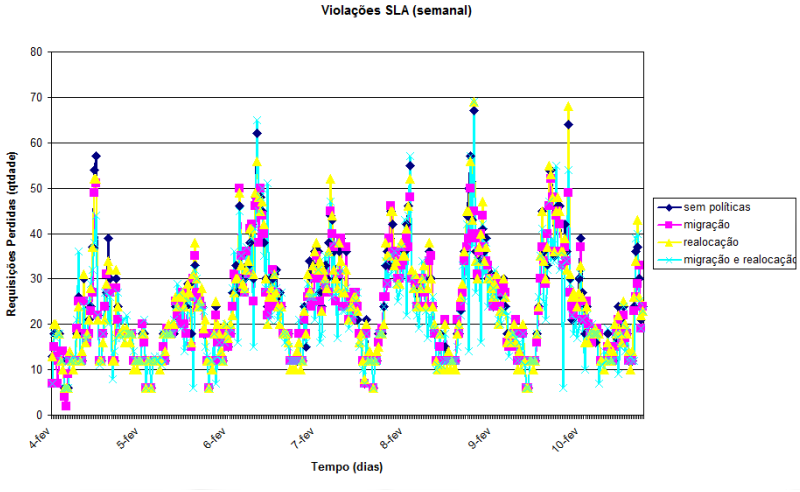

Figura 30 – Requisições Perdidas nos Experimentos numa Semana.

4.5.3 Migrações e Realocações

Inicialmente considerou-se que não há migrações de máquinas virtuais em ambientes de computação em nuvem sem políticas implementadas, ou seja, as máquinas virtuais ficam fixas nas máquinas físicas.

No caso da implementação da política de migração de máquinas virtuais, concentrando as máquinas virtuais em um mínimo possível de

máquinas físicas, existe uma quantidade significativa de migrações de máquinas virtuais, pois acaba considerando a prioridade de manter um número variável de máquinas virtuais ativas. Com a demanda de muitos processos por vez acaba ocorrendo diversas migrações com o intuito de diminuir o consumo. No período de um dia de trabalho no centro de dados houveram 6.028 migrações de máquinas virtuais, conforme visualizado na Figura 31.

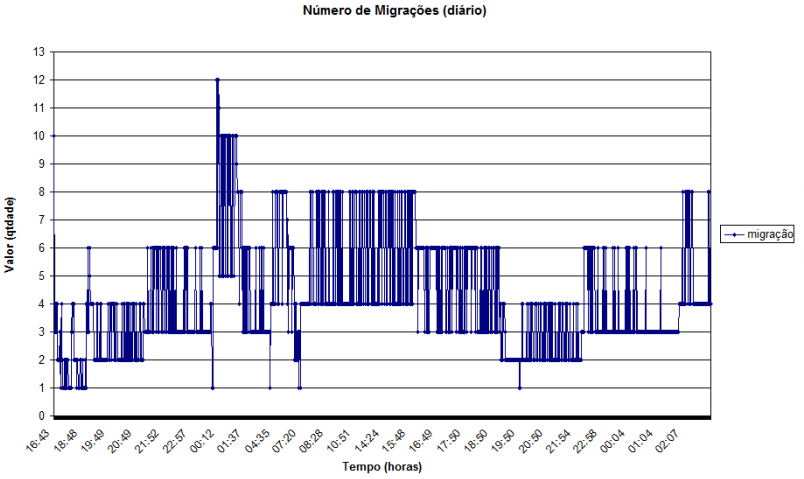

Figura 31 – Quantidade de Migrações num Dia.

Já na implementação da política de realocação de máquinas virtuais, considerando que em alguns casos de aumento no processamento, pico de carga de trabalho, não seria necessário a migração de máquinas virtuais para outras máquinas físicas, ocorreram 134 realocações de máquinas virtuais, conforme distribuição apresentada na Figura 32.

Considerando a implementação das abordagens de realocação em migração de máquinas virtuais ocorrendo ao mesmo tempo, conforme visualizado na Figura 33, houve uma redução significativa no número de migrações de MV, na ordem de 45,93%. Demonstrando assim uma qualidade importante na abordagem, contribuindo para a estabilidade do ambiente de computação em Nuvem Verde.

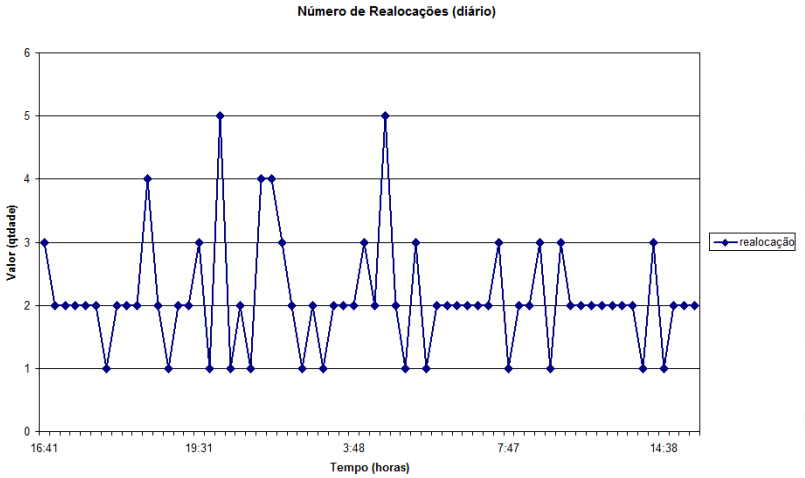

Figura 32 – Quantidade de Realocações num Dia.

