

Revista da Graduação

Vol. 5

No. 2

2012

14

Seção: Faculdade de Informática

Título: EDUCLOUD 2: Implementando mecanismos de elasticidade em uma nuvem privada para ambientes acadêmicos

Autor: Giuseppe Alves Lopes e Jair de Mello Junior

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL
FACULDADE DE INFORMÁTICA
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

GIUSEPPE ALVES LOPES
JAIR DE MELLO JUNIOR

EDUCLOUD 2
Implementando mecanismos de elasticidade em uma nuvem privada
para ambientes acadêmicos

Porto Alegre

2012

GIUSEPPE LOPES

JAIR JUNIOR

**Implementando mecanismos de elasticidade em uma nuvem privada
para ambientes acadêmicos**

Trabalho de conclusão de curso de graduação
apresentado à Faculdade de Informática da
Pontifícia Universidade Católica do Rio Grande
do Sul, como requisito parcial para obtenção do
grau de Bacharel em Sistemas de Informação.

Orientador: Prof. Dr. Tiago Ferreto

Porto Alegre

2012

RESUMO

A computação em nuvem vem se consolidando cada vez mais diante do mundo tecnológico. Novos conceitos como escalabilidade e elasticidade de processos, serviços e infraestrutura, estão se tornando referência no mercado. Este trabalho apresenta uma proposta de extensão da ferramenta Educloud para suportar operações de elasticidade. O Educloud é uma ferramenta para criação de um ambiente de computação em nuvem usando recursos acadêmicos. O trabalho propõe a inclusão de funcionalidades como: *scale-up* e *scale-out*, além da inclusão de uma unidade de armazenamento centralizado. Foram realizadas diversas pesquisas sobre as principais características e tecnologias existentes na computação em nuvem para suportar funções de elasticidade. Estas modificações na ferramenta possibilitarão uma vivência prática das principais características de elasticidade em um ambiente acadêmico.

ABSTRACT

Cloud computing has been increasingly consolidating itself in the technological world. New concepts such as processes scalability and elasticity, services and infrastructure, are becoming market references. This paper presents an extension proposal to the Educloud tool, to support elastic operations. Educloud is a tool for creating a cloud computing environment using academic resources. The paper proposes the addition of features such as: *scale-up* and *scale-out*, and also the inclusion of a centralized storage unit. Several researches have been performed in order to evaluate the main features and technologies existing in cloud computing to support elasticity functions. These modifications on the tool provide a practical experience of the main features of elasticity in an academic environment.

LISTA DE ILUSTRAÇÕES

FIGURA 1: MODELOS DE SERVIÇOS. [8]	13
FIGURA 2: TIPOS DE COMPUTAÇÃO DE NUVEM. [15].....	16
FIGURA 3: CICLO DE VIDA DE UMA MÁQUINA VIRTUAL NO OPENNEBULA. [23].....	18
FIGURA 4: ARQUITETURA DO EUCALYPTUS. [14].....	20
FIGURA 5: ARQUITETURA DO OPENSTACK. [25].....	22
FIGURA 6: CONSOLIDAÇÃO DE SERVIDORES. [4].....	24
FIGURA 7: RELAÇÃO ENTRE MÁQUINAS VIRTUAIS E O VMM. [1]	26
FIGURA 8: VIRTUALIZAÇÃO TOTAL. [1]	27
FIGURA 9: PARA-VIRTUALIZAÇÃO. [1].....	28
FIGURA 10: SISTEMA DE ARMAZENAMENTO CENTRALIZADO. [28]	31
FIGURA 11: ARQUITETURA MODULAR DO VIRTUALBOX. [18].....	33
FIGURA 12: ARQUITETURA DO EDUCLOUD 2.	40
FIGURA 13: DIAGRAMA DE SEQUÊNCIA PARA CRIAÇÃO DE UMA MÁQUINA VIRTUAL.	44
FIGURA 14: DIAGRAMA DE SEQUÊNCIA PARA INICIAR UMA MÁQUINA VIRTUAL.	46
FIGURA 15: DIAGRAMA DE SEQUENCIA PARA FUNÇÃO SCALE-UP.	49
FIGURA 16: VIRTUAL SERVER VIA ROTEAMENTO DIRETO. [37].....	51
FIGURA 17: DIAGRAMA DE SEQUENCIA PARA FUNÇÃO SCALE-OUT.	52
FIGURA 18: DIAGRAMA DE CLASSES DA API DE CONTROLE.	55
FIGURA 19: TELA PARA AUTENTICAÇÃO DE USUÁRIOS.	56
FIGURA 20: TELA PARA VISUALIZAÇÃO DOS CONTROLADORES DE NODO.	57
FIGURA 21: TELA PARA VISUALIZAÇÃO E CADASTRO DE TEMPLATES.	57
FIGURA 22: TELA PARA GERENCIAMENTO DE MÁQUINAS VIRTUAIS.....	58

FIGURA 23: TELA PARA REALIZAÇÃO DE SCALE-UP.....	58
FIGURA 24: RESULTADOS DO TESTE PARA CADASTRO DE TEMPLATES.....	61
FIGURA 25: RESULTADOS DO TESTE PARA CRIAÇÃO DE VM.....	62
FIGURA 26: RESULTADOS DO TESTE PARA INICIALIZÇÃO DE VM.....	63
FIGURA 27: RESULTADOS DO TESTE PARA CRIAÇÃO DE VM PARA SCALE-OUT.....	65
FIGURA 28: RESULTADOS DO TESTE PARA INICIALIZAÇÃO DE VM PARA SCALE-OUT.....	66
FIGURA 29: RESULTADOS DO TESTE PARA FUNÇÃO DE SCALE-OUT.....	67

LISTA DE TABELAS

TABELA 1: RESULTADOS DE VALIDAÇÃO EDUCLOUD 2.....	67
---	----

SUMÁRIO

1 INTRODUÇÃO	9
2 FUNDAMENTAÇÃO TEÓRICA.....	11
2.1 COMPUTAÇÃO EM NUVEM	11
2.1.1 Características	12
2.1.1.1 Elasticidade.....	12
2.1.1.2 Pay-as-you-go	12
2.1.1.3 Amplo acesso.....	13
2.1.2 Modelos de Serviços	13
2.1.2.1 Infraestrutura como Serviço (IaaS – Infrastructure as a Service)	14
2.1.2.2 Plataforma como Serviço (PaaS - Platform as a Service)	14
2.1.2.3 Software como Serviço (SaaS – Software as a Service)	14
2.1.3 Tipos de Computação em Nuvem	15
2.1.3.1 Nuvem Publica	15
2.1.3.2 Nuvem Privada.....	15
2.1.3.3 Nuvem Híbrida.....	15
2.1.4 Exemplos de nuvens computacionais.....	16
2.1.4.1 Amazon Elastic Cloud Computing (Amazon EC2).....	16
2.1.4.2 OpenNebula.....	17
2.1.4.3 Eucalyptus	18
2.1.4.4 OpenStack	21
2.1.4.5 Educloud	22
2.2 VIRTUALIZAÇÃO.....	24
2.2.1 Conceitos	24
2.2.1.1 Monitor de Máquinas Virtuais (VMM) ou Hypervisor.....	25
2.2.2 Tipos de Virtualização	26
2.2.2.1 Virtualização Total.....	26
2.2.2.2 Para-virtualização	28
2.2.3 Funcionalidades Comuns	28
2.2.3.1 Escalabilidade.....	29
2.2.3.2 Live Migration.....	29
2.2.3.3 Sistema de Armazenamento Centralizado	30
2.2.4 Exemplos de Sistemas de Virtualização	31
2.2.4.1 VMware	32
2.2.4.2 Xen	32

2.2.4.3 <i>VirtualBox</i>	33
2.3 ELASTICIDADE NOS AMBIENTES DE COMPUTAÇÃO EM NUVEM	34
2.3.1 Scale-up e Scale-out	34
2.3.2 Armazenamento Centralizado	35
2.3.3 Comparativo Educloud	35
3 EDUCLOUD 2	37
3.1 DESCRIÇÃO GERAL	37
3.2 OBJETIVOS ESPECÍFICOS	37
4 MODELAGEM E IMPLEMENTAÇÃO	39
4.1 ARQUITETURA DO EDUCLOUD 2	39
4.2 DESCRIÇÃO DAS FUNCIONALIDADES	41
4.2.1 Armazenamento Centralizado	41
4.2.1.1 <i>Cadastro de Templates</i>	42
4.2.2 Gerenciamento de Máquinas Virtuais	43
4.2.2.1 <i>Criação e Remoção</i>	44
4.2.2.2 <i>Inicialização e Parada</i>	45
4.2.3 Scale-up	47
4.2.4 Scale-out	49
4.2.4.1 <i>Linux Virtual Server (LVS)</i>	50
4.2.4.2 <i>LVS x Educloud 2</i>	52
4.2.5 API de Controle	53
4.2.6 Interface de Usuário	56
5 VALIDAÇÃO	60
6 CONCLUSÃO E TRABALHOS FUTUROS	68
REFERÊNCIAS BIBLIOGRÁFICAS.....	71

1 INTRODUÇÃO

A computação em nuvem vem se tornando uma tecnologia largamente utilizada. Cada vez mais as empresas e instituições migram suas aplicações para o ambiente de nuvem. Desta forma, essa tecnologia passa a ser classificada como essencial.

Caracterizar a computação em nuvem não é uma tarefa fácil, pois o tema envolve diversas tecnologias, tais como: sistemas de computação distribuída, *clusters*, sistemas de armazenamento centralizado ou distribuído em pares e virtualização. Geralmente é possível encontrar um *datacenter* com diversos servidores mais robustos, que com uma camada de virtualização, nos permite criar diversas máquinas virtuais, dando ao usuário final a ideia de recursos infinitos. Novos servidores físicos são agregados de forma transparente, tornando-se disponíveis para criação de novas máquinas virtuais logo que instalados.

A tecnologia de computação em nuvem oferece diversos benefícios a seus usuários, oferecendo muitas vezes funcionalidades que jamais seriam possíveis com a utilização de servidores físicos. A possibilidade de utilizar um *hardware* escalável, permite expandir ou encolher suas configurações de acordo com sua demanda e em tempo real. A maioria dos fornecedores utiliza a metodologia *pay-per-use*, onde é cobrado somente o tempo que foi utilizado de cada configuração. Outro ponto que é importante destacar é a flexibilidade de criação de máquinas, discos, redes, balanceadores de carga, além de possuírem uma vasta biblioteca de modelos de máquinas pré-instaladas.

A virtualização tem uma parcela muito grande e importante no cenário da computação em nuvem. Com ela que é possível desenvolver todo o conceito de criação e distribuição de máquinas virtuais existentes. Ela está praticamente intrínseca na arquitetura de computação em nuvem, pois é o núcleo desta tecnologia.

A elasticidade, ou escalabilidade, além de colaborar para caracterizar a computação em nuvem, passa a se tornar um diferencial muito grande para a utilização dessa tecnologia. A configuração dinâmica, ou *scale-up*, de máquinas virtuais é um ponto importante no ambiente de computação em nuvem, pois ele que permite que as mesmas sejam redimensionadas de forma dinâmica, de acordo com

as demandas do ambiente, passando a ser um dos principais pontos da computação em nuvem. Ainda tratando-se de elasticidade, também é possível trabalhar com o conceito de *scale-out*, com a utilização de um balanceador de carga, podemos ter mais de uma máquina respondendo por um mesmo serviço, permitindo que um ambiente possa ser facilmente ampliado horizontalmente.

O Educloud [17] é uma plataforma de Computação em Nuvem focada no ambiente acadêmico. Ele permite montar uma arquitetura de computação em nuvem de forma rápida e sem dependência de *hardware* específico. Ele implementa as tarefas mais comuns de um ambiente de computação em nuvem, permitindo gerenciar todo o ciclo de vida de uma máquina virtual, além de disponibilizar uma API de controle, que facilita muito a sua integração com outras aplicações. Desta forma, o principal objetivo do Educloud é oferecer a possibilidade de estudar uma plataforma de computação em nuvem em um ambiente acadêmico, sem a necessidade de uma grande infraestrutura para isto. Contudo, o Educloud ainda não possui nenhuma característica de elasticidade em sua plataforma, o que atualmente é apresentado na maioria dos fornecedores de computação em nuvem.

Este trabalho apresenta o processo de desenvolvimento de novas funcionalidades ao sistema Educloud, agregando a este as características de elasticidade necessárias para um ambiente de computação em nuvem.

O trabalho está dividido em 6 capítulos. O Capítulo 2 apresenta o referencial teórico, onde são apresentados os principais conceitos existentes em torno de computação em nuvem, virtualização, modelos de serviço e um estudo aprofundado dentre as principais ferramentas existentes no mercado. O Capítulo 3 apresenta o Educloud 2, descrevendo de forma geral a aplicação e seus principais objetivos. O Capítulo 4 apresenta o processo de modelagem e implementação realizado durante o desenvolvimento do Educloud 2, apresentando detalhes das novas funcionalidades agregadas e também a sua nova arquitetura. O Capítulo 5 apresenta o processo de validação das novas funcionalidades do sistema, além de realizar um comparativo entre a versão anterior e a atual. Por fim, o Capítulo 6 apresenta a conclusão deste trabalho e propostas de trabalhos futuros para o ambiente Educloud.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta os principais conceitos envolvidos entre virtualização e computação em nuvem. São expostas suas principais funcionalidades e características, bem como um estudo entre as principais ferramentas existentes no mercado.

2.1 COMPUTAÇÃO EM NUVEM

Segundo Taurion, o termo computação em nuvem surgiu em 2006 em uma palestra de Eric Schmidt, da Google, sobre como sua empresa gerenciava seus *datacenters* [7]. Hoje, computação em nuvem, se apresenta como uma tendência, tornando mais dinâmico o mundo da tecnologia.

A nuvem é uma representação para a Internet ou infraestrutura de comunicação entre os componentes arquiteturais, baseada em uma abstração que oculta à complexidade de infraestrutura. Cada parte desta infraestrutura é provida como um serviço e estes serviços são normalmente alocados *em datacenters* [8].

Computação em nuvem é, em essência, um sistema em grande escala de computação distribuída. Ele veio com o intuito de promover um novo modelo de negócios para a indústria de tecnologia da informação, transformando a utilização de recursos computacionais em serviços.

O modelo de computação em nuvem foi desenvolvido com o objetivo de fornecer serviços de fácil acesso e de baixo custo e garantir características tais como disponibilidade e escalabilidade. Este modelo visa fornecer, basicamente, três benefícios. O primeiro benefício é reduzir o custo na aquisição e composição de toda infraestrutura requerida para atender as necessidades das empresas, podendo essa infraestrutura ser composta sob demanda e com recursos heterogêneos e de menor custo. O segundo é a flexibilidade que esse modelo oferece no que diz respeito à adição e troca de recursos computacionais, podendo assim, escalar tanto em nível de recursos de hardware quanto software para atender as necessidades das empresas e usuários. O terceiro benefício é prover uma abstração e facilidade de acesso aos usuários destes serviços. Com isso, os usuários dos serviços não

precisam conhecer aspectos de localização física e de entrega dos resultados destes serviços [8].

Computação em nuvem é, portanto, uma maneira bastante eficiente de maximizar e flexibilizar os recursos computacionais. Além disso, uma nuvem computacional é um ambiente redundante e resiliente por natureza. Resiliente pode ser definido como a capacidade de um sistema de informação continuar a funcionar corretamente, apesar do mau funcionamento de um ou mais dos seus componentes [7].

2.1.1 Características

Os sistemas de computação em nuvem existentes possuem diversas características em comum. Elas são: elasticidade, *pay-as-you-go* e amplo acesso.

2.1.1.1 Elasticidade

A elasticidade refere-se a capacidade de aumentar a quantidade de recursos alocados de forma rápida, em alguns casos automaticamente, caso haja a necessidade de escalar com o aumento da demanda, e liberados, na retração dessa demanda. Para os usuários, os recursos disponíveis para uso parecem ser ilimitados e podem ser adquiridos em qualquer quantidade e a qualquer momento. [9]

A virtualização tem um papel fundamental na elasticidade, possibilitando instanciar de forma rápida réplicas de máquinas virtuais.

2.1.1.2 *Pay-as-you-go*

Baseando-se nos serviços essenciais de utilidade pública como, água, eletricidade e telefone, os serviços de computação em nuvem adotam, geralmente, o mesmo modelo de cobrança, onde o usuário paga somente o que consome, podendo a qualquer momento usar mais ou menos recursos e pagar somente com base neste uso. A computação em nuvem proporciona serviços de tecnologia da Informação sob demanda com pagamento baseado no seu uso.

O usuário pode adquirir unilateralmente recursos computacionais, como tempo de processamento no servidor ou armazenamento na rede, na medida em que lhe for necessário e principalmente sem depender de um contato em cada provedor de serviços [9].

2.1.1.3 Amplo acesso

Uma das grandes vantagens e característica da computação em nuvem é sua grande possibilidade de utilização e integração com vários sistemas. Através da internet ou redes próprias é possível ter acesso as *Application Programming Interface* e interfaces disponibilizadas pelos principais fornecedores, tornando ágil o acesso à infraestrutura ou topologia da mesma. Usuários com pouco conhecimento conseguem criar servidores com imagens de sistemas operacionais, alocar discos e configurar sua rede forma rápida e ágil.

2.1.2 Modelos de Serviços

A computação em nuvem distribui os recursos na forma de serviços. Com isso, é possível dividir a computação em nuvem em três modelos, em relação aos serviços oferecidos, conforme a ilustração apresentada na Figura 1.

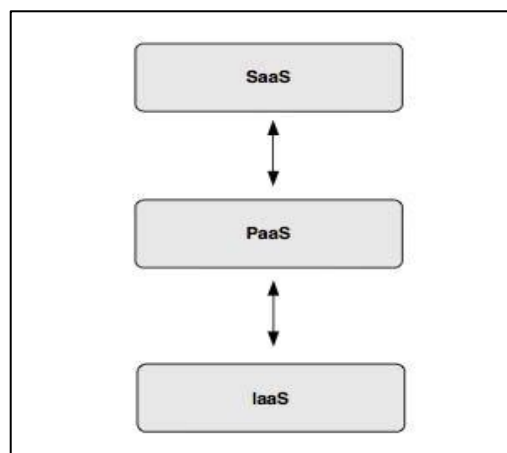


Figura 1: Modelos de Serviços. [8]

2.1.2.1 Infraestrutura como Serviço (IaaS – *Infrastructure as a Service*)

O principal objetivo do IaaS é tornar mais fácil e acessível o fornecimento de recursos, tais como servidores, rede, armazenamento e outros recursos de computação fundamentais para construir um ambiente sob demanda, que podem incluir sistemas operacionais e aplicativos. A IaaS possui algumas características, tais como uma interface única para administração da infraestrutura, API para interação com *hosts*, *switches*, balanceadores de carga, roteadores e o suporte para a adição de novos equipamentos de forma simples e transparente [8].

A IaaS se refere a uma infraestrutura computacional baseada em técnicas de virtualização de recursos de computação [8]. Esta infraestrutura pode escalar dinamicamente mais recursos assim como liberar a utilização destes recursos de acordo com a necessidade das aplicações que hospeda. O Amazon Elastic Cloud Computing (EC2) [15] e o Eucalyptus [14] são exemplos de IaaS.

2.1.2.2 Plataforma como Serviço (PaaS - *Platform as a Service*)

A PaaS oferece uma infraestrutura de alto nível de integração para implementar e testar aplicações na nuvem [8]. O usuário somente tem controle sobre as aplicações implantadas e, possivelmente, as configurações das aplicações hospedadas nesta infraestrutura. A PaaS fornece um sistema operacional, linguagens de programação e ambiente de desenvolvimento para as aplicações, auxiliando a implementação de sistemas de software, já que contém ferramentas de desenvolvimento e colaboração entre desenvolvedores [8].

2.1.2.3 Software como Serviço (SaaS – *Software as a Service*)

Os modelos de SaaS dispõem de sistemas de software com propósitos específicos que estão disponíveis para os usuários através da Internet. Os sistemas de software são acessíveis normalmente a partir de, um navegador Web. No SaaS, o usuário não administra ou controla a infraestrutura subjacente, como a rede, servidores, sistemas operacionais, armazenamento ou mesmo as características individuais da aplicação.

