

Arthur Jorge

**Modelagem e implementação em computação
em nuvem ao simulador de grades
computacionais iSPD**

São José do Rio Preto

2015

Arthur Jorge

Modelagem e implementação em computação em nuvem ao simulador de grades computacionais iSPD

Monografia apresentada ao Departamento de Ciências de Computação e Estatística do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, como parte dos requisitos necessários para aprovação na disciplina Projeto Final.

Universidade Estadual Paulista "Júlio de Mesquita Filho" (UNESP)

Instituto de Biociências, Letras e Ciências Exatas (IBILCE)

Prof^a. Dr^a. Renata Spolon Lobato.

São José do Rio Preto

2015

Jorge, Arthur.

Modelagem e implementação em computação em nuvem ao simulador de grades computacionais iSPD / Arthur Jorge. -- São José do Rio Preto, 2015

44 f. : il., tabs.

Orientador: Renata Spolon Lobato

Trabalho de conclusão de curso (bacharelado – Ciência da Computação) – Universidade Estadual Paulista “Júlio de Mesquita Filho”, Instituto de Biociências, Letras e Ciências Exatas

1. Computação. 2. Sistemas de computação virtual. 3. Computação em nuvem. 4. Computação em grade (Sistemas de computador) 5. Simulação (Computadores) I. Lobato, Renata Spolon. II. Universidade Estadual Paulista "Júlio de Mesquita Filho". Instituto de Biociências, Letras e Ciências Exatas. III. Título.

CDU – 681.3.025

Ficha catalográfica elaborada pela Biblioteca do IBILCE
UNESP - Câmpus de São José do Rio Preto

Arthur Jorge

Modelagem e implementação em computação em nuvem ao simulador de grades computacionais iSPD

Monografia apresentada ao Departamento de Ciências de Computação e Estatística do Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, como parte dos requisitos necessários para aprovação na disciplina Projeto Final.

Prof^a. Dr^a. Renata Spolon Lobato Arthur Jorge

Banca Examinadora:

Prof. Dr. Aleardo Manacero Jr.

Prof. Me. Geraldo Francisco Donega
Zafalon.

São José do Rio Preto

2015

Agradecimentos

Primeiramente gostaria de agradecer a minha mãe Isabel Cristina e a minha vó Maria pelo apoio nos estudos e nas horas que mais precisei.

A minha orientadora Profa. Dra. Renata Spolon Lobato, pela orientação nesse projeto e durante a graduação e pela paciência de me ajudar sempre que necessitei, agradeço também ao Prof. Dr. Aleardo Manacero pela ajuda, pelos conselhos durante toda graduação, além da amizade.

Aos meus amigos Guilherme Catalano, Giovane, Guilherme do Valle, Thulio Rodrigues, José Arthur, Walter, Vitor, Breno que me acompanharam desde da escola até o fim desta etapa.

Aos meus amigos de sala Vinicius (Sagat), Fernando Zerati (Marrone), André, Guilherme (Wiki), Rafael (Gaúcho), Felipe (Baby), Danilo (Fofão), Carlos (Master), e todos os outros que não citei aqui, que me apoiaram durante toda graduação.

Aos meus amigos do laboratório Mário (Jabuka), Gabriel Saraiva, João, Renan (Alboy), Fernanda, Lucas (Tesouro), Cassio, Leonardo, Victor Hugo (Pala), Lúcio, Gabriel Covello (Cokinha), pelo companhia durante todo o projeto e pelas noites de sinuca.

Agradeço especialmente aos meus amigos Diogo Tavares e Denison Menezes por trabalhar comigo neste projeto, enfrentando todos os problemas que apareceram, que se tornaram além de companheiros grandes amigos.

Por fim, gostaria de agradecer à UNESP pelo apoio e financiamento concedidos juntamente com o CNPq com bolsas PIBIC/Reitoria, à FAPESP que auxiliou o projeto, fornecendo equipamentos ao laboratório do GSPD.