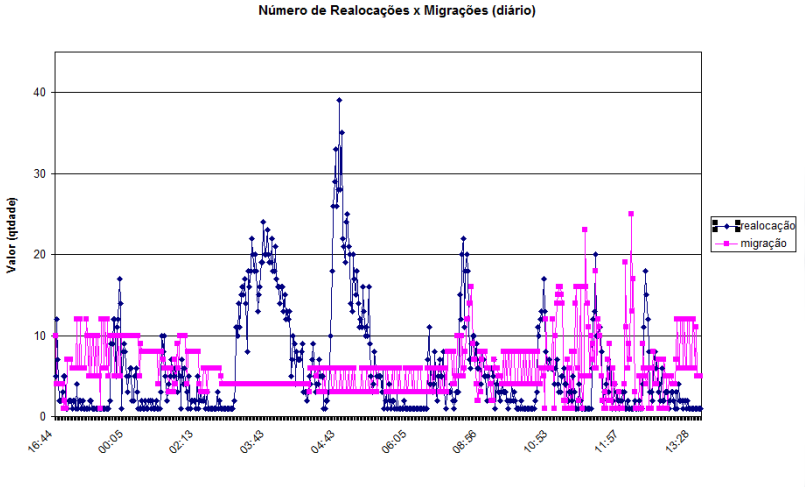

Figura 33 – Quantidade de Migrações e Realocações num Dia.

4.6 MELHORIAS NO AMBIENTE DE NUVEM VERDE

Com as ações pró-ativas de **Flexibilidade** do ambiente, demonstra-se que o número médio de máquinas físicas ligadas vão diminuir em 20%,

para o cenário descrito acima.

Para melhorias na **Disponibilidade** do ambiente, espera-se diminuir as violações de SLA, dependendo do cenário, em até 8%. Desde a possibilidade de grupos isolados do Serviços de Nuvens em diferentes cenários de carga inesperados (tais como ataques de negação de serviço), outros serviços não seriam afetados pela saturação da Nuvem.

Com melhorias de **Custos** do ambiente, houve uma redução significativa do CAPEX, para o cenário descrito acima, visulaizou-se uma redução de aproximadamente 80% do consumo de energia do centro de dados global. A redução do CAPEX seria derivada do corte de máquinas físicas (configuração mínima). E a redução de OPEX vem com a otimizações no uso dos recursos e do corte de máquinas físicas.

A pesquisa alcançou na **Sustentabilidade**, com a gerência do sistema de refrigeração, com o modelo proposto, os centros de dados poderiam reduzir o consumo no sistema de refrigeração, de 64.824 milhões (100%) BTUs anuais para 31.763 milhões (49%) anuais, atingindo uma economia de emissão de calor de aproximadamente 50% neste cenário. O que contribui ainda mais para a sustentabilidade dos centros de dados, pois com a redução no número de máquinas físicas e a divisão por *clusters* tem-se uma redução nos equipamentos do sistema de refrigeração.

4.7 SUMÁRIO DO CAPÍTULO

Com base nos dados de desempenho da análise analítica e principalmente com base na simulação do ambiente com dados reais, percebe-se algumas habilidades de um sistema baseado em elementos como serviço.

Os resultados alcançados ainda com as políticas de migrações de máquinas virtuais e realocações de máquinas virtuais, ajustando a utilização das máquinas físicas promoveram uma melhoria na sustentablidade do ambiente tornando-o flexível, disponível e com o custo/benefício adequado a realidade de centros de dados heterogêneos.

5 CONCLUSÕES

Neste trabalho, foi proposto uma abordagem para alocação de máquinas virtuais em ambientes de *Nuvem Verde*, baseada em uma base teórica sólida, com suporte à auto-gerenciamento. A grande questão a ser respondida era: qual o modelo de alocação e distribuição de máquinas virtuais em ambientes de Computação em Nuvem Verde (isto é, configuração interna) que garanta níveis de performance de serviço contínuos e aceitáveis em resposta a variações de carga imprevisíveis?

A solução proposta consiste na criação de uma estrutura baseada em serviços, assim como o modelo de *Teoria da Organização*, capazes de sentir o ambiente em que estão e atuar no mesmo de acordo com políticas pré-definidas. É exatamente a maleabilidade da organização dos elementos, ou seja, na alocação dos recursos como serviços, que proporciona à *Nuvem Verde*, a capacidade de auto-gestão desejada.