2.1.3 Tipos de Computação em Nuvem

Nesta seção serão apresentados os tipos de computação em nuvem, tornando mais claro o entendimento quanto ao nível de acesso de cada tipo.

2.1.3.1 Nuvem Publica

Nuvens públicas fornecem acesso a recursos de computação para o público em geral através da Internet. O fornecedor de nuvem pública permite que os clientes criem recursos normalmente através de uma interface de serviço web. Clientes alocam os recursos conforme necessário em um modelo *pay-as-you-go*. Nuvens públicas oferecem acesso a grandes quantidades de recursos escaláveis de forma temporária, sem a necessidade de investimento de capital na estrutura de *datacenter*.

2.1.3.2 Nuvem Privada

Nuvens privadas provem aos usuários acesso imediato aos recursos de computação hospedados na infraestrutura de uma organização. Funciona exatamente como a nuvem pública, com a ressalva de que os recursos são provisionados na infraestrutura da organização, sendo assim, impactado pelas regras de firewall, segurança e de negócios providas pela organização.

2.1.3.3 Nuvem Híbrida

Uma nuvem híbrida combina recursos de computação extraídos de uma ou mais nuvens públicas e uma ou mais nuvens privadas. Na Figura 2, é possível ter uma visão dos três tipos de nuvem e sua relação. A nuvem privada, protegido sobre as regras de firewall e segurança da empresa, a nuvem publica sobre a política da fornecedora do serviço e a nuvem híbrida como uma camada intermediária entre as nuvens, podendo interagir entre as máquinas virtuais de ambas as nuvens.

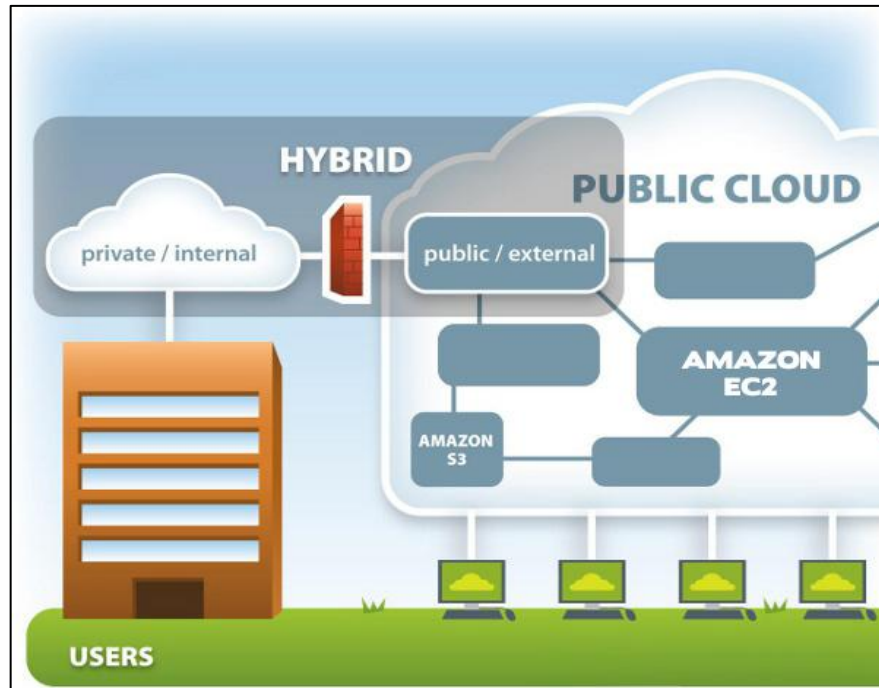


Figura 2: Tipos de Computação de Nuvem. [15]

2.1.4 Exemplos de nuvens computacionais

Nesta seção será apresentado um breve descritivo das principais soluções de computação em nuvem da atualidade. São apresentadas as suas características e recursos principais.

2.1.4.1 Amazon Elastic Cloud Computing (Amazon EC2)

Amazon Elastic Compute Cloud (Amazon EC2) é um serviço web que oferece capacidade computacional redimensionável na nuvem. Ele é projetado para tornar escalável a computação, tornando mais fácil para os desenvolvedores [15].

A interface da Amazon EC2 é um serviço web simples que permite obter e configurar capacidade computacional com um mínimo de atrito. Fornece o controle completo de seus recursos de computação e permite rodar este no ambiente computacional da Amazon [15]. O modelo utilizado é o *pay-per-use*, onde você paga exatamente o que consome. A Amazon ainda fornece uma série de ferramentas para desenvolvedores.

O Amazon EC2 implementa a virtualização de armazenamento de forma completa. É possível criar um volume de armazenamento e “atachá-lo” a um servidor rodando na nuvem.

Outro recurso incorporado no Amazon EC2 é o *AutoScaling*, onde a instância tem pré-configurações que permitem que ela realize a adição ou remoção automática de recursos para atender novas demandas [15].

2.1.4.2 OpenNebula

A infraestrutura virtual disponibilizada pelo OpenNebula provê as funcionalidades necessárias para instalar, monitorar e controlar máquinas virtuais em um ambiente de recursos fisicamente distribuídos. Ela foi desenhada para aceitar a integração com diferentes *hypervisors* e configurações de infraestrutura.

O OpenNebula é composto por três componentes principais. O OpenNebula Core é o componente central que controla o ciclo de vida das máquinas virtuais, realizando operações básicas (instalação, monitoração, migração ou remoção). O Core também provê controles básicos e uma interface de monitoração para servidores físicos. O *Capacity Manager* é um módulo instalável que controla a funcionalidade pelo Core. O *Capacity Manager* ajusta a instalação de máquinas virtuais baseado em uma série de políticas pré-definidas. O *Default Capacity Scheduler* implementa uma regra de pareamento para criação e suporta ações de restrições definidas pelo usuário. A fim de prover uma abstração da camada de virtualização, o OpenNebula utiliza *Virtualizer Access Drivers* plugáveis que expõem a funcionalidade básica do *hypervisor* (instalação, monitoração ou *shutdown* de uma máquina virtual). Assim, o OpenNebula fica restrito a nenhum ambiente, provendo um controle uniforme das camadas independente da tecnologia de virtualização utilizada. [23]

Desta forma, o OpenNebula cria uma infraestrutura física para suportar a execução de um determinado serviço de trabalho. Além disso, o OpenNebula é capaz de dimensionar dinamicamente a infraestrutura de interface com uma nuvem externa. Um *driver* de virtualização da Amazon EC2 está incluído no OpenNebula, e os *drivers* podem ser desenvolvidos para a interface com outras nuvens. Esta

integração de nuvem externa com recursos internos permite o acesso efetivo à capacidade computacional de terceiros.

O OpenNebula suporta em sua arquitetura de forma completa todo ciclo de vida de uma máquina virtual. A Figura 3 ilustra o ciclo de vida de uma máquina virtual na plataforma do OpenNebula.

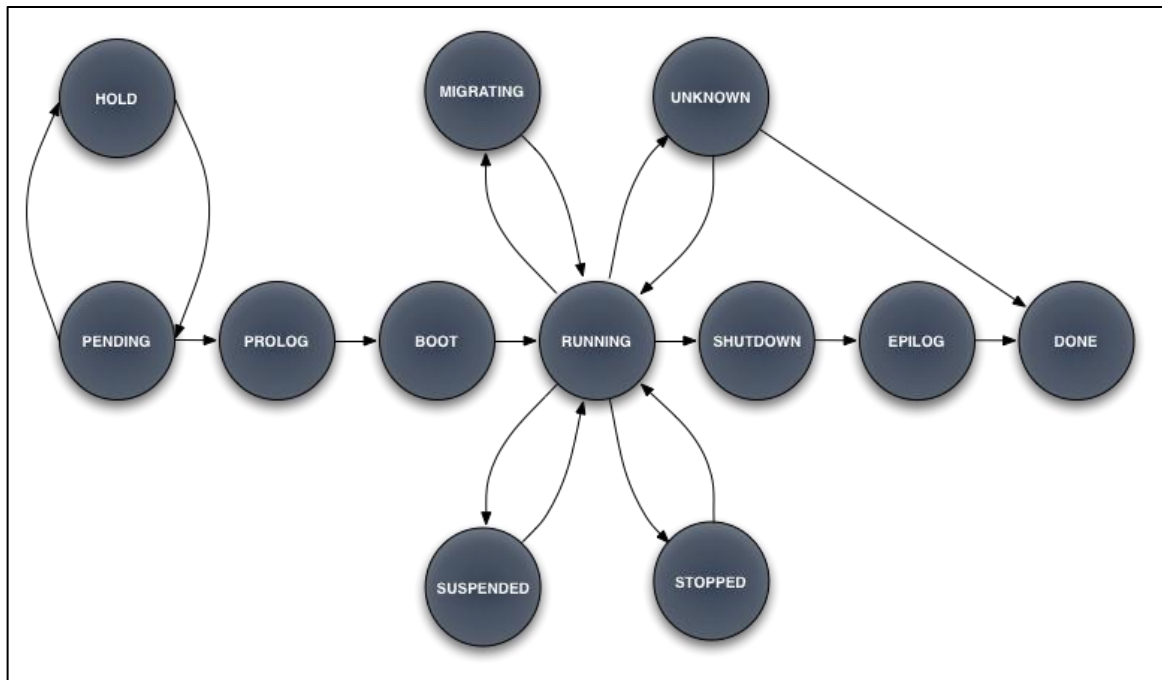


Figura 3: Ciclo de vida de uma máquina virtual no OpenNebula. [23]

Fora do comum somente os estados de Prolog e Epilog, o primeiro é quando transfere-se os arquivos de imagem da VM e de disco para o *host*, e o segundo é quando efetua-se a limpeza desses arquivos do *host*, antes de concluir o processo de *shutdown*. Como analisado na Figura 3, existe um estado de “*migrating*”. O OpenNebula suporta a migração de máquinas virtuais de um *host* para outro, usando uma técnica chamada *live migration*, permitindo que possa ser realizada essa migração com o máquina virtual em execução, diminuindo o *downtime* da mesma no processo de migração de um *host* para outro. [23]

2.1.4.3 Eucalyptus

O Eucalyptus permite a criação de nuvens privadas, sem requisitos de infraestrutura existente ou necessidade de introduzir *hardware* especializado. Ele implementa um IaaS, que é acessível através de uma API. A interface é compatível

com a EC2 e S3 da Amazon. Esta compatibilidade permite que qualquer nuvem Eucalyptus possa ser transformada em uma nuvem híbrida, capaz de gerar recursos de computação de nuvem pública. O Eucalyptus também é compatível com uma série de ferramentas e aplicativos que seguem o padrão da EC2 e S3 da Amazon. [14]

Um dos pontos fortes do Eucalyptus é sua arquitetura baseada em camadas de responsabilidade, fazendo com que os componentes da nuvem privada se acoplem de forma transparente.

Os principais componentes dessa arquitetura são:

Cloud Controller (CLC): O CLC é o ponto de entrada da nuvem para administradores, desenvolvedores, gerentes de projeto e usuários finais. Ele é responsável por consultar os *Node Controllers* para obter informações sobre recursos, tomada de decisões de alto nível de programação, e implementá-las, fazendo pedidos para os *Cluster Controllers (CC)*. O CLC é também a interface para a plataforma de gestão. Em essência, o CLC é responsável por expor e gerenciar os recursos subjacentes virtualizados (servidores, rede e armazenamento).

Cluster Controller (CC): O CC geralmente é executado em um cluster de máquina *front-end*, ou qualquer outra máquina que tenha conectividade de rede para ambas as máquinas, o *Node Controller (NC)* e para a máquina executando o CLC. CC's reúnem informações sobre um conjunto de máquinas virtuais (VM) e agendam a execução de VM em nós específicos. O CC também gerencia a instância de rede virtual. Todos os nós associados a um único CC devem estar no mesmo domínio de rede (*Ethernet*).

Storage Controller (SC): O SC fornece funcionalidade semelhante à *Amazon Elastic Block Storage* e é capaz de fazer interface com vários sistemas de armazenamento (NFS, iSCSI, etc.). O armazenamento em bloco elástico é um dispositivo de bloco que pode ser conectado a um sistema de arquivos de uma instância.

Node Controller (NC): O NC é executado em cada máquina que é designada para hospedar VM's. O NC controla as atividades das VM's, incluindo a execução, verificação e suspensão de instâncias de VM's. Também busca e limpa cópias locais de imagens de instâncias (o *kernel*, o sistema de arquivos raiz, e a imagem *ramdisk*),

e ele ainda consulta e controla o software (sistema operacional hospedeiro e o *hypervisor*) em resposta a consultas e solicitações de controle do CC. O NC também é responsável pela gestão do terminal de rede virtual. A Figura 4 ilustra os componentes e seus relacionamentos. [14]

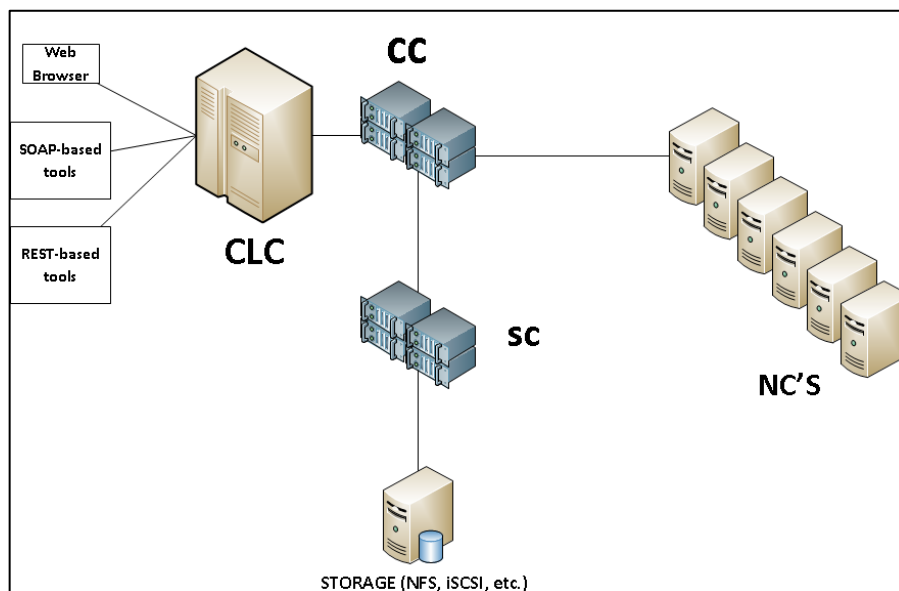


Figura 4: Arquitetura do Eucalyptus. [14]

O Eucalyptus contempla de forma completa as operações básicas de utilização de instâncias de máquinas virtuais, contudo, algumas funcionalidades ele ainda não suporta.

Embora o Eucalyptus não tenha nativo uma implementação de *scale-out* para garantir a clonagem de instâncias de máquinas virtuais, ele suporta a utilização de balanceadores de carga em seu *core*.

Quanto a *scale-up* e *scale-down* no Eucalyptus, de forma nativa ele não possui componentes para que os níveis mais acima possam realizar essas operações de elasticidade, contudo, por ser uma solução de código aberto, e ter uma arquitetura modular e com suas camadas bem definidas, alguns desenvolvedores já estenderam suas características, assim, suportando *scale-up* e *scale-down*.

O Eucalyptus não implementa padrões de elasticidade que poderiam torná-lo completo. Suas deficiências são equilibradas com sua integração com o serviço da Amazon, que contempla de forma completa essas questões de elasticidade. [14]

2.1.4.4 OpenStack

O *OpenStack* é uma colaboração global de desenvolvedores e especialistas em computação em nuvem, para o desenvolvimento de uma plataforma, de código aberto, de computação em nuvem para nuvens públicas e privadas. O projeto foi fundado em meados de 2010 pela *Nasa* em parceria com a *Rackspace*. Ele visa oferecer soluções para todos os tipos de nuvens por ser simples de implementar, altamente escalável e rico em recursos. A tecnologia consiste em uma série de projetos inter-relacionados que entregam vários componentes para uma solução de infraestrutura de nuvem pública ou privada. [26] [27]

O *OpenStack* é composto por três componentes principais, são eles:

OpenStack Compute: é um controlador de nuvem, usado para inicializar instâncias virtuais para um usuário ou um grupo. Também é usado para configurar a rede para cada instância ou projeto que contenha várias instâncias de um projeto particular.

OpenStack Object Storage: é um sistema para armazenar objetos em um sistema de grande capacidade, altamente escalável e redundante. O *Object Storage* tem uma variedade de aplicações, tais como backup ou arquivamento de dados, servidor de gráficos ou vídeos (*streaming* de dados para o navegador do usuário), armazenamento secundário ou terciário de dados estáticos, desenvolvendo novas aplicações com integração de dados de armazenamento.

OpenStack Image Service: é um sistema de pesquisa e recuperação de imagens de máquinas virtuais. Ele pode ser configurado de três maneiras: (i) usando *OpenStack ObjectStore* para armazenar imagens, (ii) utilizando solução da *Amazon Simple Storage (S3)* de armazenamento diretamente, ou (iii) através de armazenamento S3, com armazenamento de objetos como o intermediário para o acesso S3. Na Figura 5 é possível ter uma visão geral dos componentes do *OpenStack*. [25]

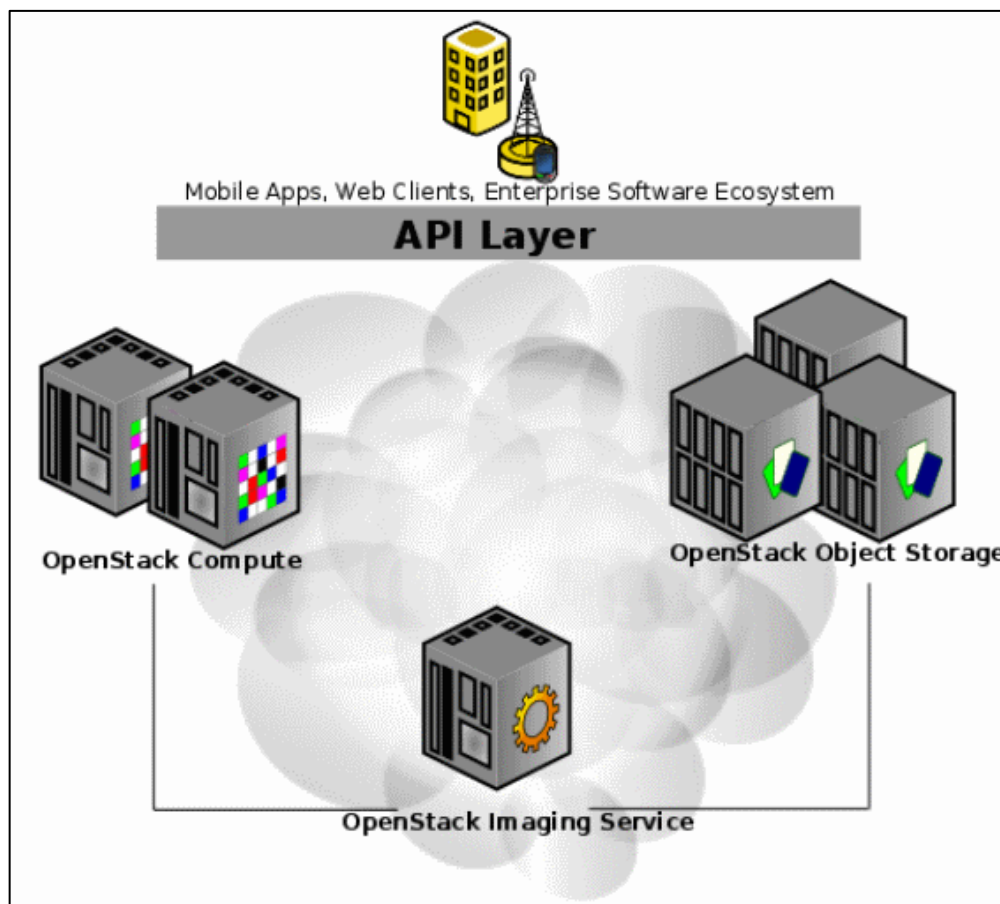


Figura 5: Arquitetura do OpenStack. [25]

O OpenStack suporta um grande número de *hypervisors*, como: Hyper-V, KVM, LXC, VMWare e Xen, entre outros.