"Nunca se esqueça de quem você é, porque é certo que o mundo não se lembrará. Faça disso sua força. Assim, não poderá ser nunca a sua fraqueza. Arme-se com esta lembrança, e ela nunca poderá ser usada para lhe magoar"

-George R.R. Martin

Resumo

A utilização de sistemas distribuídos conhecidos como computação em nuvem tem aumentando gradativamente nos últimos anos. A simulação desses sistemas tão utilizados é uma alternativa barata e eficiente para avaliação de desempenho, devido ao custo temporal e financeiro. Este trabalho tem como objetivo o estudo de computação em nuvem para a inclusão de modelagem e simulação em nuvem ao iSPD (*iconic Simulator of Parallel and Distributed Systems*) de forma que seja possível a simulação de forma intuitiva e eficiente através da interface icônica presente na ferramenta, permitindo que um usuário sem conhecimento de programação consiga fazer um modelo de simulação.

Palavras-chaves: computação em nuvem. modelagem. simulação.

Abstract

The use of distributed systems known as cloud computing has gradually grown in the last years. The simulation of systems so used is a cheap and efficient alternative to performance evaluation in terms of time and money. The aim of this work is to study cloud computing so that modelling and cloud simulation can be included to iSPD (*iconic Simulator of Parallel and Distributed Systems*) in such a way that intuitive and efficient simulation is possible through the tool's iconic interface, allowing that a user without programming knowledge can make a simulation model.

Key-words: cloud computing. modeling. simulation.

Lista de ilustrações

Figura 1 – Representação dos módulos do iSPD	22
Figura 2 – Representação do DTD das características que são salvas	26
Figura 3 – Janela de escolha do modelo	27
Figura 4 – Interface icônica do iSPD	28
Figura 5 – Janela de configuração das máquinas virtuais	29
Figura 6 – Janela do gerenciador de escalonadores	30
Figura 7 – Diagrama das classes de serviço do motor e suas funcionalidades	30
Figura 8 – Janela de gerenciamento dos alocadores	31
Figura 9 – Janela de métricas globais com as métricas de custo	33
Figura 10 – Janela de configuração das tarefas	34
Figura 11 – Janela do modelo gerado para os testes	35
Figura 12 – Gráfico dos testes realizados	36
Figura 13 – Gráfico de processamento no iSPD	37
Figura 14 – Métricas globais no iSPD	37
Figura 15 – Métricas de recursos no iSPD	38
Figura 16 – Log gerado no CloudSim	39

Lista de tabelas

Tabela 1 – Tabela comparativa entre simuladores.	21
Tabela 2 – Tabela de dados dos testes realizados com o tempo de simulação em segundos.	36

Lista de abreviaturas e siglas

BoT	<i>Bag of Tasks</i>
DAG	<i>Directional Acyclic Graph</i>
XML	<i>Extensible Markup Language</i>
Flops	<i>FLoating-point Operations Per Second</i>

Sumário

1	INTRODUÇÃO	12
1.1	Motivação	12
1.2	Objetivo	13
1.3	Organização do texto	13
2	REVISÃO BIBLIOGRÁFICA	14
2.1	Computação em nuvem: Definições e conceitos	14
2.1.1	Origem do conceito de computação em nuvem	14
2.1.2	Definição de computação em nuvem	14
2.1.3	Características de Computação em Nuvem e suas principais tecnologias	15
2.1.4	Gerenciamento da carga de trabalho	16
2.1.5	Classes de serviços de computação em nuvem	17
2.2	Simuladores de computação em nuvem	18
2.2.1	CloudSim	18
2.2.2	iCanCloud	19
2.2.3	GreenCloud	20
2.2.4	Comparativo entre os simuladores	20
2.3	iSPD	21
2.4	Considerações finais	23
3	MODELAGEM E IMPLEMENTAÇÃO EM COMPUTAÇÃO EM NUVEM	24
3.1	Modelagem da classe de serviço para o iSPD	24
3.2	Caracterização de tarefas em nuvem	24
3.2.1	Implementação das interfaces para a simulação de computação em nuvem	27
3.3	Motor de simulação	28
3.3.1	Implementação do motor de simulação	29
3.3.2	Política de alocação das máquinas virtuais	31
3.4	Considerações finais	31
4	RESULTADOS OBTIDOS	33
4.1	Procedimento para realização dos testes	33
4.2	Resultados dos testes realizados	34
4.2.1	Testes comparativos com o CloudSim	34
4.3	Considerações Finais	39

5	CONCLUSÕES	40
5.1	Dificuldades	40
5.2	Próximas ações	40
5.3	Publicações	41
	Referências	42

1 Introdução

Nos últimos anos a utilização de computação em nuvem tem aumentado gradativamente. Esta tecnologia tem o objetivo de disponibilizar recursos computacionais sob demanda através da infraestrutura da internet, utilizando um modelo utilitário em que se paga pelo tempo e pela quantidade de recursos que são utilizados.