Para proporcionar auto-gerenciamento distribuído de forma fácil, garantir os níveis de performance de serviço contínuos e aceitáveis em resposta a variações de carga imprevisíveis, e reduzir o consumo de recursos, o sistema deve suportar suas três sub-áreas: políticas de migração de máquinas virtuais, políticas de realocação de máquinas virtuais e um modelo baseado em teoria da organização. Este trabalho abordou estas sub-áreas da seguinte forma:

- com a capacidade do modelo em efetuar diagnósticos e adaptações sobre o sistema, controlando situações inesperadas de acordo com regras pré estabelecidas, não importando a complexidade do ambiente.
- com a capacidade dada ao sistema de se configurar sem nenhuma intervenção externa, deslocando máquinas virtuais para outras máquinas físicas.
- com a capacidade do sistema de se otimizar de acordo com a disponibilidade das máquinas físicas, adaptando as máquinas virtuais subutilizadas.

Verificou-se através dos testes realizados que atendendo as três subáreas, consegue-se alcançar a eficiência desejada no início do trabalho, respondendo a pergunda principal.

Os testes apresentados comprovam a validade do sistema, para os testes foi utilizado o simulador de computação em nuvem, denominado

CloudSim da Universidade de Melbourne, da Austrália. No simulador foram implementadas melhorias para a interação baseada em serviços proposta no modelo de *Teoria da Organização*. Na seqüência foram implementadas as políticas de migração e realocação de máquinas virtuais. Por fim, através do monitoramento e controle do modelo de *Teoria da Organização*, verificou-se uma redução no número de migrações (45% em média considerando um dia de simulação), assim como no número de violações de SLA, constatado pela redução do número de requisições perdidas (7,34% em média considerando um dia de simulação). Além disso, a abordagem simplifica o modelo de gerenciamento, na qual consegue-se gerenciar os recursos (ligando / desligando máquinas) de cada elemento, reduzindo o consumo de energia do centro de dados (87% em média).

Utilizar a *Teoria da Organização* em ambientes de computação em nuvem verde mostrou-se eficiente, pois além de permitir ao sistema se auto-configurar, controlando a entrada e saída de recursos, é possível garantir, através da auto-otimização, da implementação de novas políticas, a eficiência energética, ou seja, sustentabilidade da computação em nuvem.

5.1 PRINCIPAIS CONTRIBUIÇÕES

Este trabalho tem como principal contribuição a apresentação de uma abordagem utilizando um modelo de gerenciamento baseado em Teoria da Organização, juntamente com políticas de migração e políticas de realocação de recursos, para proporcionar uma melhor eficiência na gestão de recursos, tentando resolver problemas do gerenciamento centralizado.

Na Seção 2.7 foram analisados trabalhos de pesquisa que envolvam gerenciamento de recursos dinâmicos, que tentam de alguma forma a melhoria no gerenciamento de recursos e a eficiência, porém na análise nenhum dos trabalhos atendeu totalmente aos requisitos desejados no trabalho, requisitos de **Flexibilidade**, **Disponibilidade**, **Custo** e **Sustentabilidade**, que neste trabalho foram alcançados, apresentados na Tabela 6.

Desta a contribuição inerente a este trabalho é a de proporcionar uma infraestrutura que:

Flexibilidade: o sistema garante uma orquestração física e lógica, pre-

	Flexib.	Dispon.	Custo	Sustent.
Energy-Efficient	Parcial	Sim	Sim	Parcial
Sandpiper	Parcial	Não	Sim	Não
Load Balancing	Parcial	Parcial	Parcial	Não
Green Cloud	Parcial	Não	Sim	Sim
Modelo Teoria da Org.	Sim	Sim	Sim	Sim

Tabela 6 – Comparação Principais Trabalhos

vendo e executando as mudanças necessárias de acordo com a demanda da Nuvem;

Disponibilidade: o sistema faz o balanceamento de carga automaticamente e tem alta disponibilidade de recursos;

Custo: o sistema adota uma configuração minimalista, assim como estratégias inteligentes para a utilização dos recursos;

Sustentabilidade: o sistema faz adaptações, para a redução no consumo de energia em todos os equipamentos.

5.2 TRABALHOS FUTUROS

Este trabalho teve como escopo propor uma abordagem para alocação de máquinas virtuais em ambientes de *Nuvem Verde* e mostrar que a mesma é viável. Com a conclusão do mesmo, verificou-se a abertura de outras questões que podem ser tratadas em trabalhos futuros:

- criação de um serviço responsável pela segurança no modelo de *Teoria da Organização*;
- execução de um estudo mais aprofundado sobre qualidade de serviço com o intuito de avaliar quais são os melhores parâmetros a serem controlados e priorizados;
- implementação e validação das políticas e modelos propostos em um ambiente real;
- análise e implementação de políticas associadas a redes neurais.

REFERÊNCIAS BIBLIOGRÁFICAS

ALLIANCE-CSA, C. S. *Security Guidance for Critical Areas of Focus in Cloud Computing*. 2009. <http://www.cloudsecurityalliance.org>. Acessado em 30/4/2009.