Esta plataforma nos permite realizar as operações de *live migration*, contudo, como a mesma possui um leque grande de *hypervisors* e nem todos suportam esse tipo de operação nativamente, Em sua versão atual (1.4.3) só é possível realizar *live migration* se o host e as VM's rodarem o Ubuntu como sistema operacional, além de ser necessário um *armazenamento* centralizado.

Uma outra característica interessante do OpenStack é a possibilidade de realizar *scale-up* e *scale-down on demand*, possibilitando expandir e recolher recursos de uma instância de forma transparente. [25]

2.1.4.5 Educloud

O Educloud é uma solução de código aberto para nuvens privadas com foco principal no ambiente educacional [17]. Ele foi desenvolvido durante um trabalho de

conclusão da Pontifícia Universidade Católica do Rio Grande do Sul. Ele foi projetado para ser uma ferramenta de fácil utilização, para que qualquer pessoa possa implementar sua infraestrutura, sem a necessidade de *hardware* dedicado. A principal motivação para seu desenvolvimento foi possibilitar a vivência da computação em nuvem em um ambiente acadêmico, de forma ágil e fácil, utilizando-se de *desktops* ou até mesmo de um *laptops* para isso. Ele utiliza um modelo de serviço IaaS e suporta o VirtualBox como sistema de virtualização.

Um dos principais objetivos do Educloud é tornar dinâmica a criação de máquinas virtuais, mesmo em ambientes compartilhados, ou seja, mesmo que um recurso, uma máquina, esteja em uso por um usuário, ainda assim é possível criar uma máquina virtual neste recurso.

A arquitetura do Educloud consiste basicamente em quatro componentes: Controlador de Nuvem, Controlador de Nodo, API de Controle e Interface do Usuário.

O Controlador de Nuvem é responsável por administrar e controlar os nodos e suas instâncias, os usuários da nuvem, expor a interface remota, além das funções e rotinas de controle.

O Controlador de Nodo executa as tarefas locais nos hosts. Ele recebe as instruções do controlador de nuvem e as executa no host. Podendo haver várias instâncias do controlador de nodo no Educloud, ele intermedia as mensagens entre o controlador de nuvem e o VirtualBox [18].

A API de Controle é um componente Java que abstrai o acesso às funcionalidades e informações expostas pelas interfaces do Controlador da Nuvem. Através da API de Controle é possível que desenvolvedores integrem outras aplicações Java às funcionalidades existentes no Educloud [17].

A interface de usuário consiste em uma aplicação web que permite a administração das máquinas virtuais através do navegador web. Este componente torna fácil e ágil o acesso às informações da nuvem.

Como visto anteriormente, as principais ferramentas do mercado como Amazon EC2, Eucalyptus, OpenNebula e OpenStack, implementam uma série de funcionalidades que os tornam de fato estruturas elásticas. O Educloud atualmente

não possui essas características e propriedades e esta proposta visa justamente evoluir o Educloud para que contemple essas funcionalidades.

2.2 VIRTUALIZAÇÃO

O conceito de virtualização está a cada dia se tornando mais popular dentre as áreas de tecnologia, oferecendo um ambiente de gerenciamento simplificado através da consolidação de servidores. Segundo Carissimi [1], a maioria das infraestruturas de rede trabalham com a filosofia de “um servidor por serviço”, dedicando um servidor exclusivo para cada aplicação. A virtualização surge como solução para essa filosofia, onde através de um único *host*, torna-se possível virtualizar diversos servidores a fim de utilizar o máximo do recurso disponível.

2.2.1 Conceitos

Em um ambiente de produção, constantemente os desenvolvedores precisam testar novos processos, sem afetar os usuários do sistema. Uma prática comum é manter um servidor menor exclusivo para execução de testes e desenvolvimentos. A virtualização permite criar os dois ambientes em cima da mesma máquina, tornando-os completamente isolados uns dos outros [4]. Na Figura 6 estão representadas duas máquinas virtuais usando o mesmo hardware. A máquina virtual 1, responsável pela aplicação principal, e a máquina virtual 2, sendo uma suíte de testes com uma cópia idêntica da aplicação executada na máquina virtual 1.

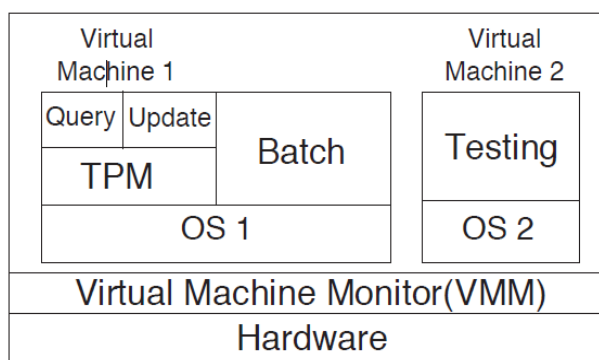


Figura 6: Consolidação de Servidores. [4]

A prática de consolidação de servidores vem se tornando comum em *datacenters*, devido ao alto impacto na redução de custo para manter a

infraestrutura física como espaço, energia elétrica, cabeamento, refrigeração, suporte e manutenção a vários sistemas [1].

O conceito de máquina virtual surgiu como uma forma de *time-sharing* utilizada nos antigos *mainframes*. Normalmente uma empresa era capaz de manter apenas um *mainframe*, devido ao seu alto custo. Com isso, a mesma máquina era utilizada para o desenvolvimento de novas aplicações e também para a implantação de novas aplicações, o que tornava o ambiente muito instável, devido as constantes interrupções durante os processos de implantação [2].

Uma máquina virtual pode ser conceituada como um emulador de um ambiente computacional completo, a qual será executada sobre um *host* físico, permitindo a execução de diversas instâncias do mesmo *hardware*. Segundo Rose, Ela é como uma cópia totalmente protegida e isolada do *hardware* subjacente da máquina física [2].

Conceitualmente, um computador pode operar de dois modos: modo usuário e modo supervisor. No modo usuário normalmente são executadas todas as aplicações, permitindo somente a execução de instruções não-privilegiadas, que são aquelas que não permitem o acesso direto ao conjunto de instruções da CPU. Já o modo supervisor, permite um controle total sobre o *hardware*, tornando possível executar qualquer tipo de instrução requisitada ao processador. Os sistemas operacionais trabalham nesse modo, porém através de um *bit* de controle, é possível alterar o modo de execução antes de entregar o controle da CPU para aplicação do usuário [3] [4].

É possível implementar máquinas virtuais de duas formas: Como uma aplicação de sistema operacional (Máquina Virtual de Processo), ou como uma camada de *software* posicionada entre o *hardware* da máquina e o sistema operacional (*Virtual Machine Monitor*) [1].

2.2.1.1 Monitor de Máquinas Virtuais (VMM) ou *Hypervisor*

Ao virtualizar um sistema, é criada uma camada de abstração do *hardware* chamada de *Virtual Machine Monitor* (VMM) ou *Hypervisor*. Essa camada cria uma

interface de comunicação entre o *hardware* físico e as máquinas virtuais, permitindo o acesso a recursos compartilhados [4].

Um VMM tem a função de hospedar máquinas virtuais, facilitando seu gerenciamento [2]. Através do VMM é realizada a virtualização dos recursos compartilhados pelas máquinas virtuais, como, processadores, dispositivos de entrada e saída, memória física e de armazenamento [1]. Desta forma, o VMM é responsável por gerenciar o *hardware* físico do recurso, trabalhando em modo supervisor, enquanto as máquinas virtuais trabalham em modo usuário [3]. Segundo Ferrazani [3], quando uma máquina virtual tenta executar uma instrução privilegiada, é gerada um interrupção que será tratada pelo VMM, que será o responsável por tratar e executar essa instrução. A Figura 7 apresenta um ambiente virtualizado a partir de um VMM, demonstrando sua relação com as máquinas virtuais.

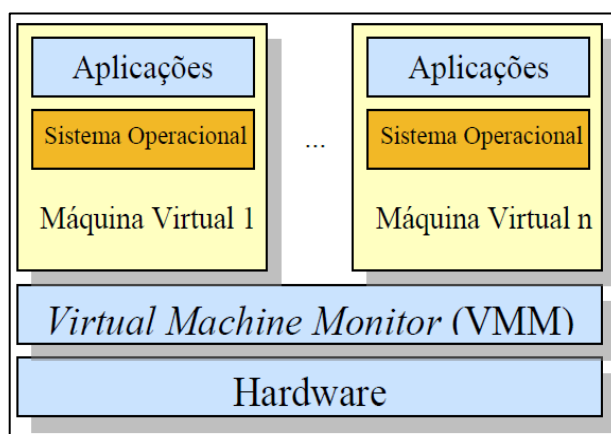


Figura 7: Relação entre máquinas virtuais e o VMM. [1]

2.2.2 Tipos de Virtualização

Os conceitos de virtualização apresentados anteriormente podem ser aplicados em um ambiente computacional de muitas formas. Nesta seção serão apresentados os principais tipos ou técnicas de virtualização. Sendo esses Virtualização Total e Para-virtualização.

2.2.2.1 Virtualização Total

O modo de virtualização total cria uma instância virtual do *hardware* físico existente, permitindo que sistema operacional e aplicações possam ser executados

sem perceber essa virtualização. Essa forma de implementação permite a execução de um sistema operacional padrão, ou seja, não é necessário realizar nenhuma alteração no mesmo para permitir a virtualização [1]. Na Figura 8 é apresentada a estrutura utilizada pela virtualização total, utilizando sistemas operacionais originais.

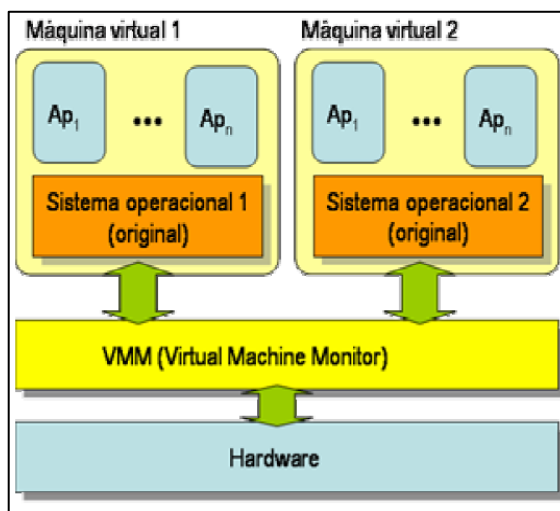


Figura 8: Virtualização total. [1]

O modo de virtualização total vem acompanhado de alguns problemas conhecidos [1]:

- É muito complexo para uma máquina virtual simular o comportamento exato de um dispositivo, devido a imensa diversidade de produtos existentes. Com isso, surge a necessidade de criar um conjunto de dispositivos genéricos, causando muitas vezes uma subutilização do *hardware*.
- Como o sistema operacional da máquina virtual não é modificado, todas as instruções solicitadas devem ser testadas pela VMM antes de liberar o acesso, elevando assim, o custo de processamento.
- O VMM precisa contornar alguns problemas técnicos quanto a implementação de alguns sistemas operacionais. Em sistemas convencionais, a memória virtual é implantada através de paginação, e existe um processo muito complexo em torno de gerência de alocação, liberação e controle de acesso às páginas. Teoricamente isso não seria um problema, mas esse tratamento também representa um aumento de processamento.

2.2.2.2 Para-virtualização

A para-virtualização é uma solução para os problemas conhecidos da virtualização total. A principal característica desse modelo é a necessidade de alterar o sistema operacional, permitindo que, ao executar qualquer instrução que possa alterar o estado do sistema, uma chamada direta a VMM seja realizada. Essa mudança representa um grande ganho de desempenho, uma vez que não será necessário testar todas as instruções solicitadas. A para-virtualização soluciona o problema de acesso aos dispositivos de *hardware*, permitindo o acesso via *drivers* através da própria máquina virtual, eximindo a necessidade da utilização de *drivers* genéricos, subutilizando a capacidade do dispositivo [3]. Na Figura 9 é apresentada a estrutura utilizada pela para-virtualização, utilizando sistemas operacionais modificados de acordo com o VMM.

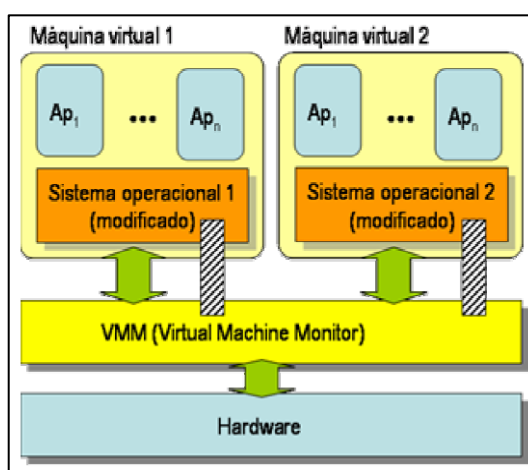


Figura 9: Para-virtualização. [1]

Embora a para-virtualização apresentasse um ganho de desempenho significativo frente à virtualização total, essa disparidade tem sido superada devido à presença de instruções de virtualização nos processadores Intel e AMD, que favorecem a virtualização total [3].

2.2.3 Funcionalidades Comuns

O processo de virtualização envolve diversas funcionalidades ou características comuns a todos os aplicativos de gerenciamento de máquinas

virtuais. Abaixo serão apresentadas as principais funcionalidades apontadas neste trabalho.

2.2.3.1 Escalabilidade

Antes da existência das técnicas de virtualização, uma simples atualização de memória poderia se tornar uma tarefa muito custosa, deixando diversos serviços inacessíveis por muito tempo. Operação essa que só poderia ser realizada perante a instalação de novos recursos na máquina física, ou até mesmo migrar esta aplicação para outra máquina que tivesse uma capacidade de desempenho superior [11].

A virtualização permite a realocação dos recursos de forma dinâmica (*scale-up* ou *vertical-scaling*), permitindo dimensionar uma máquina virtual de acordo com a demanda dos processos existentes, ou seja, é possível aumentar a quantidade de memória, quantidade de núcleos de processamento e armazenamento sem a necessidade de desligar a máquina. Dessa forma, uma máquina virtual pode sofrer um *scale-up* para atender uma demanda solicitada por uma aplicação, e ao seu término será possível retorná-la ao seu estado original. Caso a máquina virtual não possa ser redimensionada devido a limitações de capacidade da máquina física, a mesma poderá ainda, ser facilmente migrada para outra máquina física que possua uma capacidade de processamento superior.

Outra técnica conhecida para aumentar a capacidade é o *scale-out* ou *horizontal-scaling*. Ao invés de redimensionar uma máquina virtual, a mesma passa a ser replicada através de um processo de clonagem, gerando diversas instâncias da mesma máquina virtual, e aumentando assim sua capacidade de processamento. Para a utilização de *scale-out*, é necessária a utilização de um sistema de balanceamento de carga, que pode ser definido como um divisor da carga de trabalho dentre as diversas instâncias virtuais.

2.2.3.2 Live Migration

Migrar máquinas virtuais consiste em transferir uma máquina virtual de um servidor físico para outro, de forma que os processos agregados à mesma não sofram nenhum tipo de impacto durante o procedimento de migração. Técnicas de

migração são constantemente utilizadas para o gerenciamento de balanceamento de carga entre duas ou mais máquinas, além de garantir o funcionamento de servidores de alta disponibilidade, possibilitando transferir uma máquina virtual para outro servidor físico, em caso de alguma falha de *hardware* [5].

O ponto chave para tornar o processo de migração possível, é que uma máquina virtual trabalha com um *hardware* também virtual. Segundo Alkmim e Uchôa [5], o encapsulamento de todo o estado de *hardware* e *software* da máquina virtual em alguns arquivos cria um ambiente propício para a migração. Desta forma, é possível migrar uma máquina virtual para uma máquina física diferente, mesmo que essa possua um *hardware* diferente.

Todo o processo de migração deve ocorrer de forma transparente para todas as aplicações da máquina virtual, bem como para todos os clientes que estejam conectados a ela. A única mudança que poderá ser percebida será uma rápida desaceleração, acompanhada de um ganho considerável de desempenho, considerando uma migração para uma máquina física de desempenho superior [6].

Ao iniciar o processo de migração é necessário garantir que os seguintes estados permaneçam os mesmos após o procedimento: (i) estado dos dispositivos virtuais, os (ii) estados das conexões de rede externas e o (iii) estado da memória física. Ao migrar uma máquina virtual é necessário realizar uma cópia de toda sua memória da origem para o destino. Devido ao grande volume de dados, essa cópia precisa ser iniciada ainda com a máquina virtual rodando, a fim de minimizar o tempo com a máquina parada, para copiar os dados bloqueados [5,6].

2.2.3.3 Sistema de Armazenamento Centralizado

Com o constante crescimento do volume de dados das empresas, a presença de uma estrutura consistente para armazenamento de dados se tornou indispensável. Os sistemas de armazenamento (*storages*) surgiram como uma solução para o compartilhamento de dados, trazendo segurança, velocidade e consistência aos mesmos. A ideia de centralizar os dados em um local compartilhado entre todos os servidores oferece benefícios como: velocidade de acesso aos dados, facilidade no gerenciamento, redundância em caso de falha de algum disco, etc.

A flexibilidade é um dos requisitos básicos para um ambiente virtualizado, a utilização de um sistema de armazenamento centralizado se torna indispensável para tornar essa característica possível. Manter os discos de uma máquina virtual em uma unidade de armazenamento centralizado permite que esta possa ser facilmente transportada para outro host em caso de falha e ausência de recursos computacionais dentre os hardwares físicos do ambiente. Dessa forma, o seu disco não estaria encapsulado junto do arquivo que carrega o estado do hardware virtual da VM, sendo este apenas mapeado durante a sua inicialização. Na Figura 10 é apresentado um ambiente utilizando um sistema de armazenamento centralizado, onde todas as máquinas virtuais e máquinas físicas utilizam os espaços de armazenamento disponibilizados remotamente.

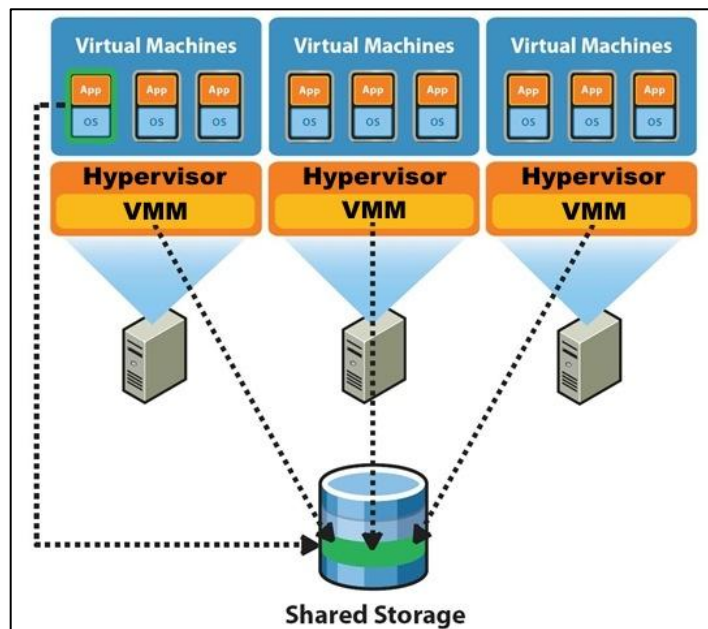


Figura 10: Sistema de Armazenamento Centralizado. [28]

2.2.4 Exemplos de Sistemas de Virtualização

Nesta seção serão apresentados alguns exemplos de ferramentas utilizadas para soluções de virtualização. As ferramentas descritas são: VMware, Xen e VirtualBox.

2.2.4.1 VMware

O VMWare é a ferramenta de virtualização mais popular para arquiteturas x86 [2]. Essa ferramenta pode ser definida como uma infraestrutura de virtualização completa, possuindo produtos destinados de *desktops* a *datacenters*. Os produtos VMware estão divididos em três categorias: gestão de automatização, infraestrutura virtual e virtualização de plataformas.