Embora a utilização desses sistemas esteja cada vez mais frequente, a análise de desempenho através de *benchmarks* é inviável devido ao custo temporal, pois dependendo do tamanho do sistema, o tempo de um teste pode ser altamente elevado, e do custo financeiro, pois o custo de um sistema com muitas máquinas para a realização de testes, chega a ser impraticável.

Portanto a simulação desses sistemas surge como uma solução barata e eficiente para avaliação de desempenho.

O iSPD (*iconic Simulator of Parallel and Distributed Systems*), simulador de grades computacionais do GSPD (Grupo de Sistemas Paralelos e Distribuídos), tem como objetivo facilitar a utilização de simuladores, fornecendo ao usuário uma interface icônica para produção de modelos de simulação de maneira fácil e intuitiva, sem a necessidade de um conhecimento amplo de programação. O objetivo é adicionar simulação em nuvem ao iSPD, aproveitando a interface e o motor presente na ferramenta.

O presente projeto está inserido no contexto do projeto “Desenvolvimento de Plataforma de Simulação de Grades Flexível e Orientada à Avaliação de Desempenho: Módulo de Interpretação de Modelos Externos e Módulo de Exportação de Modelos Icônicos”, que conta com auxílio de bolsa de iniciação científica PIBIC/Reitoria.

1.1 Motivação

Na computação em nuvem a avaliação de desempenho é algo extremamente importante, tanto para provedores que precisam decidir qual é a melhor política de alocação e escalonamento para seu sistema, quanto para o cliente que precisa avaliar o desempenho para selecionar a melhor configuração dos recursos para sua aplicação. Outro fator é a diminuição do custo de implementação, pois com a simulação não há necessidade de ter todo o sistema físico para ter se conhecimento do seu desempenho. A motivação desse trabalho é facilitar a simulação em nuvem usando como meio a interface icônica presente no iSPD.

1.2 Objetivo

O objetivo desse projeto é realizar um estudo em computação em nuvem, para adicionar ao simulador de grades computacionais iSPD, a opção de simulação dos ambientes em nuvem, implementando as classes de serviços e adicionando políticas de alocação de máquinas virtuais no motor de simulação.

Fornecendo ao usuário uma interface icônica para produção de modelos de simulação de maneira fácil e intuitiva, sem a necessidade de um conhecimento amplo de programação.

1.3 Organização do texto

Este trabalho está estruturado em 5 capítulos. Após esta introdução, no capítulo 2 se faz uma visão geral da ferramenta iSPD, seguido de uma fundamentação teórica sobre computação em nuvem e simuladores de computação em nuvem. No capítulo 3, apresenta-se a especificação da caracterização e modelagem dos serviços de computação em nuvem para o iSPD e o funcionamento do motor de simulação, e suas políticas. No capítulo 4 apresentam-se os resultados dos testes realizados comparando o iSPD com o simulador CloudSim. Por fim, no capítulo 5 apresentam-se as conclusões sobre este trabalho e as direções futuras.

2 Revisão bibliográfica

Neste capítulo é apresentado uma fundamentação teórica a respeito dos tópicos abordados neste trabalho, dividindo-se em três partes. Na primeira parte apresenta-se conceitos e serviços de computação em nuvem. Na segunda parte apresenta-se os principais simuladores de computação em nuvem e uma comparação entre eles, e na terceira parte uma introdução ao simulador de grades computacionais iSPD em que se apresenta suas principais funções e características.

2.1 Computação em nuvem: Definições e conceitos

Nesta seção define-se o conceito de computação em nuvem, sua origem, principais características e, por fim, suas classes de serviço.

2.1.1 Origem do conceito de computação em nuvem

O primeiro uso conceito de computação em nuvem é atribuído a John McCarthy, em 1961. Ele propôs a ideia de que a computação deveria ser utilizada como um serviço de utilidade pública, assim como água e energia, em que os usuários só pagam pelo que usam, como ocorre no modelo atual de computação em nuvem. (RITTINGHOUSE; RANSOME, 2009).