ALVAREZ-NAPAGAO, S. et al. Norms, organisations and semantic web services: The alive approach. *Workshop on Coordination, Organization, Institutions and Norms at MALLOW'09*, 2009.

ALVAREZ-NAPAGAO, S. et al. conciens: Organizational awareness in real-time strategy games. In: *Proceeding of the 2010 conference on Artificial Intelligence Research and Development: Proceedings of the 13th International Conference of the Catalan Association for Artificial Intelligence*. Amsterdam, The Netherlands, The Netherlands: IOS Press, 2010. p. 69–78. ISBN 978-1-60750-642-3. <http://portal.acm.org/citation.cfm?id=1893268.1893280>.

ARMBRUST, M. et al. *Above the Clouds: A Berkeley View of Cloud Computing*. [S.l.], Feb 2009. <http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html>.

BALEN, D. d. O. et al. Sistema para Gerência Autonômica de Grades Computacionais. *SBRC - Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos, WGRS - XIV Workshop de Gerência e Operação de Redes e Serviços*, Porto Alegre, RS, Brasil, 2009.

BARROSO, L. A.; HöLZLE, U. The Case for Energy-Proportional Computing. *Computer*, IEEE Computer Society Press, Los Alamitos, CA, USA, v. 40, n. 12, p. 33–37, 2007. ISSN 0018-9162.

BELOGLAZOV, A. et al. A taxonomy and survey of energy-efficient data centers and cloud computing systems. In: *Advances in Computers (in press, accepted on Sept. 6, 2010)*. [S.l.]: Elsevier, 2011. ISBN 978-0-12-012141-0.

BRANDIC, I. Towards self-manageable cloud services. *Computer Software and Applications Conference, Annual International*, IEEE Computer Society, Los Alamitos, CA, USA, v. 2, p. 128–133, 2009. ISSN 0730-3157.

BRYNJOLFSSON, E.; HOFMANN, P.; JORDAN, J. Economic and business dimensions: Cloud computing and electricity: beyond

the utility model. *j-CACM*, v. 53, n. 5, p. 32–34, may 2010. ISSN 0001-0782.

BUYYA, R.; BELOGLAZOV, A.; ABAWAJY, J. H. Energy-efficient management of data center resources for cloud computing: A vision, architectural elements, and open challenges. *CoRR*, abs/1006.0308, 2010. Informal publication. <http://dblp.uni-trier.de/db/journals/corr/corr1006.htmlabs-1006-0308>.

BUYYA, R.; RANJAN, R.; CALHEIROS, R. N. InterCloud: Utility-oriented federation of cloud computing environments for scaling of application services. In: *Proceedings of the 10th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP'10)*. Busan, Korea: Springer, 2010. p. 13–31.

BUYYA, R.; YEO, C. S.; VENUGOPAL, S. Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities. *High Performance Computing and Communications, 2008. HPCC '08. 10th IEEE International Conference on High Performance Computing and Communications*, Ieee, p. 5–13, 2008.

CALHEIROS, R. N. et al. Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services. *CoRR*, abs/0903.2525, 2009. Informal publication. <http://dblp.uni-trier.de/db/journals/corr/corr0903.htmlabs-0903-2525>.

CHANG, H. H.; CHOU, P. B.; RAMAKRISHNAN, S. An ecosystem approach for healthcare services cloud. In: *ICEBE*. IEEE Computer Society, 2009. p. 608–612. <http://dblp.uni-trier.de/db/conf/icebe/icebe2009.htmlChangCR09>.

CHAVES, S. A. de; URIARTE, R. B.; WESTPHALL, C. B. Implantando e Monitorando uma Nuvem Privada. *VIII Workshop em Clouds, Grids e Aplicações - Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, Porto Alegre, RS, Brasil, p. 1–12, 2010.

COULOURIS, G.; DOLLIMORE, J.; KINDBERG, T. *Sistemas Distribuídos - Conceito e Projeto*. 4 ed.. ed. [S.l.]: bookman, 2007. ISBN 978-856-003-149-8.

D'AURIOL, B. J. et al. Visualizations of human activities in sensor-enabled ubiquitous environments. In: *ICUMT*. IEEE, 2009. p. 1–6. <http://dblp.uni-trier.de/db/conf/icumt/icumt2009.htmldAuriolHLL09>.

DIGNUM, F. et al. Organizing web services to develop dynamic, flexible, distributed systems. In: KOTSIS, G. et al. (Ed.). *iiWAS*. ACM, 2009. p. 225–234. ISBN 978-1-60558-660-1. <http://dblp.uni-trier.de/db/conf/iiwas/iiwas2009.htmlDignumDPV09>.

DIGNUM, V. A model for organizational interaction: based on agents, founded in logic. *PhD Thesis, Utrecht University*, 2004.

DURKEE, D. Why cloud computing will never be free. *Commun. ACM*, v. 53, n. 5, p. 62–69, 2010. <http://dblp.uni-trier.de/db/journals/cacm/cacm53.htmlDurkee10>.