A categoria de gestão de automatização tem como objetivo facilitar o gerenciamento de toda infraestrutura virtual, de forma centralizada e automatizada. Nessa categoria são disponibilizados recursos como: monitoração do sistema, auxílio na conversão de sistemas físicos para virtuais, recuperação de desastres, etc. [1]. Os produtos de infraestrutura virtual tem foco em processos entre as máquinas virtuais, ofertando funcionalidades para atender requisitos de alta-disponibilidade, *backup* e migração de máquinas virtuais. Já a categoria de virtualização de plataformas, tem por objetivo a geração de máquinas virtuais, criando as abstrações de *hardware* citadas anteriormente. Essa categoria possui diversas versões de produtos, cada uma destinada a um ambiente específico [1, 3].

2.2.4.2 Xen

O Xen é uma ferramenta de código aberto que, diferentemente do VMware, teve no seu início a utilização do modelo de para-virtualização, baseando-se na arquitetura x86 [1]. Para implementar a para-virtualização o Xen trabalha com dois conceitos básicos: os domínios e o *hypervisor*. As diversas máquinas virtuais são os chamados domínios, e podem ser classificadas como privilegiadas (domínio 0) e não-privilegiadas (domínio U). O *hypervisor* é o responsável pela comunicação dentre os domínios e o *hardware* físico, gerenciando as requisições de acordo com o tipo de domínio [3].

A partir do Xen 3.0, uma versão destinada a virtualização total passou a ser oferecida, permitindo a execução de sistemas operacionais não modificados. A para-virtualização sempre apresentou um desempenho muito superior do que a virtualização total, porém, com as novas tecnologias de processadores que

oferecem suporte de *hardware* para virtualização, a virtualização total passou a apresentar resultados similares a para-virtualização.

2.2.4.3 VirtualBox

O VirtualBox é um software de virtualização total que possui parte do seu código fonte aberto. Ele é disponibilizado tanto para uso profissional quanto para uso doméstico [18]. O seu diferencial é o suporte a múltiplas plataformas, tais como os sistemas 32 e 64 bits com processadores Intel e AMD. Atualmente o VirtualBox possui versões disponíveis para sistemas Windows, Linux, Macintosh e OpenSolaris [18]. Ele foi criado inicialmente como um software proprietário. Somente em 2007 teve sua versão Open Source lançada, já na versão 2.0. Desenvolvido inicialmente pela empresa Innotek, que foi adquirida pela Sun Microsystems no início de 2008. Atualmente encontra-se em sua versão 4.1.16.

As definições de configuração de máquinas virtuais são armazenadas em XML e são totalmente independentes das máquinas locais. Por isso, as definições podem ser facilmente transferidas para outros computadores.

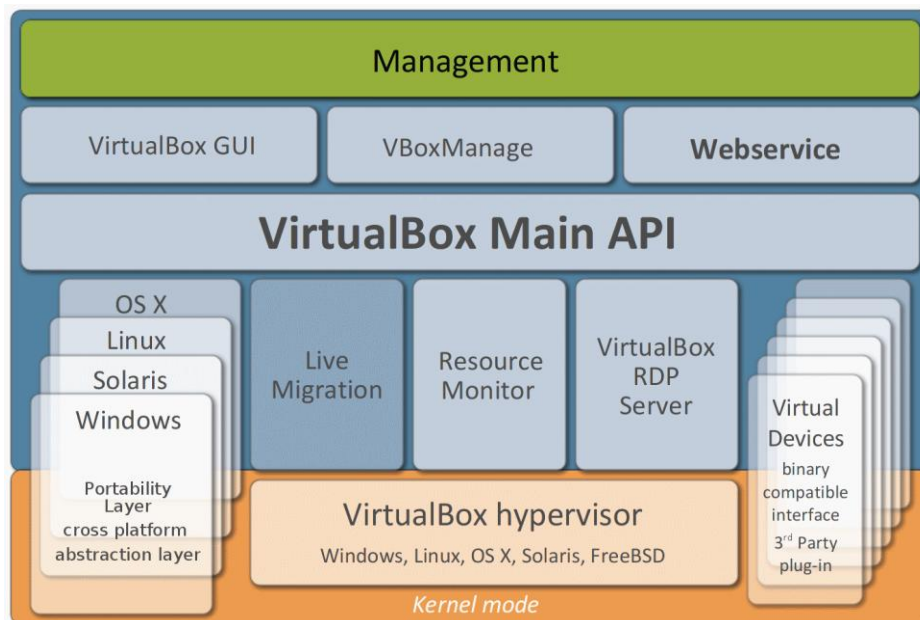


Figura 11: Arquitetura modular do VirtualBox. [18]

O VirtualBox possui interfaces de programação interna bem definidas e um desenho cliente/servidor. Isso torna fácil o controle de várias interfaces de uma só vez. A Figura 11 apresenta uma das grandes vantagens do VirtualBox [18], sua

arquitetura modular, permitindo que se consiga interagir com as demais camadas através dos componentes.

A Figura 11 divide basicamente em duas áreas sua arquitetura, a área laranja representa o código que é executado no modo *Kernel*, a área azul corresponde ao código que fica a disposição do usuário, para novas implementações e extensões.

A Figura 11 ainda mostra os principais meios de acesso a funcionalidades de gerenciamento e alteração das instâncias criadas, que pode ser através da interface GUI do VirtualBox [18] ou com o VboxManage, interface de linha de comando, ou ainda através do *webservice* disponibilizado.

As três interfaces permitem a maioria das funcionalidades, sendo as duas últimas citadas as mais completas para manutenção e gerenciamento das máquinas virtuais.

2.3 ELASTICIDADE NOS AMBIENTES DE COMPUTAÇÃO EM NUVEM

Nesta seção será apresentada uma análise comparativa entre as principais vantagens das ferramentas estudadas e o sistema Educloud, apontando os benefícios que podem ser atingidos ao adicionar funcionalidade de elasticidade no Educloud.

2.3.1 Scale-up e Scale-out

O uso de virtualização permite uma maior flexibilidade na gerência dos recursos em nuvens computacionais. Facilitando a criação e clonagem de máquinas virtuais; permitindo a alteração de configurações de forma dinâmica e durante a sua execução; permitindo migrar máquinas virtuais entre máquinas físicas diferentes sem a necessidade da interrupção do serviço, além de permitir a inclusão de novos recursos de acordo com a demanda apresentada, como armazenamento por exemplo.

A capacidade de redimensionar uma máquina virtual é uma função definida como básica em todas as ferramentas analisadas. Porém não basta apenas ser possível redimensioná-la, as alterações nas configurações precisam acontecer de forma dinâmica e sem afetar o estado atual da máquina virtual. As técnicas de *scale-*

up ou *vertical scaling*, são umas das principais características que uma aplicação elástica pode oferecer.

Outra característica encontrada dentre as ferramentas analisadas, é a *horizontal scaling* ou *scale-out*, que é a capacidade de replicação das máquinas virtuais, gerando novas instâncias de si mesmas, ou simplesmente "clonagem". Quando ocorre a necessidade de um aumento de processamento de uma máquina virtual, a mesma pode ser clonada diversas vezes, gerando cópias idênticas que passam a trabalhar em conjunto, porém transparecendo a existência de uma única máquina através do uso de um balanceador de carga que distribui as requisições entre as cópias de máquinas virtuais.

2.3.2 Armazenamento Centralizado

Para criar um ambiente elástico e com alto nível de flexibilidade, é indispensável à criação de uma unidade de armazenamento compartilhada entre todas as máquinas virtuais. Esta unidade de armazenamento é responsável por manter os discos de todas as máquinas virtuais da nuvem, oferecendo um alto nível de segurança aos dados e principalmente oferecendo desempenho para todo o ambiente da nuvem. Com a utilização de uma unidade de armazenamento centralizado, as máquinas virtuais passam a utilizar discos mapeados remotamente. Com esta prática, processos como a criação e inicialização de máquinas virtuais na nuvem, são diretamente afetados, pois os discos não necessitam mais ser transportados junto com a VM para o host que será responsável por sua execução.

Outro ponto interessante da utilização de uma unidade de armazenamento centralizado são os benefícios oferecidos para execução de outras funcionalidades de elasticidade como o *scale-out*. Quando da necessidade de clonar uma máquina virtual, discos de origem e destino estarão no mesmo local tornando este processo simples e ágil.

2.3.3 Comparativo Educloud

O Educloud é um sistema de nuvem privada, desenvolvido para ambientes acadêmicos. Atualmente suas funcionalidades contemplam todo o processo de

gerenciamento do ciclo de vida de uma máquina virtual em uma nuvem. Contudo, com a constante evolução da tecnologia de computação em nuvem, apenas gerenciar o ciclo de vida de uma máquina virtual não é o suficiente para prover uma ferramenta de aprendizado eficaz. O Educloud não oferece suporte a funcionalidades que provêm elasticidade a um ambiente de nuvem, das quais são requisitos básicos dentre as principais ferramentas disponíveis no mercado.

Para transformar o Educloud em uma ferramenta eficaz para o aprendizado e vivência da computação em nuvem, é necessária a disponibilização de funcionalidades como *scale-up*, permitindo redimensionar uma máquina virtual em plena execução, *scale-out*, permitindo ampliar o poder de processamento de uma aplicação de acordo com a demanda do ambiente, e principalmente que em sua arquitetura seja prevista a utilização de uma unidade de armazenamento centralizado. Uma das vantagens do uso de armazenamento centralizado é simplificar a gerência dos discos das VMs através da centralização em uma única máquina, além de permitir o uso de recursos de virtualização como *live-migration*.

3 EDUCLOUD 2

Neste capítulo é apresentada a definição do sistema Educloud 2, desenvolvido neste trabalho. É apresentada uma descrição geral do sistema e os principais aspectos motivacionais que conduziram o projeto, além de apresentar os objetivos utilizados durante a implementação desta nova versão da ferramenta.

3.1 DESCRIÇÃO GERAL

Com o passar dos anos, o até então conceito de computação em nuvem, vem se tornando a cada dia mais presente em diversos cenários. Em ambientes corporativos ou em ambientes acadêmicos, a demanda pela economia de recursos físicos, espaço e energia elétrica, cada vez mais se tornam prioridade.

Dentre os diversos benefícios oferecidos pelos ambientes na nuvem, as características de elasticidade são peças chave em qualquer ferramenta destinada à computação em nuvem. A elasticidade oferece conforto a seus usuários uma vez que todos os seus recursos computacionais são controlados pela nuvem, de acordo com sua demanda de processamento. Quando da necessidade de alto desempenho para execução de alguma tarefa, basta solicitar um *upgrade* à nuvem, e assim que não for mais necessário, retorna-se ao seu estado original.

Desta forma, a ferramenta Educloud 2 foi desenvolvida com o propósito de agregar algumas características de elasticidade, hoje ausentes na versão atual da ferramenta. Com as características de elasticidade necessárias, será possível manter uma gerência completa dos recursos computacionais de cada máquina virtual existente na nuvem privada.

3.2 OBJETIVOS ESPECÍFICOS

Como mencionado anteriormente, o presente trabalho teve como objetivo implementar melhorias na ferramenta Educloud. Atualmente, as principais aplicações baseadas em nuvens privadas existentes no mercado oferecem inúmeros recursos que facilitam o gerenciamento das máquinas virtuais da nuvem, visando principalmente agregar elasticidade ao ambiente.

A versão atual do Educloud possui recursos muito limitados quanto à facilidade do gerenciamento de seus recursos, oferecendo apenas funções básicas de gerenciamento como: “criar”, “ligar”, “desligar” e “remover” uma máquina virtual. Ao agregar novas funcionalidades ao sistema Educloud, torna-se possível gerenciar o funcionamento das máquinas virtuais de forma dinâmica, de acordo com a demanda de trabalho de cada usuário.

Através da função de *scale-up*, é possível redimensionar os principais recursos de cada máquina virtual, sem a necessidade de reiniciar a mesma. A possibilidade de realocação da quantidade de memória física e quantidade de processadores em plena execução é um dos maiores benefícios desta versão. Outra facilidade oferecida pela funcionalidade de *scale-up* é o gerenciamento do CAP dos processadores alocados para a máquina virtual. A função CAP possibilita gerenciar o percentual de utilização dos processadores alocados para uma máquina virtual, permitindo realizar uma subutilização deste recurso quando necessário. Em uma máquina virtual que possui dois processadores alocados, e um CAP de 50% configurado, significa que os processadores estarão trabalhando somente com metade de sua capacidade total.

Pensando diretamente no desempenho da ferramenta, e também como um ponto chave para o desenvolvimento de suas demais funcionalidades, foi adicionado o conceito de armazenamento centralizado ao Educloud. Uma unidade de armazenamento centralizado permite que os discos de uma máquina virtual sejam disponibilizados através de um *storage* local, onde cada máquina virtual possuirá apenas uma referência a esse disco compartilhado pelo *storage*. Essa funcionalidade, além de aumentar o nível de segurança dos dados da aplicação, facilita muito a operação do usuário, eximindo a necessidade de movimentar os dados de cada máquina virtual entre servidor e nodos a cada necessidade de execução da mesma.

A escalabilidade de processos também passa a ser disponibilizada no Educloud 2 através da função de *scale-out*. Esta funcionalidade é responsável por duplicar (clonar) uma instância de máquina virtual, criando um cluster de processamento para uma determinada aplicação. Como pré-requisito para implementação desta funcionalidade, surge à necessidade da utilização de uma ferramenta para realizar o balanceamento de carga entre as instâncias clonadas.

4 MODELAGEM E IMPLEMENTAÇÃO

Neste capítulo é apresentado o processo de modelagem e implementação referente a nova versão do sistema Educloud. São descritos de forma detalhada cada uma de suas novas funcionalidades e as demais alterações necessárias.

4.1 ARQUITETURA DO EDUCLOUD 2

A arquitetura do Educloud 2 mantém basicamente a mesma estrutura apresentada em sua versão anterior. Sua estrutura está dividida em quatro componentes principais: Controlador de Nuvem, Controlado de Nodo, API de Controle e Interface do Usuário.

O Controlador de Nuvem é responsável pelo gerenciamento de todas as atividades da nuvem privada. Seu papel principal é prover o controle, através da Interface de Usuário ou da API de Controle, entre as máquinas virtuais alocadas na nuvem e os Controladores de Nodo. [17]

O Controlador de Nodo prove o completo gerenciamento de um *host* físico (Nodo), presente na nuvem do Educloud. É ele o responsável pela execução das tarefas locais em cada host, através da integração com a ferramenta VirtualBox [18]. Ao permitir a alocação de diversos Controladores de Nodo na mesma nuvem, estes são gerenciados e mantidos através do Controlador de Nuvem. [17]

A Interface de usuário é o componente responsável pela interação com o usuário do Educloud 2 através de um navegador de internet. Ela prove acesso a todas as funcionalidades presentes atualmente no sistema, permitindo inclusão, alteração e remoção de tarefas na nuvem. [17]

Por fim, a API de Controle oferece um repositório de acesso a todas as funcionalidades oferecidas pelo Controlador de Nuvem. Por ser um componente Java, oferece uma grande facilidade para integrar novas funcionalidades ao sistema, onde isto foi de vital importância durante o processo de desenvolvimento do Educloud 2. [17]

A Figura 12 apresenta uma definição geral da arquitetura do Educloud 2. Basicamente foi mantida a mesma estrutura da versão anterior do sistema, porém, agregando os novos componentes necessários.

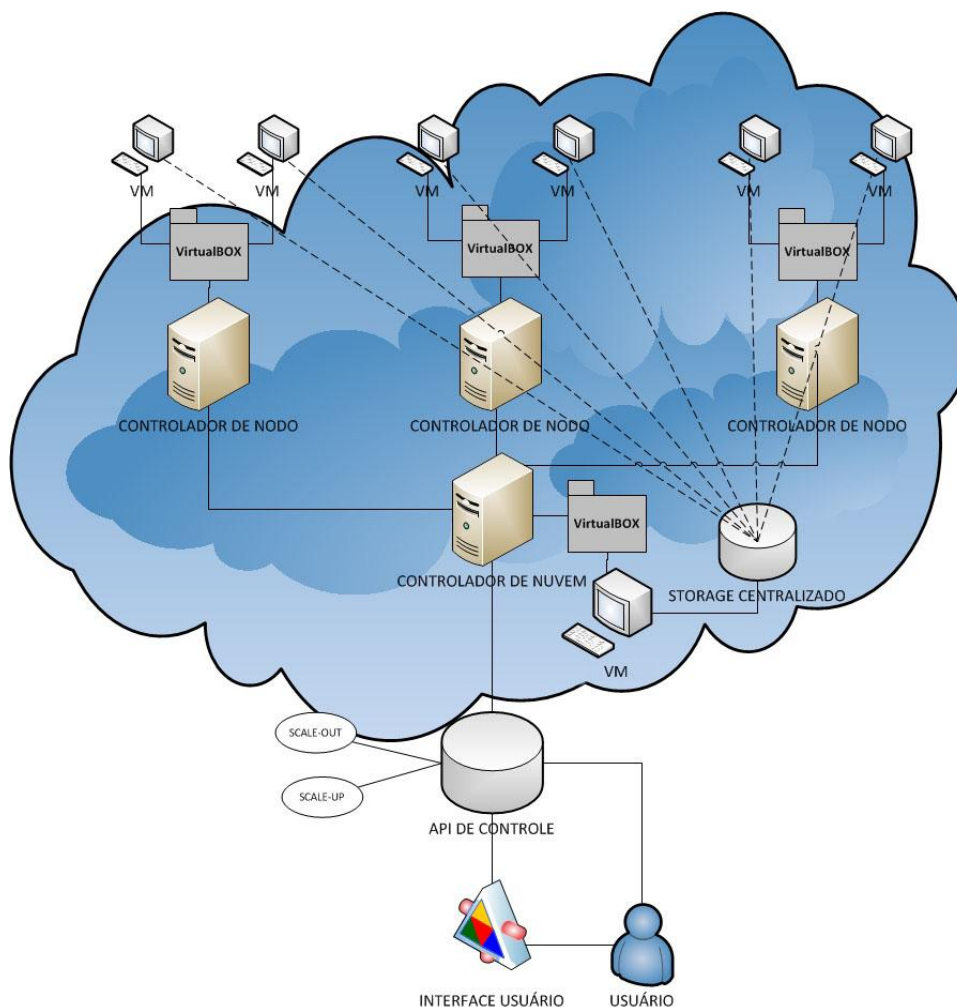


Figura 12: Arquitetura do Educloud 2.

Como elemento principal da nova versão do sistema Educloud, uma unidade de armazenamento centralizado foi adicionada à arquitetura do sistema. Desta forma, todas as máquinas virtuais gerenciadas pelo Educloud, mantêm suas unidades de disco armazenadas remotamente nesta unidade. Sendo assim, cada uma das máquinas virtuais possuirá apenas um *target* iSCSI direcionado para a unidade de armazenamento centralizado do Educloud 2. Para viabilizar a implantação desta funcionalidade, a ferramenta VirtualBox [18], passa também a ser requisito do componente Controlador de Nuvem, onde este será responsável por manter a máquina virtual que fará o papel de unidade de armazenamento centralizado.

Além das modificações realizadas na arquitetura principal do Educloud, foram adicionadas novas funcionalidades ao sistema, a fim de garantir um ambiente mais elástico e dinâmico para seus usuários. Sendo responsável pelo redimensionamento das máquinas virtuais, a função de *scale-up*, oferece ao usuário a possibilidade de alterar os recursos computacionais alocados para cada uma de suas VMs em plena execução. Para esta versão está sendo disponibilizado o redimensionamento da quantidade de processadores, quantidade de memória e o gerenciamento do CAP dos processadores.

Com o objetivo de aumentar o poder de processamento de um serviço, a função *scale-out*, foi adicionada ao Educloud 2. Utilizando-se de uma função responsável por clonar máquinas virtuais, disponibilizada pela ferramenta VirtualBox [18], cada máquina virtual que for criada com suporte a *scale-out*, poderá ser multiplicada formando uma espécie de cluster de processamento. Para viabilizar a implantação desta função, foi necessária a utilização de um balanceador de carga, responsável este, por dividir a carga de trabalho entre as máquinas virtuais multiplicadas.