Embora a ideia de computação utilitária tenha se tornado popular nos anos seguintes, ela não foi desenvolvida devido a limitação da tecnologia existente à época. Somente em 1999 esta tecnologia foi desenvolvida com o surgimento da Salesforce.com (SALESFORCE, 2015), primeira empresa a disponibilizar aplicações na internet. Com o sucesso dessa empresa, grandes empresas da área se interessaram no desenvolvimento da tecnologia, como a Amazon (AMAZON, 2015), Google (GOOGLE, 2012) e Microsoft (MICROSOFT, 2013).

2.1.2 Definição de computação em nuvem

O termo computação em nuvem é recente, portanto não há uma definição formal padrão. No entanto, diversos autores têm buscado uma definição, algumas dessas definições estão destacadas no parágrafos a seguir.

Uma das definições de computação em nuvem de acordo com (VAQUERO et al., 2008) é “Computação em Nuvem, como o próprio nome sugere, engloba as chamadas nuvens, que são ambientes que possuem recursos (*hardware*, plataformas de desenvolvimento e/ou serviços) acessados virtualmente e de fácil utilização. Esses recursos, devido à virtualização, podem ser reconfigurados dinamicamente, de modo a se ajustar em uma determinada variável, permitindo,

assim, um uso otimizado dos recursos. Esses ambientes são, em geral, explorados através de um modelo *pay-per-use*.”

Segundo (BUYYA; BROBERG; GOSCINSKI, 2011) computação em nuvem é um tipo de sistema paralelo e distribuído que consiste em uma coleção de computadores interconectados e virtualizados, que são dinamicamente provisionados e apresentados como um ou mais recursos computacionais unificados.

Outra definição abordada é de (ARMBRUST et al., 2009) que define como "A computação em nuvem é um conjunto de serviços de rede ativados, proporcionando escalabilidade, qualidade de serviço, infraestrutura barata de computação sob demanda e que pode ser acessada de uma forma simples e pervasiva e enumera como principais características "(i) A ilusão de recursos computacionais infinitos; (ii) Eliminação na necessidade de submissão para um *front-end* pelos usuários da nuvem (iii) A capacidade de pagar pelo uso como necessário".

Em uma visão governamental, o NIST (*National Institute of Standards and Technology*) define computação em nuvem como "... um modelo de negócio *pay-per-use* (pago por utilização) que permite, convenientemente, o acesso, através de rede, sob demanda, a um arranjo compartilhado de recursos computacionais configuráveis (rede, servidores, armazenamento, aplicações, serviços) que podem ser rapidamente providos e liberados com o mínimo esforço de gerenciamento ou interação do provedor de serviço" (MELL; GRANCE, 2011).

Embora existam diferentes definições, todas possuem características em comum, essas características serão abordadas na próxima subseção.

2.1.3 Características de Computação em Nuvem e suas principais tecnologias

Nesta seção serão abordados as principais características de computação em nuvem, descritas a seguir:

- **Autoatendimento:** Consiste em oferecer ao usuário uma interface que seja capaz de configurar, requisitar, pagar e utilizar recursos contratados sem a necessidade de um operador humano.(BUYYA; BROBERG; GOSCINSKI, 2011);
- **Medição e pagamento por utilização:** Consiste em implementar soluções de pagamento, cobrança e monitoração dos recursos contratados, de forma que esta cobrança seja transparente ao usuário, por exemplo, o custo referente a uma hora de um tipo de equipamento. (BUYYA; BROBERG; GOSCINSKI, 2011) (MELL; GRANCE, 2011);
- **Elasticidade rápida:** Recursos podem ser adquiridos de forma rápida e elástica, transmitindo ao usuário a ideia de recursos infinitos, disponíveis sob demanda. O serviço deve fornecer recursos em qualquer quantidade e a qualquer momento, cobrindo a demanda solicitada pelo sistema hospedado. A elasticidade é importante tanto para aumentar a

quantidade de recursos disponíveis, quanto para reduzi-la quando já não é mais necessário. (BUYYA; BROBERG; GOSCINSKI, 2011);

- **Estrutura configurável:** Consiste em oferecer ao usuário um sistema facilmente configurável, pois uma estrutura de computação em nuvem pode receber uma grande demanda de usuários que necessitam de serviços distintos. (BUYYA; BROBERG; GOSCINSKI, 2011);
- **Amplo acesso à rede:** Recursos são disponibilizados por meio da rede e acessados através de dispositivos de acesso (*tablets, notebooks, smartphones, etc.*). A interface de acesso à nuvem não obriga que os usuários a mudar em suas condições e ambientes de trabalho, como por exemplo, linguagens de programação e sistemas operacionais. (MELL; GRANCE, 2011).