ELMROTH, E.; LARSSON, L. Interfaces for placement, migration, and monitoring of virtual machines in federated clouds. *Grid and Cooperative Computing, 2009. GCC '09. Eighth International Conference on*, p. 253 –260, aug. 2009.

FARINES, J.-M.; FRAGA, J. da S.; OLIVEIRA, R. S. de. *Sistemas de Tempo Real*. 12ª Escola de Computação, IME-USP, São Paulo-SP, 2000. <http://www.das.ufsc.br/ romulo/>.

FILHO, P. J. D. F. *Introdução à Modelagem e Simulação de Sistemas com Aplicações em Arena*. 2. ed. Florianópolis, SC, Brasil: Visual Books, 2008. ISBN 978-857-502-228-3.

FOSTER, I.; KESSELMAN, C.; TUECKE, S. The anatomy of the grid: Enabling scalable virtual organizations. *Int. J. High Perform. Comput. Appl.*, Sage Publications, Inc., Thousand Oaks, CA, USA, v. 15, n. 3, p. 200–222, 2001. ISSN 1094-3420.

FOSTER, I. et al. Cloud computing and grid computing 360-degree compared. *Grid Computing Environments Workshop, 2008. GCE '08*, p. 1 –10, nov. 2008.

GOOGLE, G. C. *Google data center power usage efficiency*. 2010. <http://www.google.com/corporate/datacenters/measuring.html>. Acessado em 30/8/2010.

GREENPEACE, T. N. G. I. *Make IT Green: Cloud computing and its contribution to climate change*. 2010. <http://www.greenpeace.org/>. Acessado em 30/02/2010.

GRUBER, C. G. Capex and opex in aggregation and core networks. In: *Optical Fiber Communication Conference*. Optical Society of America, 2009. p. 1–3. <http://www.opticsinfobase.org/abstract.cfm?URI=OFC-2009-OThQ1>.

HARRIS, J. *Green Computing and Green IT Best Practices on Regulations and Industry Initiatives, Virtualization, Power Management, Materials Recycling and Telecommuting*. London, UK, UK: Emereo Pty Ltd, 2008. ISBN 1921523441, 9781921523441.

HUEBSCHER, M. C.; MCCANN, J. A. A survey of autonomic computingdegrees, models, and applications. *ACM Comput. Surv.*, ACM, New York, NY, USA, v. 40, p. 7:1–7:28, August 2008. ISSN 0360-0300. <http://doi.acm.org/10.1145/1380584.1380585>.

HYPERVISOR, X. *Xen Hypervisor*. 2010. <http://xen.org/>. Acessado em 30/4/2010.

KIM, K. H.; BELOGLAZOV, A.; BUYYA, R. Power-aware provisioning of cloud resources for real-time services. In: *MGC '09: Proceedings of the 7th International Workshop on Middleware for Grids, Clouds and e-Science*. New York, NY, USA: ACM, 2009. p. 1–6. ISBN 978-1-60558-847-6.

KIRSCHNICK, J. et al. Toward an architecture for the automated provisioning of cloud services. *Communications Magazine, IEEE*, v. 48, n. 12, p. 124 –131, 2010. ISSN 0163-6804.

KURP, P. Green computing. *Commun. ACM*, v. 51, n. 10, p. 11–13, 2008. <http://dblp.uni-trier.de/db/journals/cacm/cacm51.htmlKurp08>.

LEONHARD, W.; MURRAY, K. *Green Home Computing For Dummies*. [S.l.]: For Dummies, 2009. ISBN 0470467452, 978-047-046-745-9.

LIN, G. et al. Cloud computing: It as a service. *IT Professional*, v. 11, n. 2, p. 10 –13, mar. 2009. ISSN 1520-9202.

LIU, L. et al. Greencloud: a new architecture for green data center. In: *ICAC-INDST '09: Proceedings of the 6th international conference industry session on Autonomic computing and communications industry session*. New York, NY, USA: ACM, 2009. p. 29–38. ISBN 978-1-60558-612-0.

LLORENTE, I. M. et al. A grid infrastructure for utility computing. In: *WETICE*. IEEE Computer Society, 2006. p. 163–168. ISBN 0-7695-2623-3. <http://dblp.uni-trier.de/db/conf/wetice/wetice2006.htmlLlorenteMHL06>.

MCLAUCHLAN, L.; MEHRUBEOGLU, M. A survey of green energy technology and policy. In: *Green Technologies Conference, 2010 IEEE*. [S.l.]: IEEE, 2010. p. 1 –6.

MOTTA, F. C. P. *Teoria das Organizações*. 2. ed. Brasil: Cengage Learning, 2001. ISBN 978-8-5221-0249-5.

MURUGESAN, S. Harnessing green it: Principles and practices. *IT Professional*, v. 10, n. 1, p. 24–33, 2008. <http://dblp.uni-trier.de/db/journals/itpro/itpro10.htmlMurugesan08>.

MURUGESAN, S. Making it green. *IT Professional*, v. 12, n. 2, p. 4 –5, 2010. ISSN 1520-9202.