4.2 DESCRIÇÃO DAS FUNCIONALIDADES

Nesta seção são apresentadas as novas funcionalidades adicionadas à versão atual do sistema, o Educloud 2. Desta forma, estão apresentados de forma detalhada os processos de modelagem e implementação das funções de armazenamento centralizado, gerenciamento de máquinas Virtuais, *scale-up* e *scale-out*.

4.2.1 Armazenamento Centralizado

Uma unidade de armazenamento centralizado pode ser implantada de diversas formas, sendo a mais comum à utilização de um *hardware* dedicado, também conhecido como *storage*. Este equipamento possui um custo de aquisição e manutenção muito elevado. Para contornar esse tipo de problema existem soluções baseadas em *software*, as quais são capazes de emular uma unidade de *storage* através de uma máquina comum ou até mesmo uma máquina virtual.

Devido ao Educloud 2 ser uma aplicação voltada para ambientes acadêmicos, e também devido ao alto custo de aquisição de um *hardware* dedicado para *storage*, a unidade de armazenamento centralizado do Educloud 2, é baseada em uma ferramenta de emulação, onde esta é executada a partir de uma máquina virtual. Durante o processo de definição da arquitetura deste componente, foram estudadas ferramentas que oferecem solução dedicada para este fim, como o Openfiler [32] e o FreeNAS [36]. A utilização destas ferramentas facilitaria o processo de implantação, uma vez que todo o gerenciamento dos volumes de disco seria realizado por estas. Durante o início do processo de desenvolvimento deste componente, foi identificada uma limitação por parte destas ferramentas, quanto a sua facilidade de integração com o Educloud 2. Tanto a ferramenta FreeNAS [36] quanto a Openfiler [32], não oferecem um *webservice* para realizar chamadas remotas, permitindo apenas a interação por algum navegador de internet. Desta forma, a unidade de armazenamento centralizado do Educloud 2 foi implementada através da utilização de uma mini distribuição Linux (CentOS 6.2) [39]. Esta passa a ser responsável por todo o processo de gerenciamento dos volumes de disco, utilizando comandos do próprio sistema operacional, permitindo uma integração direta entre o Controlador de Nuvem e a unidade de armazenamento centralizado.

Ao adicionar o conceito de armazenamento centralizado na arquitetura do sistema Educloud 2, criou-se a necessidade de reavaliar o processo de ciclo de vida de uma máquina virtual dentro da aplicação. Para modificar a forma de criação e armazenamento das unidades de disco das máquinas virtuais, todo o processo de criação, remoção, início e parada destas, deverá obrigatoriamente ser modificado.

4.2.1.1 Cadastro de Templates

Todo o processo de criação de máquinas virtuais dentro do Educloud 2 é baseado no conceito de *templates*. Cada *template* carrega a imagem de uma máquina virtual pronta para uso, e estes deverão ser previamente cadastrados no sistema. Na primeira versão do Educloud, após realizar o cadastro de *templates*, estes eram armazenados em um diretório dentro do Controlador de Nuvem. Quando da necessidade da utilização deste *template*, criando uma nova máquina virtual, o mesmo era copiado para um novo diretório, que mantinha as máquinas virtuais

registradas na nuvem do Educloud 2. A cada necessidade de execução destas máquinas, elas deveriam ser copiadas do Controlador de Nuvem para o Controlador de Nodo que foi escolhido para executá-la. Todo esse processo era extremamente demorado e foi um dos principais motivadores para a inclusão de uma unidade de armazenamento centralizado no Educloud 2.

Para permitir esta alteração de arquitetura, o primeiro ponto a ser modificado foi o gerenciamento dos *templates* do Educloud. Com a necessidade da inclusão de uma máquina virtual dedicada para o gerenciamento de discos, os *templates* deveriam necessariamente ser armazenados também nesta VM. Para isso, foi desenvolvida uma nova classe, a StorageManager, a fim de prover a integração entre o Controlador de Nuvem e unidade de armazenamento. Além destas alterações, também foram necessárias algumas modificações na base de dados da aplicação, onde esta passou a armazenar informações que fazem referência entre à unidade de armazenamento centralizado e as VMs.

Para os usuários, as modificações realizadas não causaram nenhum impacto, uma vez que o procedimento para estes continuará o mesmo. Os arquivos de *template* deverão ser previamente salvos no diretório *templatedir*, utilizando o formato de disco virtual da ferramenta VirtualBox [18] (VDI), dentro do sistema de arquivos do Controlado de Nuvem.

Após a realização do cadastro, através da API do VirtualBox [18], o Controlador de Nuvem solicita a inclusão do disco VDI, informado no cadastrado, como um novo dispositivo na máquina virtual responsável pela unidade de armazenamento. Por sua vez, este é anexado como um novo dispositivo local, permitindo o melhor gerenciamento do *template* durante o processo de criação de novas máquinas virtuais.

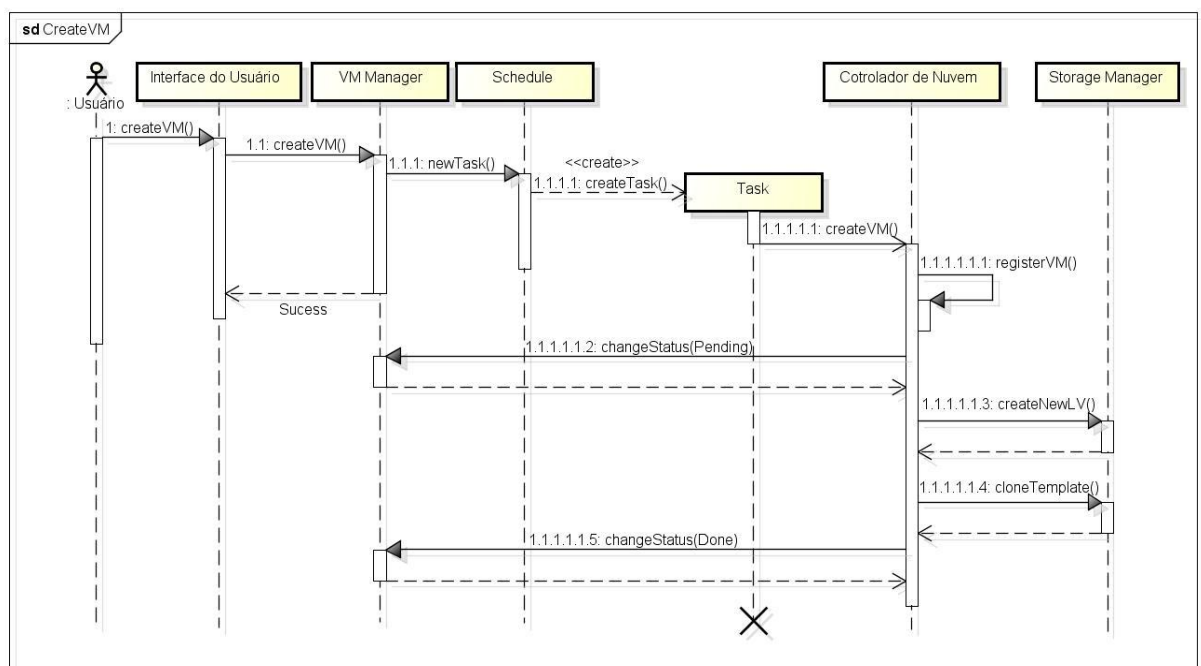
4.2.2 Gerenciamento de Máquinas Virtuais

Esta seção apresenta uma visão geral sobre os processos de gerenciamento de máquinas virtuais dentro do Educloud 2. São apresentadas as principais modificações na nova versão do sistema.

4.2.2.1 Criação e Remoção

O processo de criação de uma nova máquina virtual na nuvem do Educloud depende da existência de um *template* válido já cadastrado. Durante a solicitação de criação de uma nova VM, este *template* será passado como parâmetro para aplicação, a qual utilizará as informações deste para criar o novo volume de disco que será utilizado pela nova máquina virtual.

Ao adicionar o conceito de unidade de armazenamento centralizado na arquitetura do Educloud 2, todo o processo de criação e remoção de máquinas virtuais foi modificado. O gerenciamento das unidades de disco de cada máquina virtual passa a ser realizado pela unidade de armazenamento e não mais de forma local em cada nodo. Por este motivo, foi implementada a integração entre o Controlador de Nuvem e a unidade de armazenamento do Educloud 2. Na Figura 13 é apresentado o diagrama de seqüência referente ao processo de criação de uma nova máquina virtual.



powered by astah

Figura 13: Diagrama de seqüência para criação de uma máquina virtual.

Quando é realizada a solicitação de criação de uma nova máquina virtual, uma nova tarefa é criada dentro do sistema, onde esta será enviada para o Schedule do Educloud e terá sua execução agendada. Quando a tarefa for selecionada para execução, o Controlador de Nuvem fará o registro da nova

máquina virtual no banco de dados da aplicação, onde esta permanecerá no estado “*Pending*” até que o processo seja finalizado.

Para criar os volumes de disco dentro da unidade de armazenamento centralizado, foi utilizado o conceito de volumes lógicos do sistema operacional Linux (Logical Volume Manager) [38]. O LVM [38] permite emular novos dispositivos de armazenamento através da criação de arquivos de tamanhos fixos. Desta forma, sempre que uma solicitação para criar uma nova máquina virtual é realizada, um novo volume lógico é adicionado dentro da unidade de armazenamento do Educloud 2. Após criar o novo volume lógico, o próximo passo a ser realizado é o clone da unidade de disco que foi anexada na unidade de armazenamento durante o cadastro do *template*. Para realizar esta tarefa é utilizado o comando do sistema operacional Linux chamado “*dd*”. Este comando é responsável por copiar, converter e até mesmo clonar dispositivos de armazenamento em massa. Para isso, é informado o dispositivo *template*, já anexado na unidade de armazenamento, como dispositivo de origem e o novo volume lógico, recém criado, como dispositivo de destino. Ao final do processo, o novo volume lógico possuirá uma cópia idêntica ao disco do *template*.

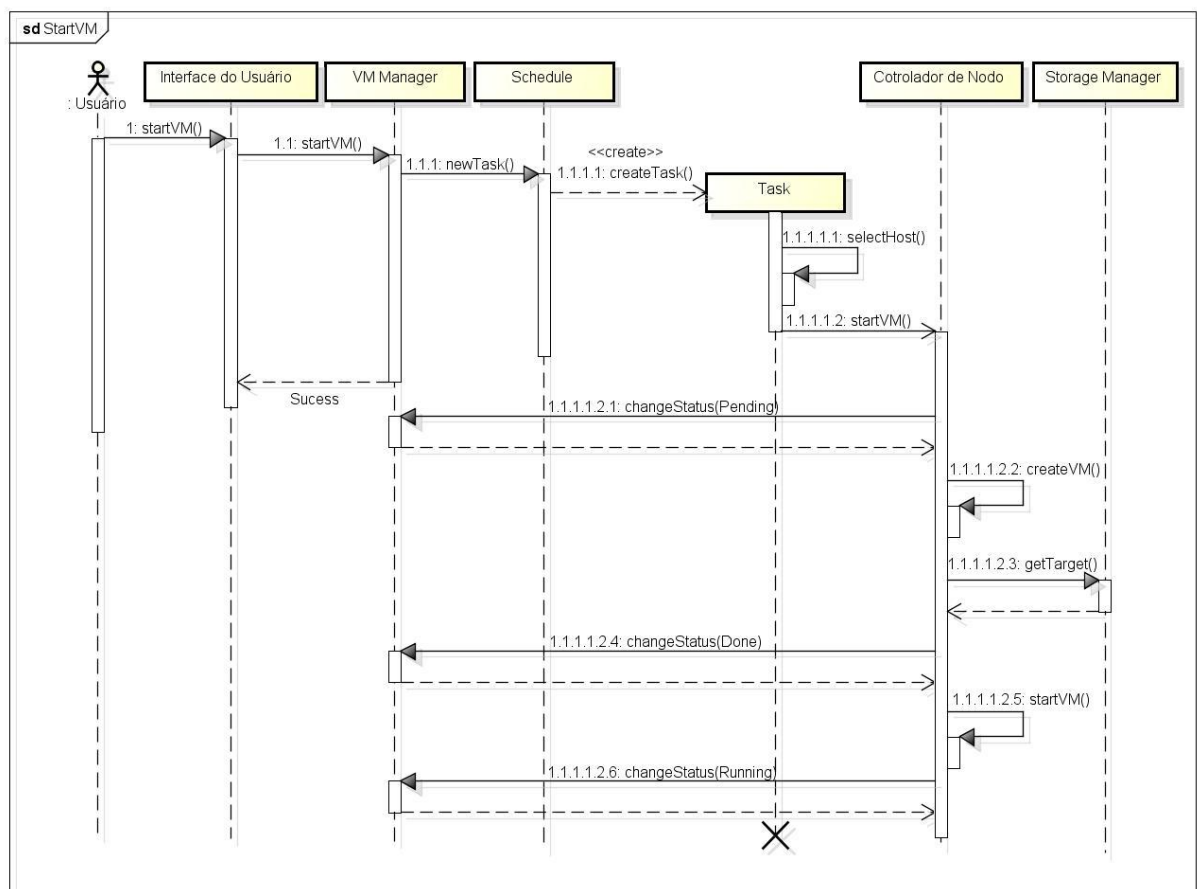
Após a conclusão destas etapas um novo objeto de máquina virtual estará criado dentro do Educloud 2, e seu respectivo volume de disco já estará criado na unidade de armazenamento centralizado.

O processo de remoção de uma máquina virtual, além de remover as informações registradas no banco de dados da aplicação, também faz a remoção física do volume lógico que pertence à mesma dentro da unidade de armazenamento centralizado. Desta forma, é garantido que toda a informação gerada durante a existência desta máquina virtual, será excluída da aplicação.

4.2.2.2 Inicialização e Parada

Assim como as demais funcionalidades vistas até o momento, o processo de inicialização de uma máquina virtual também sofreu algumas modificações, a fim de suportar a nova unidade de armazenamento centralizado do Educloud 2. Para tornar possível a vinculação do volume lógico, criado na unidade de armazenamento, e da máquina virtual a ser iniciada, é preciso criar uma referência para este disco, através

de um *target* iSCSI. Este processo faz com que o volume lógico criado anteriormente se torne público para a nuvem e então permita que este seja referenciado dentro da nova máquina virtual. Assim como nos processos de criar volume lógico e clonar *template*, a criação do *target* iSCSI é realizada através da classe *StorageManager*, responsável pela interação com a unidade de armazenamento do Educloud 2. A Figura 14 apresenta o diagrama de seqüência referente ao processo de inicialização de uma máquina virtual.



powered by astah

Figura 14: Diagrama de seqüência para iniciar uma máquina virtual.

Assim como no processo de criação de máquinas virtuais, uma nova tarefa é criada na solicitação de inicialização de uma VM. Quando esta é selecionada para execução, é feita uma verificação dos recursos disponíveis na nuvem, a fim de identificar em qual nodo será iniciada esta VM. Este processo de escolha do nodo destino é realizado por um subcomponente do Controlador de Nuvem, o *NodeSelector*, o qual é responsável pela aplicação de políticas de alocação dentre os nodos existentes. [17]

Mesmo tratando-se de um processo de inicialização, dentro do Educloud, as máquinas virtuais somente são criadas fisicamente durante este processo. Esta estratégia garante a integridade das máquinas virtuais criadas na nuvem, uma vez que estas não possuem nenhum vínculo com um nodo em específico, podendo ser iniciadas em qualquer nodo presente na nuvem.

Para criação e gerenciamento de *targets* iSCSI dentro da unidade de armazenamento, é utilizada uma ferramenta do sistema operacional Linux “*tgtadm*” (*SCSI Target Administration Utility*). Esta ferramenta permite transformar um dispositivo de armazenamento local em um dispositivo remoto na nuvem, criando para este um endereço público através da utilização do protocolo iSCSI. O processo é basicamente dividido em duas etapas, criação do novo endereço público (*target*) e vinculação deste a uma unidade de volume lógico local. Na etapa de criação, um novo *target* é criado na aplicação, onde este é nada mais do que uma chave de conexão externa que será utilizada para localizar remotamente o novo dispositivo. Após a criação do novo *target*, este deverá ser vinculado a um volume lógico local existente. Esta tarefa simplesmente irá informar o endereço público “x” para o volume lógico “y”. Desta forma, um novo dispositivo iSCSI passa a ser disponibilizado para a aplicação.

Após a solicitação de uma tarefa de inicialização, o Controlador de Nodo seleciona a máquina virtual a ser iniciada e solicita ao StorageManager a criação de um novo *target* referenciando o disco que foi anteriormente criado para a mesma. Com base no nome do volume lógico, criado para esta máquina virtual, um novo *target* é disponibilizado, e assim permite a inicialização da máquina solicitada.

O processo de parada de uma máquina virtual, além de encerrar a execução da mesma, realiza a exclusão do *target* iSCSI junto à unidade de armazenamento. Este procedimento foi adotado para evitar a inconsistência nas informações do banco de dados do sistema, devido a alguma falha na unidade de armazenamento.

4.2.3 Scale-up

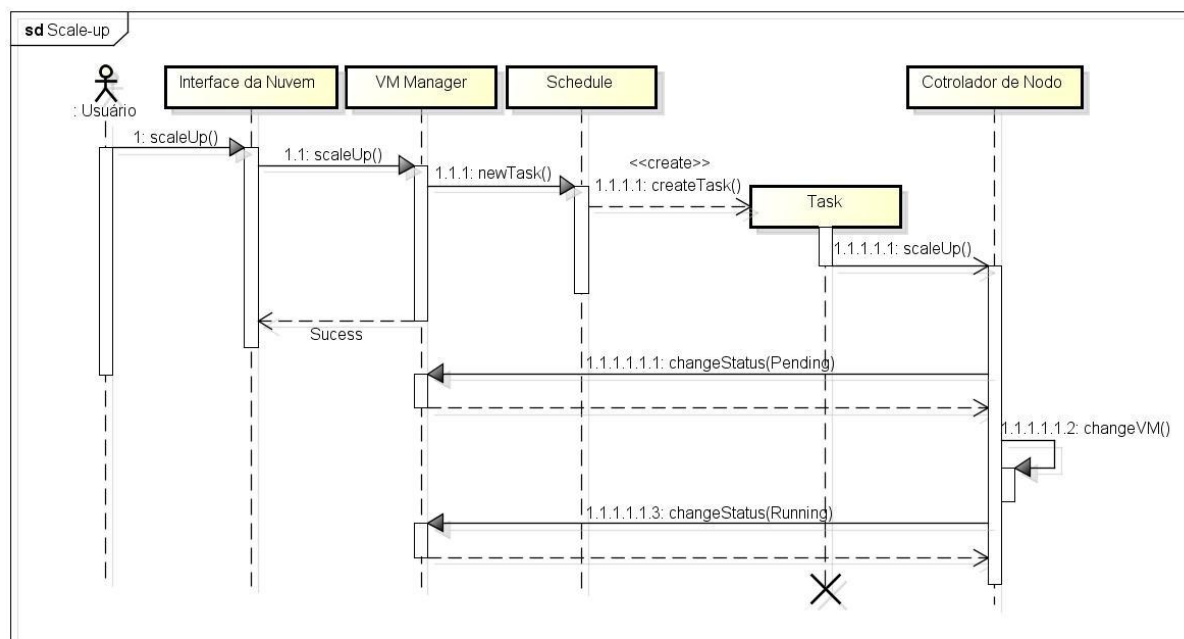
O processo de *scale-up* consiste em uma solicitação de redimensionamento de uma máquina virtual, permitindo alterar quantitativamente os recursos computacionais alocados para esta.

Com o objetivo de adicionar ao Educloud 2 características de elasticidade, o processo de *scale-up* foi implementado de modo a permitir que o redimensionamento das máquinas virtuais possa ocorrer com estas em plena execução. Isto foi possível devido a uma nova *release* da ferramenta VirtualBox [18], a qual passou a permitir o redimensionamento das máquinas sem a necessidade de uma reinicialização das mesmas. Desta forma, o usuário pode gerenciar os recursos computacionais de sua máquina virtual de forma mais dinâmica e de acordo com seu ambiente. Entretanto, alguns sistemas operacionais não oferecem suporte para alterações dinâmicas de *hardware*. Sendo assim, a reinicialização da máquina virtual pode ser necessária nesses casos.

Como descrito anteriormente, durante o processo de *scale-up*, uma máquina virtual pode ser redimensionada com base em seus recursos computacionais previamente alocados. Dentro do Educloud 2, o processo de *scale-up* abrange três recursos de *hardware* da VM, sendo estes: quantidade de memória, quantidade de processadores e capacidade de processamento.

O parâmetro quantidade de memória define a quantidade de memória física que é alocada para máquina virtual, baseando-se na quantidade real de memória presente no nodo que está executando a VM. Durante o processo de *scale-up*, esse parâmetro poderá ser alterado a qualquer momento, de acordo com a necessidade do usuário.

No nível de processamento da máquina virtual, são disponibilizados dois parâmetros para alteração. O parâmetro quantidade de processadores indica o número de processadores virtuais destinados à VM. Estes são disponibilizados pelo nodo com base na quantidade de núcleos de cada processador existente neste. Além disso, o *scale-up* também oferece a alteração do parâmetro capacidade de processamento (*CAP*). Este parâmetro, como base na quantidade de processadores previamente alocados, define um percentual de utilização destes pela máquina virtual. Sendo assim, é possível manter uma quantidade “X” de processadores alocados para uma VM, utilizando apenas 20% de sua capacidade total. Desta forma, é possível realizar um *scale-up* na máquina virtual, mesmo que não existam processadores disponíveis no nodo, sendo necessário apenas ajustar seu *CAP* para 100%. A Figura 15 apresenta o diagrama de sequência para a realização do processo de *scale-up*.



powered by astah®

Figura 15: Diagrama de sequencia para função scale-up.

Assim como nas demais funcionalidades do Educloud 2, ao solicitar a execução de um *scale-up*, uma nova tarefa é criada no Schedule da aplicação. Esta tarefa cria uma solicitação de alteração dos recursos previamente alocados para a VM, sendo em nível de memória ou de processamento. Estas alterações possibilitam a realocação dos recursos computacionais do nodo para a máquina virtual em questão, seja para mais ou para menos. Contudo, algumas alterações somente serão realizadas se existirem recursos disponíveis para estas, como a quantidade de memória e a quantidade de processadores. Esta verificação é realizada internamente pelo Educloud 2 utilizando os recursos disponibilizados pelo VirtualBox [18] para isto.

4.2.4 Scale-out

Para agregar ao Educloud 2 características de escalabilidade, a função *scale-out* foi implementada. As técnicas de *horizontal scaling* ou *scale-out*, permitem aumentar ou diminuir a capacidade de processamento de uma aplicação através da duplicação, ou clonagem, de uma máquina virtual. Para viabilizar a implantação desta funcionalidade, além da utilização da ferramenta VirtualBox [18], responsável pelo gerenciamento das máquinas virtuais, uma aplicação para realizar o balanceamento de carga foi utilizada. Esta ferramenta é responsável por dividir a

carga de trabalho de uma aplicação qualquer, entre as diversas instâncias de máquinas virtuais clonadas, transparecendo a existência de uma única máquina para o usuário final.

A ferramenta para realizar o balanceamento de carga no Educloud 2 é o *LVS* (*Linux Virtual Server*) [37]. Durante a etapa de planejamento deste projeto, alguns pontos levaram a definição desta ferramenta para compor a arquitetura do Educloud 2. Dentre estes é possível citar a sua versatilidade quanto a requisitos de hardware e software, sua política de licenciamento livre e principalmente por ser um serviço nativo no sistema operacional Linux.

4.2.4.1 Linux Virtual Server (LVS)

Como uma solução para aplicações de alto desempenho, o *Linux Virtual Server* (LVS) [37], implementa um balanceador de carga, a fim de distribuir a carga de trabalho de maneira uniforme entre dois ou mais servidores do *cluster* em questão. Desta forma, aplicações que utilizam o LVS garantem uma otimização de uso dos recursos do *hardware* e aumento considerável no desempenho, diminuindo a possibilidade de sobrecarga. O LVS permite a abstração de um grande cluster de servidores, tornando público a seu cliente externo um único servidor. Esse servidor visível ao cliente é considerado um servidor virtual, onde os servidores reais são controlados pelo *director* do LVS, que é o responsável pelo balanceamento de carga em si.

O *director*, funciona como uma forma de intermediador da aplicação que o LVS está controlando, sendo ele o responsável direto por realizar o balanceamento da carga de trabalho. O *director* pode utilizar três formas para fazer o encaminhamento das solicitações:

- LVS-NAT: Realiza uma tradução dos endereços de rede entre o servidor virtual e os servidores reais (NAT);
- LVS-DR: Os endereços físicos (MAC) dos pacotes são alterados e reenviados diretamente para o servidor real escolhido (roteamento direto);

- LVS-Tun: Utiliza a técnica de tunelamento, onde os pacotes são encapsulados e remetidos ao servidor real escolhido.

Para a implementação da funcionalidade de *scale-out* no Educloud 2 foi utilizado o modelo LVS-DR. O principal fator que influenciou esta escolha foi a realidade do ambiente de nuvem privada disponibilizado pela aplicação, onde todas as máquinas virtuais da nuvem compartilham a mesma rede, e assim exigem a necessidade da utilização de técnicas de NAT ou Tunelamento. A Figura 16 apresenta a modelo de balanceamento de carga LVS-DR, ou roteamento direto, utilizado no Educloud 2.

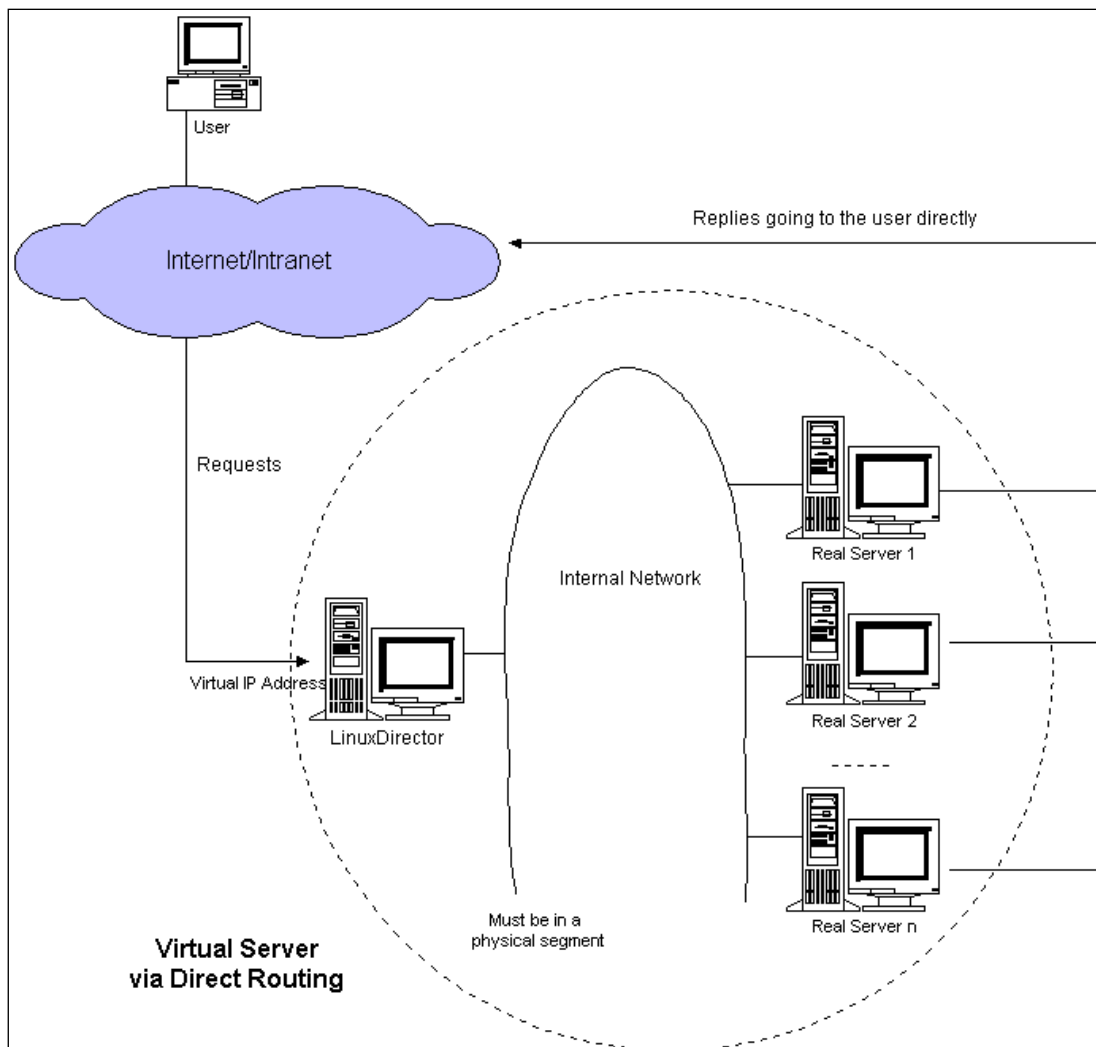


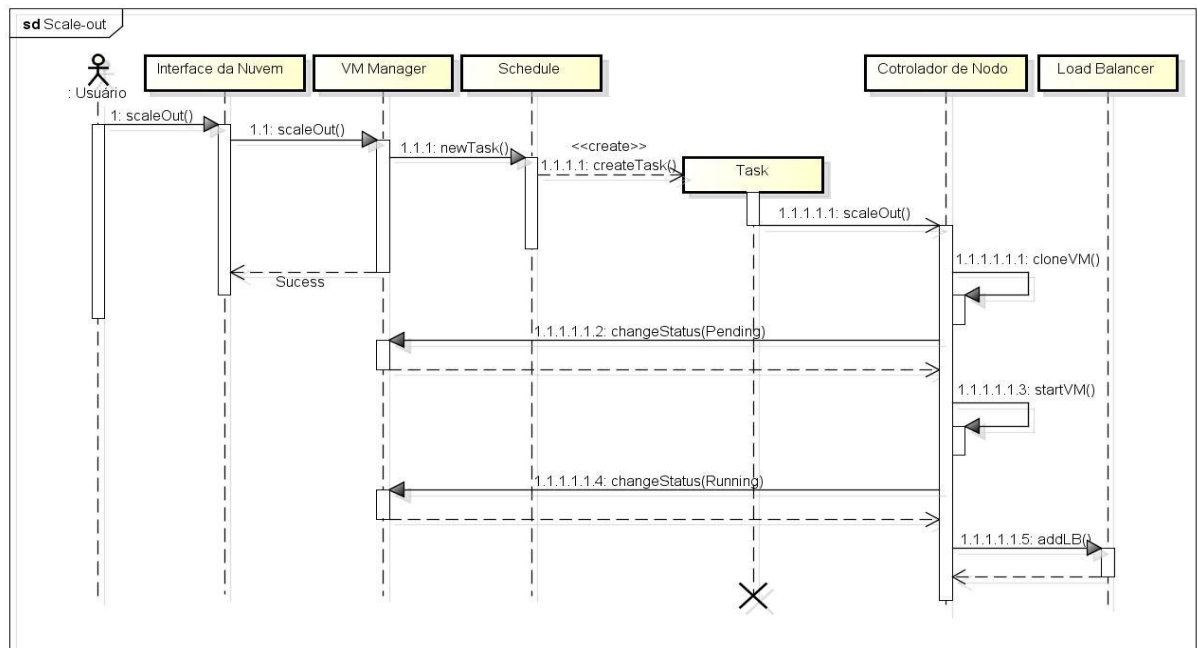
Figura 16: Virtual Server via roteamento direto. [37]

O LVS é muito versátil, permitindo sua utilização em diversos cenários. Ele permite a utilização desde servidores reais idênticos, rodando uma única aplicação

em cada, até servidores reais com hardware e sistemas operacionais diferentes, rodando diversas aplicações também diferentes. [31]

4.2.4.2 LVS x Educloud 2

Através do Educloud 2, é oferecido ao usuário a capacidade de realizar *scale-out* em suas máquinas virtuais. Para isso, deverá ser utilizada uma VM que ofereça suporte a este serviço. Esta definição é realizada durante o processo de criação de uma nova máquina virtual, onde é possível solicitar que a mesma possua suporte a *scale-out*. Desta forma, ao criar a máquina virtual, uma nova instância de um balanceador de carga é criada, e automaticamente, a nova VM será vinculada a este. Após a criação de uma máquina virtual com suporte a *scale-out*, este poderá ser realizado quantas vezes forem necessárias. A cada chamada da função *scale-out*, uma nova instância da máquina virtual será duplicada e adicionada ao seu balanceador de carga. Essa duplicação acontece através de uma função de clonagem disponibilizada pela aplicação VirtualBox [18], onde uma máquina virtual é copiada fisicamente gerando uma cópia idêntica da mesma. A Figura 17 apresenta o diagrama de sequência referente à função de *scale-out*.



powered by astah

Figura 17: Diagrama de sequencia para função scale-out.

Quando uma máquina virtual com suporte a *scale-out* é criada no Educloud 2, além desta, uma outra máquina virtual é também criada na aplicação. Esta outra máquina virtual é uma unidade de balanceamento de carga, a qual é necessária para a distribuição da carga de serviço dentre as VMs do cluster. O processo de criação desta máquina é automático e transparente para o usuário, onde este somente terá acesso ao endereço IP da mesma.

No momento em que é feita a solicitação de *scale-out*, uma nova tarefa é criada no Schedule do sistema. O seu processo de execução está dividido basicamente em duas etapas: clonar a máquina de origem e adicionar a nova máquina ao balanceador de carga correspondente.

A função de clonagem foi desenvolvida exclusivamente para suportar esta nova funcionalidade da aplicação. Contudo, esta função já é oferecida pelas versões mais recentes da ferramenta VirtualBox [18], porém, oferecendo suporte somente a arquivos de disco no formato VDI. Desta forma, a função de clonagem do Educloud 2, foi desenvolvida para oferecer suporte a unidade de armazenamento centralizado.

Após a máquina de origem estar clonada e iniciada, para finalizar o processo de *scale-out*, esta deverá ser adicionada ao balanceador de carga correspondente a sua máquina de origem. Esse processo é realizado pela classe BalancerManager, que é responsável pela integração com os balanceadores de carga da aplicação. Internamente, esta classe faz uso do comando *ipvsadm* para viabilizar a execução do *scale-out*. Este comando é disponibilizado pela aplicação LVS e possibilita a configuração e o gerenciamento dos balanceadores de carga necessários para o *scale-out*. Através de uma sintaxe simples e objetiva, alguns parâmetros são informados, com o objetivo de definir o modelo de LVS a ser utilizado, no caso do Educloud 2 LVS-DR, e também a localização das máquinas que irão compor o cluster, com base em seu endereço IP.

4.2.5 API de Controle

A API de controle foi desenvolvida durante a primeira versão do sistema Educloud. Seu principal objetivo é oferecer uma integração fácil para outros sistemas Java que queiram utilizar o Educloud. Desta forma, desenvolvedores

podem criar uma integração direta entre a API e outro sistema qualquer, o qual possua características mais familiares para seus usuários. [17]

Basicamente a estrutura da API de Controle é composta por uma coleção de classes cliente do componente Controlador de Nuvem, onde estas oferecem controle sob todas as operações oferecidas pela Educloud 2. Abaixo serão listadas as principais classes disponibilizadas pela API de Controle do Educloud 2. [17]

- EduCloudConfig: Realiza uma solicitação de configuração do sistema, informando os dados necessários para iniciar uma nova sessão. Dentre os dados informados estão a localização do Controlador de Nuvem, o usuário e senha para acesso, a localização da unidade de armazenamento;
- EduCloudAuthorization: Armazena os dados referentes a autorização da nova sessão inicializada pelo usuário;
- EduCloudNodeClient: Possui operações para recuperar informações dos nodos da nuvem;
- EduCloudTemplateClient: Possui operações para realizar o gerenciamento dos *templates* da nuvem, seja na recuperação de informações, criar ou excluir os mesmos;
- EduCloudUserClient: Possui operações para realizar o gerenciamento dos usuários da nuvem, seja na recuperação de informações, criar ou excluir os mesmos;
- EduCloudVMClient: Possui operações para realizar o gerenciamento completo das máquinas virtuais existentes na nuvem, permitindo a execução de qualquer operação nas mesmas.

Durante o desenvolvimento do Educloud 2, foram necessárias algumas alterações na estrutura da API de Controle. Contudo, nenhuma modificação dentre as classes existentes foi realizada. Apenas foram agregadas as novas funcionalidades de gerenciamento das máquinas virtuais através da classe EduCloudVMClient. A Figura 18 apresenta a versão atual do diagrama de classes da API de Controle do Educloud 2.

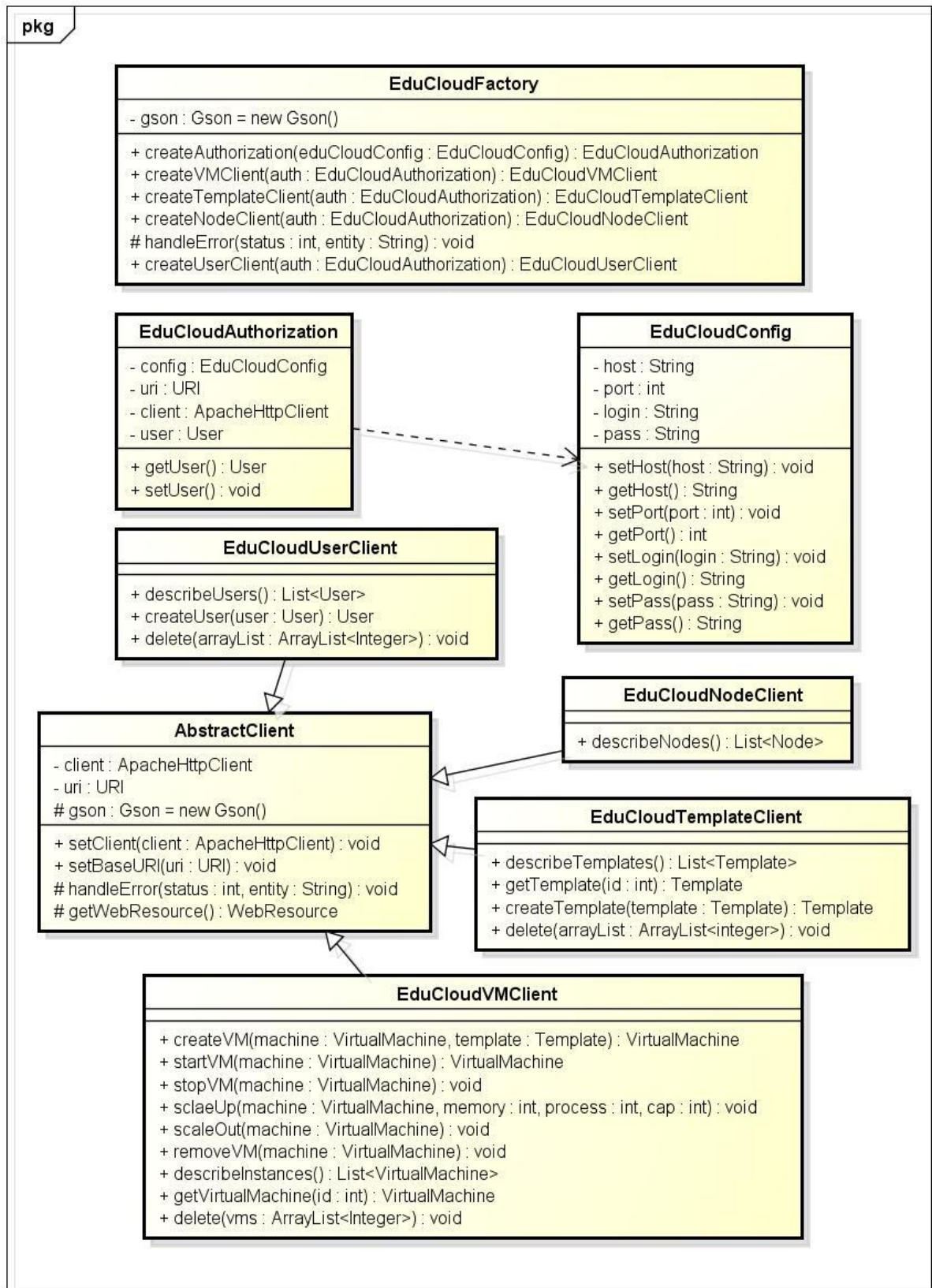


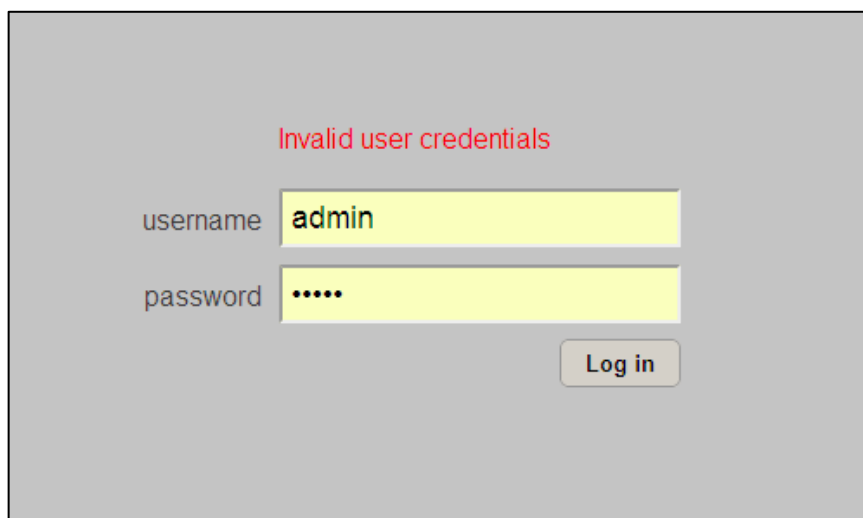
Figura 18: Diagrama de classes da API de Controle.