NIST. *National Institute of Standards and Technology Draft Definition of Cloud Computing*. 2009. <http://csrc.nist.gov/groups/SNS/cloud-computing>. Acessado em 30/4/2009.

PEGDEN, C. D.; SADOWSKI, R. P.; SHANNON, R. E. *Introduction to Simulation Using SIMAN*. 2nd. ed. New York, NY, USA: McGraw-Hill, Inc., 1995. ISBN 0070493200.

PINHEIRO, E. et al. Load balancing and unbalancing for power and performance in cluster-based systems. In: *Proceedings of the Workshop on Compilers and Operating Systems for Low Power (COLP'01)*. [s.n.], 2001. p. 182–195. <http://research.ac.upc.es/pact01/colp/paper04.pdf>.

QUILLINAN, T. B. et al. Developing agent-based organizational models for crisis management. *Proceedings of the 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009), Decker, Sichman, Sierra and Castelfranchi (eds.)*, p. 45–52, Jan 2009.

SCHEDULER, X. C. *Xen Credit Scheduler*. 2010. <http://wiki.xensource.com/xenwiki/CreditScheduler>. Acessado em 30/3/2010.

SCHMIDT, M. et al. Efficient distribution of virtual machines for cloud computing. *Parallel, Distributed, and Network-Based Processing, Euromicro Conference on*, IEEE Computer Society, Los Alamitos, CA, USA, v. 0, p. 567–574, 2010. ISSN 1066-6192.

SHULZ, G. *The Green and Virtual Data Center*. New York: CRC Press - Taylor Francis Group, 2009. ISBN 978-1-4200-8666-9.

SIDDIQI, A. et al. Use of information and mobile computing technologies in healthcare facilities of saudi arabia. In: . [S.l.: s.n.], 2009. p. 289 –294.

SPEC, S. P. E. C. *Standard Performance Evaluation Corporation - Spec*. 2010. <http://www.spec.org/>. Acessado em 30/4/2010.

TANENBAUM, A. S. *Redes de Computadores*. trad. 4 ed. Rio de Janeiro: Elsevier, 2003.

TANENBAUM, A. S.; STEEN, M. v. *Distributed Systems: Principles and Paradigms (2nd Edition)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 2006. ISBN 0132392275.

UNHELKAR, B. Green it: The next five years. *IT Professional*, v. 13, n. 2, p. 56 –59, 2011. ISSN 1520-9202.

VÁZQUEZ-SALCEDA, J. et al. Combining organisational and coordination theory with model driven approaches to develop dynamic, flexible, distributed business systems. In: AKAN, O. et al. (Ed.). *Digital Business*. Springer Berlin Heidelberg, 2010, (Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, v. 21). p. 175–184. ISBN 978-3-642-11532-5. 10.1007/978-3-642-11532-5_20. <$http://dx.doi.org/10.1007/978-3-642-11532-5_20$>.

VOORSLUYS, W. et al. Cost of virtual machine live migration in clouds: A performance evaluation. In: *Proceedings of the 1st International Conference on Cloud Computing*. Berlin, Heidelberg: Springer-Verlag, 2009. (CloudCom '09), p. 254–265. ISBN 978-3-642-10664-4. <http://dx.doi.org/10.1007/978-3-642-10665-1_23>.

VOUK, M. Cloud computing x2014; issues, research and implementations. *Information Technology Interfaces, 2008. ITI 2008. 30th International Conference on*, p. 31 –40, jun. 2008.

VYTELINGUM, P. et al. Agent-based micro-storage management for the smart grid. In: *AAMAS*. [s.n.], 2010. p. 39–46. <http://dblp.uni-trier.de/db/conf/atal/aamas2010.htmlVytelingumVRRJ10>.

WOOD, T. et al. Sandpiper: Black-box and gray-box resource management for virtual machines. *Comput. Netw.*, Elsevier North-Holland, Inc., New York, NY, USA, v. 53, p. 2923–2938, December 2009. ISSN 1389-1286. <http://portal.acm.org/citation.cfm?id=1663647.1663710>.

YAU, S.; YANG, C.-C.; SHATZ, S. An approach to distributed computing system software design. *Software Engineering, IEEE Transactions on*, SE-7, n. 4, p. 427 – 436, 1981. ISSN 0098-5589.

APÊNDICE A – Publicações e Apresentações

A.1 ARTIGO PUBLICADO EM EVENTO INTERNACIONAL

Carlos Oberdan Rolim, Fernando Luis Koch, Carlos Becker Westphall, Jorge Werner, Armando Fracalossi, Giovanni Schmitt Salvador. *A Cloud Computing Solution for Patient's Data Collection in Health Care Institutions. The Second International Conference on eHealth, Telemedicine, and Social Medicine - eTELEMED* 2010. 10-16 de Fevereiro, 2010 - St. Maarten, Netherlands Antilles;

A.2 TUTORIAL APRESENTADO EM EVENTO INTERNACIONAL

Jorge Werner, Carlos Becker Westphall, Armando Fracalossi, Giovanni Schmitt Salvador. *Grid and Cloud Computing Management and Security. In: 8th International Information and Telecommunication Technologies Symposium*, 2009, Florianópolis - SC. *Proceedings of the 8th International Information and Telecommunication Technologies Symposium*. Florianópolis - SC : Fundação Barddal de Educação e Cultura, 2009. v. 1. p. 250-250.