4.2.6 Interface de Usuário

A interface de usuário é implementada como um portal web que proporciona acesso a todas as operações ofertadas pelo Educloud 2. Esta interface foi desenvolvida durante a primeira versão da aplicação, permitindo a visualização das máquinas virtuais, informações referentes aos *templates* cadastrados, além de gerenciar todo o ciclo de vida das máquinas virtuais. [17]

Para o desenvolvimento deste portal foi utilizada a plataforma Java, baseando-se na tecnologia JSP (Java Server Pages) [40], para o gerenciamento da camada web. Para disponibilizar as funcionalidades presentes na interface de usuário, foi utilizada a API de Controle do Educloud. Sendo assim, o principal objetivo durante o desenvolvimento desta interface, foi criar uma forma mais simples e amigável para prover acesso à aplicação. [17]

Na Figura 19 é apresentada a tela utilizada para autenticação dos usuários, a fim de obter acesso às funcionalidades da aplicação. Estes usuários devem ser previamente cadastrados no banco de dados do Educloud 2.



A imagem mostra uma interface de autenticação de usuários. No topo, há uma mensagem de erro em vermelho: "Invalid user credentials". Abaixo, há dois campos de entrada: "username" com o valor "admin" e "password" com caracteres ocultos por pontos. Um botão "Log in" está localizado à direita dos campos.

Figura 19: Tela para autenticação de usuários.

Após realizar a autenticação no sistema o usuário terá acesso ao painel de controle do sistema Educloud 2. A Figura 20 apresenta a tela de gerenciamento dos controladores de nodo, permitindo a visualização de informações como endereço IP, porta de execução do nodo e se este está ou não conectado ao VirtualBox [18].

localhost:8000/ui/nodes.jsp

EduCloud 2

EduCloud Home Virtual Machines **Cloud Resources** Templates Users

All resources

ID	Hostname	Port	Connected to Vbox	Vbox version	Memory
1	127.0.0.1	8111	true	4.1.14	1768/3048

Figura 20: Tela para visualização dos Controladores de Nodo.

Na Figura 21 é apresentada a tela de gerenciamento de *templates* oferecida no Educloud 2. Através dessa tela é possível verificar e cadastrar novos *templates*, vinculando-os automaticamente na unidade de armazenamento centralizado.

localhost:8000/ui/templates.jsp

EduCloud 2

EduCloud Home Virtual Machines Cloud Resources **Templates** Users

[New Template](#)

Select: [All](#) | [None](#) Actions... ▾

-	ID	Name	Description	Memory Size	Processors	Device	Port	OS Type
<input type="checkbox"/>	1	Ubuntu-TPL	Template de Ubuntu	256 MB	1	/dev/sdd	3	Ubuntu

Figura 21: Tela para visualização e cadastro de templates.

A Figura 22 apresenta a tela responsável pelo gerenciamento das máquinas virtuais na interface de usuário. Através desta tela é possível controlar todas as máquinas virtuais existentes na nuvem, permitindo inclusive executar as novas funcionalidades do Educloud 2 como *scale-up* e *scale-out*.

EduCloud 2

EduCloud Home Virtual Machines Cloud Resources Templates Users

[New Virtual Machine](#)

Select: [All](#) | [None](#) Actions... ▾

ID	Name	Description	Memory Size	Processors	Processor Capacity	OS Type	LoadBalancer	IP LB	State	Actions
<input type="checkbox"/> 1	Maquina1	Maquina com LB	256 MB	1	30	Ubuntu	Yes	10.9.0.212	RUNNING	Start Stop Scale Out
<input type="checkbox"/> 3	Maquina2	Maquina sem LB	256 MB	1	30	Ubuntu	No	-	DONE	Start Stop
<input type="checkbox"/> 4	clone_Maquina1	Cloned VM FROM vm 1	256 MB	1	30	Ubuntu	Yes	10.9.0.212	PENDING	Start Stop

Scale-up

Figura 22: Tela para gerenciamento de máquinas virtuais.

Na Figura 23 é apresentada a tela utilizada para realização da funcionalidade de *scale-up*. Nela é possível informar os novos valores que devem ser assumidos pela máquina virtual que foi selecionada. Para esta função é possível realizar a alteração dos parâmetros que indicam a quantidade de memória, a quantidade de processadores e o gerenciamento do CAP dos processadores.

EduCloud 2

ScaleUp - Settings

Virtual Machine Info

id	3
Name	Maquina2
Description	Maquina sem LB
Memory	256 MB
Processor	1 Cores
Processor Capacity	30 %

Save Close

Figura 23: Tela para realização de scale-up.

Na Figura 22 é possível visualizar o link de *scale-out* localizado na guia de gerenciamento de máquinas virtuais do Educloud 2. Esta função fica disponível apenas para máquinas virtuais que possuam suporte a este serviço. Após o usuário pressionar o *link*, uma chamada é realizada para a função *scale-out* existente na API do Educloud 2. Ao término da execução, uma nova máquina virtual clonada é exibida e adicionada a seu respectivo balanceador de carga.

5 VALIDAÇÃO

Neste capítulo são apresentadas as atividades realizadas durante o processo de validação do Educloud 2. São demonstrados os escopos dos testes realizados, o ambiente utilizado e os resultados obtidos.

5.1 DESCRIÇÃO GERAL DOS TESTES

Para realizar a validação do sistema Educloud 2, foram realizados alguns testes funcionais no sistema, a fim de demonstrar e avaliar sua utilização. Os testes utilizados são estruturados com base em uma descrição do cenário envolvido, seguidos da análise de seus resultados. Com a realização dos testes é possível ter uma noção do funcionamento do sistema Educloud 2, além de avaliar alguns pontos do mesmo, como usabilidade e desempenho.

5.2 DESCRIÇÃO DO AMBIENTE DE TESTE

Para a realização dos testes funcionais no sistema Educloud 2, foi utilizado um laboratório de testes localizado junto ao departamento de TI de uma indústria na cidade de Cachoeirinha - RS. Durante a execução dos testes, foi criado um ambiente contendo três computadores interligados através de uma rede gigabit-ethernet. Estes computadores foram instalados utilizando o sistema operacional Linux Ubuntu 11.10, além das demais dependências de software necessárias para a execução da aplicação.

Foram utilizadas duas configurações de hardware para os computadores destinados aos testes do sistema. A máquina responsável pelo Controlador de Nuvem, aqui também chamada de servidor, possuía um processador Intel Core 2 QUAD de 2.4Ghz e 8GB de memória. Já para os dois computadores responsáveis pelos Controladores de Nodo, foram utilizados processadores Intel Core i3 2.8Ghz e 4GB de memória.

5.3 CADASTRO DE TEMPLATES

O cadastro de *templates*, mesmo sendo muito simples de ser compreendido, possui uma grande importância dentro do sistema. Esta atividade é responsável pela primeira interação com a unidade de armazenamento centralizado do Educloud 2. O teste relatado abaixo teve por objetivo mensurar o desempenho desta atividade durante o processo de cadastro e inclusão de um novo *template* na unidade de armazenamento centralizado.

Para realizar este teste foi utilizado um arquivo de disco virtual (VDI), que foi previamente salvo dentro do diretório *templatedir* do Controlador de Nuvem. Este processo realiza uma chamada para API do Educloud 2, passando os parâmetros necessários para sua execução. Os resultados obtidos durante este teste são apresentados na Figura 24.

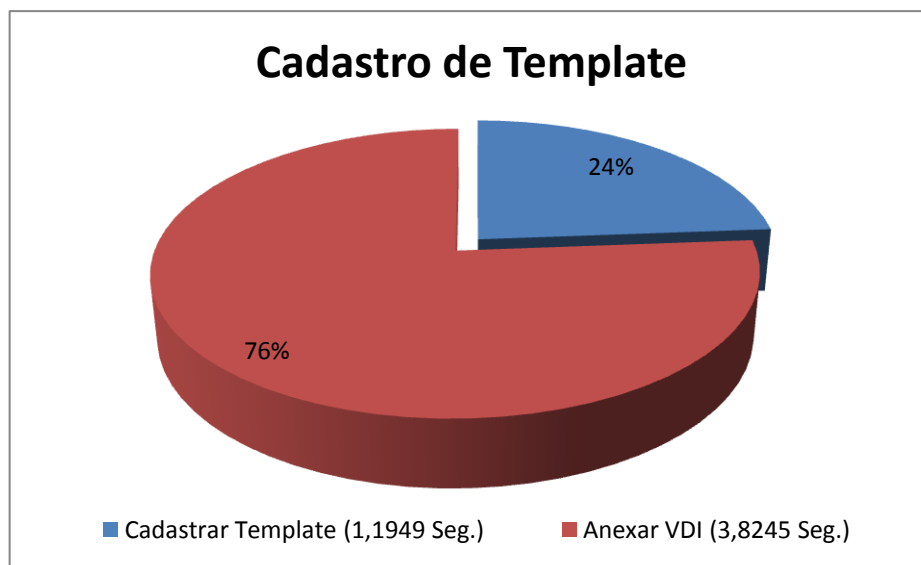


Figura 24: Resultados do teste para cadastro de templates.

O teste foi realizado utilizando um arquivo VDI de 2GB, e teve sua avaliação direcionada para as principais etapas deste processo: Cadastrar Template, responsável pela interação com banco de dados da aplicação, e Anexar VDI, responsável pela interação com a ferramenta VirtualBox [18].

Após a execução dos testes foi possível constatar a eficiência do processo de cadastro de *templates*, onde seu tempo total de execução foi de aproximadamente

5 segundos. Desta forma, foi constatado que o processo se manteve eficiente mesmo após a inclusão da unidade de armazenamento centralizado.

5.4 CRIAÇÃO DE MÁQUINAS VIRTUAIS

O processo de criação de máquinas virtuais é uma tarefa essencial para o funcionamento do Educloud 2. Esta funcionalidade atravessa todas as camadas do sistema, e diversas tarefas são realizadas. Durante a execução deste teste serão avaliadas as principais atividades executadas na integração com a unidade de armazenamento centralizado, sendo estas, criar novo volume lógico e clonar disco do *template*.

O maior desafio durante a execução desta funcionalidade está em seu tempo de execução. O processo de clonagem do *template* realiza uma cópia física do bloco de disco informado pelo *template* utilizado, tornando este um dos processos mais demorados da aplicação. Durante este teste foi criada uma máquina virtual Ubuntu utilizando um disco de 2GB. A Figura 25 demonstra as métricas de tempo obtidas durante a realização deste teste.

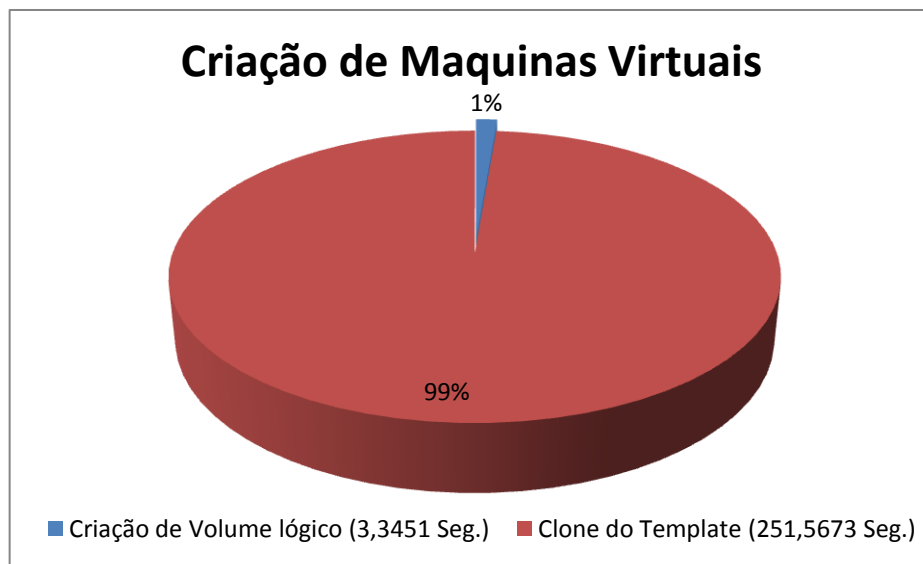


Figura 25: Resultados do teste para criação de VM.

Esta é uma das funcionalidades mais lentas do Educloud 2, porém, é notável a diferença, comparando-a com a primeira versão do Educloud, onde o processo de criação de máquinas virtuais durava em torno de 490 segundos. Após a realização dos testes foi constatado um ganho de desempenho de aproximadamente 48% em

relação à versão anterior do Educloud, onde o processo de criação de uma nova máquina virtual durou em torno de 255 segundos para ser completado.

Com a utilização da unidade de armazenamento centralizado não existe mais a necessidade de trafegar dados pela rede. Todo o gerenciamento dos discos fica a cargo da unidade de armazenamento, sendo esta responsável por administrá-los e distribuí-los via protocolo iSCSI.

5.5 INICIALIZAÇÃO DE MÁQUINAS VIRTUAIS

O processo de inicialização de máquinas virtuais do Educloud 2 é também uma atividade chave dentro da aplicação. Este processo gerencia a integração entre Controlador de Nuvem, Controladores de Nodo e unidade de armazenamento centralizado. Para este processo muitas tarefas são executadas, porém, serão analisadas somente as principais etapas executadas durante o mesmo, sendo a criação da máquina virtual no nodo, a criação do *target* iSCSI na unidade de armazenamento e a inicialização da máquina virtual no nodo.

Durante a realização deste teste foi analisado principalmente o desempenho da aplicação, através da análise do tempo gasto para conclusão de cada tarefa. A Figura 26 demonstra os resultados obtidos durante a execução da funcionalidade de inicialização de uma máquina virtual no Educloud 2.

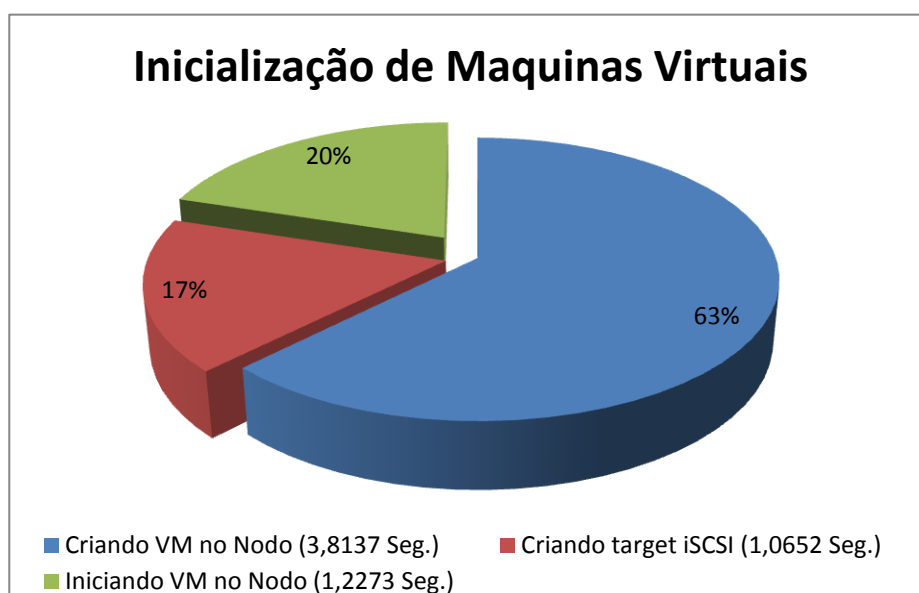


Figura 26: Resultados do teste para inicialização de VM.

A funcionalidade de inicialização de máquinas virtuais também sofreu uma grande modificação, comparando-a com sua versão anterior. A presença da unidade de armazenamento centralizado permitiu uma grande redução em seu tempo de execução, pelo fato de não haver mais a necessidade de trafegar os dispositivos de disco pela rede. Na versão anterior do Educloud, a execução do processo de inicialização de uma máquina virtual durava em torno de 222 segundos para ser completado. Já na versão atual, após a execução dos testes, foi identificado um ganho em desempenho de aproximadamente 97%, onde este foi executado em torno de 6 segundos no Educloud 2.

5.6 REDIMENSIONAMENTO DE MÁQUINAS VIRTUAIS

O processo de *scale-up*, ou redimensionamento de máquinas virtuais, é uma nova funcionalidade no sistema Educloud 2, e permite redimensionar os recursos alocados para uma máquina virtual em plena execução. Durante este processo são passíveis de alteração três parâmetros, a quantidade de memória, a quantidade de processadores e o CAP de processamento.

Durante a execução dos testes de *scale-up*, os três parâmetros disponíveis foram testados garantindo a eficácia da funcionalidade. Os testes foram realizados em uma VM Ubuntu, configurada inicialmente para utilizar 1 processador, 256MB de memória e com o CAP em 30%. Todos os parâmetros foram alterados com sucesso e máquina virtual teve seus recursos computacionais realocados de forma instantânea.

Todo o processo de execução desta funcionalidade ocorreu em menos de 1 segundo, devido a integração do Controlador de Nodo como a API da ferramenta VirtualBox [18]. O Controlador de Nodo solicita acesso direto à máquina virtual e solicita a alteração de seus recursos. Embora os recursos computacionais da VM tenham sido realocados, o sistema operacional Ubuntu não soube interpretar as alterações de forma dinâmica, neste caso, exigindo uma reinicialização da máquina virtual para este identificar a alteração de *hardware*.

5.7 ESCALABILIDADE DE MÁQUINAS VIRTUAIS

A funcionalidade de *scale-out* permite redimensionar horizontalmente uma máquina virtual, possibilitando aumentar o poder de processamento de uma aplicação de acordo com a demanda de seu ambiente. No Educloud 2, esta também é uma nova funcionalidade oferecida aos usuários, tornando seu ambiente de nuvem privada ainda mais elástico e dinâmico.

Para a utilização desta nova funcionalidade, uma nova máquina virtual com suporte a *scale-out* deverá ser criada na nuvem. Entretanto, este processo de criação é um pouco diferente do apresentado da Seção 5.4, exigindo que ao criar uma máquina virtual com suporte a *scale-out* também seja criada uma nova instância de um balanceador de carga na nuvem. Este processo de criação diferenciado torna a tarefa mais lenta, pois realiza a duplicação de dois *templates* para disponibilizar as duas VMs na nuvem. Desta forma, este processo de criação foi validado e seus resultados são apresentados na Figura 27, onde é avaliado o desempenho obtido durante sua execução.