ANEXO A – Código CloudSim

```
1  package Sensores;
2
3  import java.util.ArrayList;
4  import java.util.List;
5
6  import org.cloudbus.cloudsim.Host;
7  import org.cloudbus.cloudsim.Log;
8  import org.cloudbus.cloudsim.core.CloudSim;
9  import org.cloudbus.cloudsim.core.CloudSimTags;
10 import org.cloudbus.cloudsim.core.SimEntity;
11 import org.cloudbus.cloudsim.core.SimEvent;
12 import org.cloudbus.cloudsim.lists.VmList;
13
14 import Brokers.NewBroker;
15
16 import utilization.utilizationModelFunction;
17
18
19 public class SensorGlobal extends SimEntity {
20
21    private List<VmMonitor> vm_list;
22    private List<HostMonitor> host_list;
23
24    private static final int MONITOR = 1234567;
25    private static final int MONITOR_INTERVAL = 10;
26
27    private int broker;
28
29    private boolean stop_monitor = false;
30
31    public SensorGlobal(String name, List<VmMonitor> vms,
          List<HostMonitor> hosts) {
32       super(name);
33       // TODO Auto-generated constructor stub
34       vm_list = vms;
35       host_list = hosts;
36    }
37
38    @Override
39    public void processEvent(SimEvent ev) {
40       // Log.printLine("Evento recebido (" + CloudSim.clock
             () + "):" + ev.getTag());
41
42       switch(ev.getTag()){
43          case MONITOR:
44             // TODO arranjar algum meio para parar de ficar
                   monitorando quando a simulaÃ§Ã£o acabar.
45             /*if(CloudSim.getEntity(vm_list.get(0).getUserId
                   ()).numEventsWaiting() == 0){ // caso a
                   simulaÃ§Ã£o ainda esteja acontecendo
46                stop_monitor = true;
```

```
47              }*/
48
49              if(!stop_monitor){
50                 this.monitorar();
51                 sendNow(broker,NewBroker.IS_FINISHED);
52                 send(getId(), MONITOR_INTERVAL, MONITOR);
53                 break;
54              }
55
56
57           case CloudSimTags.END_OF_SIMULATION:
58              this.stop_monitor = true;
59              this.shutdownEntity();
60              break;
61
62           default:
63              Log.printLine(getName() + ": unknow event type");
64              break;
65        }
66
67     }
68
69     @Override
70     public void shutdownEntity() {
71        // TODO Auto-generated method stub
72
73     }
74
75     @Override
76     public void startEntity() {
77        // TODO Auto-generated method stub
78
79        Log.printLine(this.getName() + " is starting...");
80
81        for (VmMonitor v : vm_list){
82           v.setMonitor(true);
83        }
84        for (HostMonitor h : host_list){
85           h.setMonitor(true);
86        }
87
88        this.schedule(getId(), MONITOR_INTERVAL, MONITOR);
89
90        broker = vm_list.get(0).getUserId();
91     }
92
93     public void addHost(HostMonitor host){
94
95        host_list.add(host);
96
97     }
98
```

```
99    public void addVm(VmMonitor vm){
100
101      vm_list.add(vm);
102
103   }
104
105   public boolean deleteVm(VmMonitor vm){
106
107      if (vm_list.contains(vm)){
108
109         vm_list.remove(vm);
110         return true;
111
112      }else{
113         return false;
114      }
115
116   }
117
118   public void monitorar(){
119
120      List<VmMonitor> vms = new ArrayList<VmMonitor>();
121      List<HostMonitor> hosts = new ArrayList<HostMonitor>()
             ;
122
123      Log.printLine(CloudSim.clock() + ": Fazendo o
             monitoramento.");
124
125      double mediaCargaVms = 0;
126
127      for (VmMonitor v : vm_list){
128         if(v.monitor())
129            vms.add(v);
130         mediaCargaVms += v.getTotalUtilizationOfCpu(CloudSim
                .clock());
131         if(v.getTotalUtilizationOfCpu(CloudSim.clock()) >
                0.0 && v.isSpare())
132            sendNow("Broker_0", NewBroker.CREATE_VMS);
133      }
134      for (HostMonitor h : host_list){
135         if(h.monitor())
136            hosts.add(h);
137      }
138      if(vms.size() > 0 || hosts.size() > 0){
139         Log.printLine("Alguma VM ou PM ultrapassou o limiar"
                );
140      }
141
142      if((mediaCargaVms/(double)vm_list.size()) > 0.9){
143         // sendNow("Broker_0", NewBroker.CREATE_VMS);
144         sendNow("Broker_0", NewBroker.MEDIA_VM_ALTA);
145      }
```

```
146
147     }
148
149 }
```

Código Fonte A.1 – SensorGlobal.java

```
1  package experimentos.jorge;
2
3  import interfaces.Experimento;
4
5  import java.util.ArrayList;
6  import java.util.HashMap;
7  import java.util.List;
8  import java.util.Map;
9
10 import org.cloudbus.cloudsim.FederatedDatacenter;
11 import org.cloudbus.cloudsim.Host;
12 import org.cloudbus.cloudsim.Log;
13 import org.cloudbus.cloudsim.Vm;
14 import org.cloudbus.cloudsim.VmScheduler;
15 import org.cloudbus.cloudsim.core.CloudSim;
16 import org.cloudbus.cloudsim.core.CloudSimTags;
17
18 import Sensores.HostMonitor;
19 import Sensores.VmMonitor;
20 import configs.ConfigDatacenters;
21 import configs.ConfigHosts;
22
23 public class Experimento4 extends Experimento{
24
25    private List<FederatedDatacenter> dcs;
26
27    protected ConfigDatacenters configDCs;
28    protected ConfigHosts configPMs;
29
30    private int numDatacenters = 2;
31
32    private int numVms = 10;
33
34    private long storage = 1000000;
35    private int ram = 10240;
36    private int mips = 3000;
37
38    private double minThresholdPM = 0.2;
39    private double maxThresholdPM = 1.0;
40
41    private double minThresholdVM = 0.0;
42    private double maxThresholdVM = 0.9;
43
44    public Experimento4(){
```

```
45          configPMs= new ConfigHosts(storage, 4, mips, ram,
                10000, minThresholdPM, maxThresholdPM,
                minThresholdPM, maxThresholdPM, minThresholdPM,
                maxThresholdPM);
46          configDCs = new ConfigDatacenters(numDatacenters);
47          for(int i = 0; i < numDatacenters; i++){
48             configDCs.setHostsPorDatacenters(i, 100);
49          }
50          super.setIndex(4);
51       }
52
53       public ConfigDatacenters getConfigDatacenters() {
54          // TODO Auto-generated method stub
55          return configDCs;
56       }
57       public ConfigHosts getConfigHosts() {
58          // TODO Auto-generated method stub
59          return configPMs;
60       }
61
62       public double getMinThresholdVM() {
63          return minThresholdVM;
64       }
65
66       public double getMaxThresholdVM() {
67          return maxThresholdVM;
68       }
69
70       @Override
71       public void processaTratamentoPM(HostMonitor host){
72
73          List<Vm> lista_vm = host.getVmList();
74
75          // pegar a lista de ddatacenters e achar qual
                datacenter estÃ¡ o host, entÃ£o migrar as vms
                deste host para outro datacenter
76
77          FederatedDatacenter current = null;
78          for(FederatedDatacenter dc : dcs){
79             if(dc.getHostList().contains(host)){
80                current = dc;
81                break;
82             }
83          }
84
85          List<FederatedDatacenter> lista_dcs_temp = new
                ArrayList<FederatedDatacenter>();
86          lista_dcs_temp.addAll(dcs);
87          if(current != null)
88             lista_dcs_temp.remove(current);
89
90          // desalocar as vms e aloca-las em outro datacenter
```

```
91      for(Vm vm : lista_vm){
92
93         vm.setInMigration(true);
94         Map<String, Object> migracao = new HashMap<String,
               Object>();
95         migracao.put("vm", vm);
96
97         for(FederatedDatacenter dc : lista_dcs_temp){
98            Host h = null;
99            for(Host host_temp : dc.getHostList()){
100              if(host_temp.getAvailableMips() >= vm.getMips
                    ()){
101                 h = host_temp;
102              }
103           }
104           migracao.put("host", h); // Aloca para o primeiro
                 host do datacenter
105           dc.scheduleNow(dc.getId(), CloudSimTags.
                 VM_MIGRATE, migracao);
106        }
107
108     }
109
110  }
111
112  @Override
113  public void processaTratamentoVM(VmMonitor vm){
114
115
116     List<Double> mips_list = new ArrayList<Double>();
117     mips_list.add(0, vm.getCurrentRequestedTotalMips() *
            1.1);
118     HostMonitor host = (HostMonitor) vm.getHost();
119     VmScheduler vmScheduler = host.getVmScheduler();
120     vmScheduler.deallocatePesForVm(vm);
121     vmScheduler.allocatePesForVm(vm, mips_list);
122
123     Log.printLine(CloudSim.clock() + ": Capacidade da VM #
            " + vm.getId() +" alterada.");
124
125  }
126
127  @Override
128  public void setDatacenters(List<FederatedDatacenter>
        lista){
129     dcs = lista;
130  }
131
132  @Override
133  public int getNumVms() {
134     return numVms;
135  }
```

```
136
137     @Override
138     public boolean VmToHost(VmMonitor vm) {
139        // TODO Auto-generated method stub
140        return false;
141     }
142
143     @Override
144     public void verificaHost(VmMonitor vm) {
145        // TODO Auto-generated method stub
146        return;
147
148     }
149
150 }
```

Código Fonte A.2 – Experimento4.java