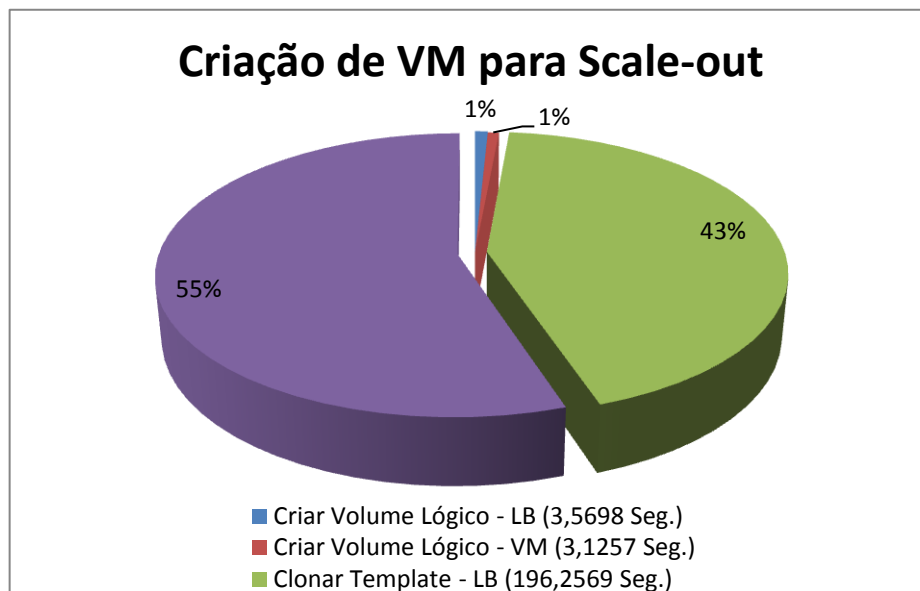


Figura 27: Resultados do teste para criação de VM para scale-out.

Após a realização do teste, constatou-se que o processo de criação de uma máquina virtual com suporte a *scale-out* é mais lento do que a criação de uma VM normal, conforme visto na seção 5.4. Isto acontece pela necessidade de criação de uma VM adicional, a qual será responsável pelo balanceamento de carga. Todo o

processo de execução foi realizado em torno de 453 segundos, caracterizando um aumento de 77%, comparando-o com a criação de uma máquina virtual normal.

Após a existência de uma máquina virtual com suporte a *scale-out* criada na nuvem, esta estará apta a ser inicializada e referenciada á seu balanceador de carga correspondente. O processo de inicialização e configuração da máquina virtual com suporte a *scale-out*, também foi avaliado e seus resultados são apresentados na Figura 28.

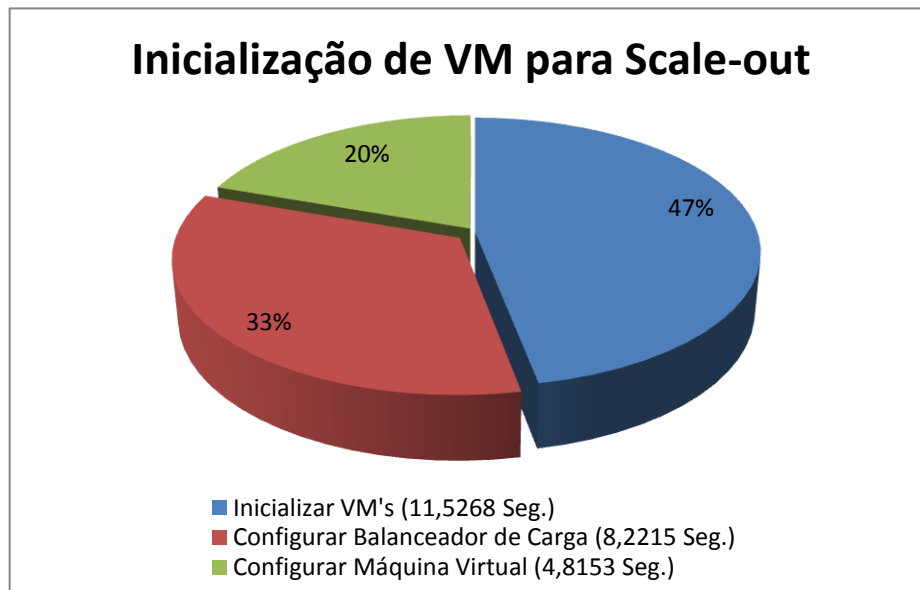


Figura 28: Resultados do teste para inicialização de VM para scale-out.

Este processo, além de ser responsável pela inicialização da máquina virtual, prepara o ambiente para realização do processo de *scale-out*. Desta forma, o balanceador de carga e a máquina de origem, recebem as configurações necessárias para suportar esta funcionalidade. Após a realização dos testes foi possível constatar que tempo total gasto para realização deste processo é de aproximadamente 25 segundos, representando um pequeno aumento comparado aos 6 segundos do processo de inicialização de uma máquina comum, visto que, a inicialização de VM para *scale-out* inicializa duas máquinas virtuais.

Após a máquina virtual criada, inicializada e com seu balanceador de carga configurado, a funcionalidade de *scale-out* ficará disponível para os usuários. Este processo é basicamente dividido em duas etapas, inicialmente será realizado o clone da VM de origem, e logo após esta será também adicionada ao balanceador

de carga existente. Para avaliar este processo foi realizado um teste de desempenho, o qual a Figura 29 apresenta seus resultados.

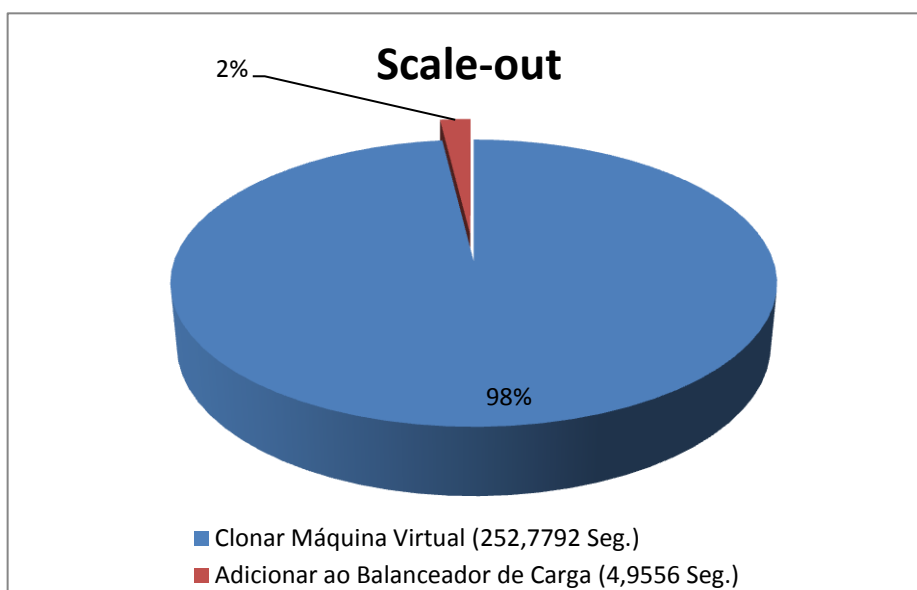


Figura 29: Resultados do teste para função de scale-out.

Após a realização do teste foi identificado que o processo responsável por clonar a máquina virtual é responsável por 98% do tempo de execução da tarefa. O tempo total de execução do processo de *scale-out* em uma máquina virtual foi identificado como de aproximadamente 258 segundos.

Com base em todos os testes e validações realizados foi montada a Tabela 1, que reúne todos os tempos encontrados na aplicação, além de realizar um comparativo com as funcionalidades presentes na primeira versão do Educloud.

Funcionalidades	Educloud 1	Educloud 2
Cadastro de Template	2 Seg.	5 Seg.
Criar VM Comum	490 Seg.	255 Seg.
Iniciar VM Comum	222 Seg.	6 Seg.
Criar VM Para Scale-out	N/A	453 Seg.
Iniciar VM Para Scale-out	N/A	25 Seg.
Scale-up	N/A	1 Seg.
Scale-out	N/A	258 Seg.

Tabela 1: Resultados de validação Educloud 2.

6 CONCLUSÃO E TRABALHOS FUTUROS

Como visto neste trabalho, a computação em nuvem envolve diversas tecnologias, que alinhadas e direcionadas, passam ao usuário a ideia de recursos infinitos. As soluções existentes flexibilizam aos usuários possibilidades outrora impraticáveis, permitindo que o poder computacional possa ser direcionado e dimensionado exatamente para a demanda.

A pesquisa inicial apontou diversos pontos, que atualmente caracterizam a computação em nuvem, além de torná-la mais atrativa, entre eles podemos citar a elasticidade como um dos principais. A possibilidade de aumentar e diminuir recursos livremente, sem restrições, torna a computação em nuvem muito mais atraente, inclusive estrategicamente para o meio empresarial.

A primeira versão do Educloud apresenta a ideia inicial de uma plataforma acadêmica de computação em nuvem, caracterizada por ser uma nuvem privada e sem a necessidade de uma infraestrutura de *hardware* dedicado. Ele representava os esforços na direção de consolidar uma plataforma de ensino de computação em nuvem no meio acadêmico. Entretanto faltava algo no Educloud, algo que o tornasse tão competitivo como as soluções pesquisadas, que permita ao meio acadêmico um cenário muito próximo do mercado. E esta foi a motivação para este trabalho.

O objetivo deste trabalho foi ambicioso, conseguir agregar o poder da elasticidade em uma plataforma acadêmica que não exige uma infraestrutura de hardware dedicada, sem remover funcionalidades ou alocar recursos de forma dedicada. Uma série de mudanças foram necessárias para conseguir direcionar a plataforma para este objetivo. Inicialmente a inclusão de um sistema de armazenamento centralizado, que, mesmo virtualizado, conseguiu atender a necessidade de alocar discos para as máquinas virtuais criadas. É importante salientar que na versão anterior eram utilizados discos em formato de arquivo, o que para o propósito inicial era adequado, mas com a inclusão do sistema de armazenamento centralizado, não pode mais ser utilizado, dessa forma, todo o processo de criação, início, parada e remoção de uma máquina virtual teve de ser refeito, a fim de garantir a consistência com a nova arquitetura incluída. A inclusão do armazenamento centralizado foi essencial para tornar o Educloud elástico.

A funcionalidade de *scale-up* de máquinas virtuais foi surpreendente por sua agilidade, mesmo com a máquina virtual em execução é possível redimensionar recursos de processamento e memória. Neste ponto, constatou-se que alguns sistemas operacionais não conseguem identificar novos recursos de processamento e memória até uma reinicialização, o que pode ser tratado em trabalhos futuros, seja incluindo alguma ferramenta que identifica este ponto no sistema que será disponibilizado ou até mesmo restringindo a funcionalidade em caso de sistema operacional sem suporte a alterações com a máquina ligada.

Quando pesquisado sobre elasticidade, um ponto que sempre chama à atenção é a inclusão de máquinas para suprir a necessidade de uma aplicação, o processo de *scale-out* foi um dos objetos dessa pesquisa, e talvez o maior dos desafios.

A plataforma de virtualização não contemplava nativamente uma forma de realizar esse processo de forma direta, pois exigia uma aplicação de balanceamento de carga para tal recurso. Foi necessário preparar uma aplicação de balanceamento e deixá-la disponível para o Educloud 2, o aspecto funcional foi levado em consideração neste ponto, pois para o usuário, não havia a necessidade de ter mais uma máquina virtual para gerenciar, o mesmo só precisaria se preocupar com a sua, e em caso de precisar alocar mais recursos, ai sim realizar o *scale-out*, clonando sua máquina atual para uma ou mais máquinas, deixando o balanceador direcionar as requisições para as máquinas que compunham à aplicação.

O processo de clonar também precisava ser transparente, pois para o usuário final, não deveria ter a necessidade de parar sua aplicação para poder realizar tal processo, cenário este que esta de acordo inclusive com as principais ferramentas pesquisadas.

Fazer com que todos esses pontos funcionassem de forma organizada, com uma arquitetura que pudesse ser estendida, reaproveitada e ainda agregasse para o meio acadêmico uma forma de aprendizado sobre ambientes de computação em nuvem elásticos compõem o objetivo deste trabalho, o qual foi realmente gratificante e empolgante. Nele foi possível colocar em prática todos os conhecimentos proporcionados por esta instituição além de exigir uma série de pesquisas adicionais e esforços inicialmente não previstos.

O Educloud em sua versão 2, atende as especificações propostas, deixando como trabalhos futuros questões como: a interligação entre nuvens privadas, uma forma mais simples de realizar atualizações no sistema de virtualização que minimize o impacto de desenvolvimento futuro e uma forma síncrona de criação de máquinas virtuais, além da funcionalidade de *Live Migration*, que embora pesquisada neste trabalho não foi implementada, mas é totalmente viável com a atual arquitetura utilizada. Estes pontos somariam ao Educloud como plataforma acadêmica de ensino de computação em nuvem.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] CARISSIMI, Alexandre. **Virtualização: da teoria a soluções**. Universidade Federal do Rio Grande do Sul. 2008.
- [2] ROSE, Robert. **Survey of System Virtualization Techniques**. 2004.
- [3] MATTOS, Diogo M. Ferrazani. **Virtualização: VMWare e Xen**. Universidade Federal do Rio de Janeiro. 2008.
- [4] MENASCÉ, Daniel A. **Virtualization: Concepts, applications and performance modeling**. George Mason University. 2005.
- [5] ALKIMIM, Gustavo P.; UCHÔA, Joaquim Quintero. **Uma solução de baixo custo para a Migração de Máquinas Virtuais**. Departamento de Ciência da Computação Universidade Federal de Lavras. 2009.
- [6] NELSON, Michel; LIM, Beng-Hong; HUTCHINS, Greg. **Fast Transparent Migration for Virtual Machines**. VMware Inc. 2005.
- [7] TAURION, Cezar. **Cloud Computing: Computação em Nuvem: Transformando o mundo da tecnologia da informação**. Editora Brasport. 2009.
- [8] RUSCHEL, Henrique; ZANOTTO, Mariana Susan; MOTA, Welton Costa. **Computação em Nuvem**. Pontifícia Universidade Católica do Paraná. 2008.
- [9] NIST. **A Definição NIST de Cloud Computing**. Disponível em: <http://www.nist.gov/index.html>. Acessado em: 01/09/2011.
- [10] SOTOMAYOR, Borja; MONTERO, Rubén Santiago; LLORENTE, Ignacio Martín; FOSTER, Ian. **Capacity Leasing in Cloud Systems using the OpenNebula Engine**. 2008.
- [11] VAQUERO, Luis M.; MERINO, LuisRodero; BUYYA, Rajkumar. **Dynamically Scaling Applications in the Cloud**. ACM SIGCOMM Computer Communication Review. 2011.
- [12] SALCHOW, Ken Jr. **Balanceamento de carga: A evolução para os Application Delivery Controller**. F5 Network Inc. 2009.

- [13] SCHICK, Shane. **Head in the Clouds? Welcome to the Future**. Disponível em: <http://www.theglobeandmail.com/news/technology/article137127.ece>. Acessado em: 01/09/2011.
- [14] EUCALYPTUS. **The Open Source Cloud Platform**. Disponível em: <http://open.eucalyptus.com/>. Acessado em: 01/09/2011.
- [15] AMAZON. **Amazon Elastic Compute Cloud (Amazon EC2)**. Disponível em: <http://aws.amazon.com/ec2/>. Acessado em: 01/09/2011.
- [16] SCHICK, Shane. **Head in the Clouds? Welcome to the Future**. Disponível em: <http://www.theglobeandmail.com/news/technology/article137127.ece>. Acessado em: 01/09/2011.
- [17] TREMPER, Diego; AMARAL, Robson. **EduCloud - Uma Proposta de Nuvem Privada para Ambientes Acadêmicos**. Trabalho de Conclusão da Faculdade de Informática. Pontifícia Universidade Católica do Rio Grande do Sul. 2011.
- [18] VIRTUALBOX. **Documentation**. Disponível em: <http://www.virtualbox.org>. Acessado em: 05/09/2011.
- [19] SOUSA, Flávio R. C.; MOREIRA, Leonardo O.; MACHADO, Javam C.. **Computação em Nuvem: Conceitos, Tecnologias, Aplicações e Desafios**. Universidade Federal do Ceará. 2009.
- [20] EUCALYPTUS. **Your Private Cloud**. Disponível em: <http://www.slideshare.net/AnnaRose1/your-private-cloud-checklist-5-steps-to-an-enterprise-grade-elastic-cloud>. Acessado em: 03/09/2011.
- [21] SAAYMANN, Joshua. **Internet Cloud Architect**. Disponível em: <http://cloudarchitect.posterous.com/>. Acessado em: 03/09/2011.
- [22] HOFF, Todd. **Eucalyptus - Build Your Own Private EC2 Cloud**. Disponível em: <http://highscalability.com/eucalyptus-build-your-own-private-ec2-cloud>. Acessado em: 01/09/2011.
- [23] OPENNEBULA. **Leading Cloud Management Solution**. Disponível em: <http://opennebula.org/>. Acessado em: 03/09/2011.

- [24] BADCOOCK, Charles. **OpenStack Adds Private Cloud-Building Features.** Disponível em: <http://www.informationweek.com/news/cloud-computing/infrastructure/231602029>. Acesso em 12/10/2011.
- [25] OPENSTACK. **Open source software for building private and public clouds.** Disponível em: <http://docs.openstack.org/diablo/openstack-compute/admin/os-compute-adminguide-trunk.pdf>. Acessado em: 03/09/2011.
- [26] HACKSPACE. **Rackspace Hosting.** Disponível em: <http://www.rackspace.com/>. Acessado em: 03/09/2011.
- [27] NASA. **National Aeronautics and Space Administration.** Disponível em: <http://www.nasa.gov/>. Acessado em: 03/09/2011.
- [28] VMWARE. **Openfiler.** Disponível em: <http://www.vmware.com/appliances/directory/344>. Acessado em: 10/10/2011.
- [29] CHILDERS, Bill. **OpenFiler: an Open-Source Network Storage Appliance.** Disponível em: <http://www.linuxjournal.com/magazine/openfiler-open-source-network-storage-appliance>. Acessado em: 10/10/2011.
- [30] LAVIGNE, Dru. **FreeNAS 8.0.1 User Guide.** iXSystems, 2011.
- [31] MACK, Joseph. **LVS-HOWTO.** AustinTek, 2011. Disponível em: <http://www.austintek.com/LVS/LVS-HOWTO/HOWTO/>. Acessado em: 23/10/2011.
- [32] OPENFILER. **Learn.** Disponível em: <http://www.openfiler.com/learn>. Acessado em: 20/10/2011.
- [33] MICROSOFT. **Dynamic Hardware Partitioning.** Disponível em: <http://technet.microsoft.com/en-us/library/cc755084%28WS.10%29.aspx>. Acessado em 14/11/2011.
- [34] DINESH, J Kumar. **Setup and Test VirtualBox Teleportation in Normal PC-Live Migration of Virtual Machines.** Disponível em: <http://www.sysprobs.com/setup-test-virtualbox-teleportation-normal-pc-live-migration-virtual-machines>. Acessado em: 14/11/2011.
- [35] NETBEANS. **NetBeans IDE.** Disponível em: <http://netbeans.org/>. Acessado em: 22/11/2011.

- [36] FREENAS. **FreeNAS - Storage for Open Source**. Disponível em: <http://www.freenas.org/>. Acessado em: 22/11/2011.
- [37] LVS. **LVS Documentation**. Disponível em: <http://www.linuxvirtualserver.org/Documents.html> Acessado em: 12/12/2011
- [38] RED HAT. **LVM Administrator's Guide**. RedHat Inc. Disponível em: <http://docs.redhat.com/> Acessado em: 13/04/2012.
- [39] CENTOS. The Community Enterprise Operating System. Disponível em: <http://www.centos.org/> Acessado em: 26/04/2012.
- [40] ORACLE. JavaServer Pages. Disponível em: <http://java.sun.com/products/jsp/docs.html> Acessado em: 20/06/2012.