Uma das tecnologias responsáveis pelo crescimento da computação é a tecnologia de internet, com a criação dos *web services* (WS) que são amplamente utilizados hoje por grandes empresas como a Amazon. Outra tecnologia são as arquiteturas orientadas a serviços (*Service-Oriented Architectures* - SOA), que permite a comunicação de aplicações através da internet.

Outra tecnologia que possui grande importância no crescimento da utilização de nuvem é a virtualização de *hardware*, que é um conceito chave de computação em nuvem. Essa tecnologia consiste em simplificar a execução de múltiplos sistemas operacionais com pilhas de software heterogêneas, executando em uma mesma plataforma física. Isso acontece através da inserção de uma nova camada entre o sistema físico e os sistemas virtualizados, que são chamados de máquinas virtuais ou VM (*Virtual Machine*). A camada de virtualização tem o objetivo de hospedar e gerenciar todas as máquinas virtuais, através do monitor de máquina virtual (*Virtual Machine Monitor* - VMM) ou *hypervisor*. Como exemplo de VMMs destaca-se o VMware ESXi (VMWARE, 2013), e o Xen (XEN, 2013).

2.1.4 Gerenciamento da carga de trabalho

Segundo (BUYYA; BROBERG; GOSCINSKI, 2011) destaca-se três capacidades básicas em relação ao gerenciamento de carga de trabalho, em um sistema virtualizado (THOMSEN, 2002), descritas a seguir:

- **Isolamento:** Esta capacidade garante que todas as instruções de programa fiquem confinadas a VM, ou seja, os erros relacionados a esse programa serão restritos àquela VM, o que garante uma maior segurança e confiabilidade. Um ponto importante desse isolamento é que o desempenho de uma VM não é afetado por outra;
- **Consolidação:** Esta capacidade consiste em hospedar vários servidores virtualizados em uma mesma plataforma física, assim essa capacidade permite uma melhor utilização dos

recursos de servidores que seriam subutilizados, como por exemplo de servidores dedicados. Outra vantagem é a possibilidade de hospedar um servidor antigo e um atualizado, garantindo assim uma maior compatibilidade de *software*;

- **Migração:** Esta capacidade consiste em salvar o estado atual de uma VM, como o disco utilizado, os arquivos configurados, memória RAM, entre outros. Após salvar esse estado, ele é suspenso, serializado e migrado para outra plataforma, em que será restaurado com todas as características salvas. Com essa capacidade há uma maior facilidade na recuperação de falhas, na manutenção do sistema e principalmente no balanceamento de carga de trabalho.

2.1.5 Classes de serviços de computação em nuvem

A computação em nuvem se divide tradicionalmente em três classes de serviço, que são descritas a seguir:

- **Infrastructure-as-a-Service (IaaS):** Esta classe de serviço oferece ao consumidor toda infraestrutura de *hardware*, como por exemplo, processamento, armazenamento, memória, rede. Para que se possa instalar uma estrutura de software arbitrária, como sistemas operacionais e aplicações.

O IaaS se baseia na virtualização da infraestrutura que receberá o serviço, este serviço permite que o usuário realize várias atividades sobre a infraestrutura, como personalizar a pilha de *software*, iniciar, parar, adicionar novos programas e até novos discos virtuais.

As ofertas de soluções, os custos, a alta disponibilidade e o nível de profissionalização dos provedores são os principais fatores da adoção do IaaS

Como exemplo de um serviço IaaS pode-se destacar a *Amazon Web Services* (AMAZON, 2015), que oferece máquinas virtuais configuráveis, de maneira parecida com servidores físicos. (MELL; GRANCE, 2011).

- **Platform-as-a-Service (PaaS):** Esta classe de serviço tem o objetivo de disponibilizar ao usuário plataformas de desenvolvimento e hospedagem de aplicações, oferecendo um alto nível de abstração, proporcionando uma plataforma mais robusta e flexível para a utilização de muitos recursos. É possível a utilização de *softwares* de maneira mais flexível, sendo possível desenvolver suas próprias aplicações baseadas em alguma tecnologia e utilizar a infraestrutura necessária, e o mais importante, adequada a aplicação desenvolvida.

Assim, esta classe de serviço oferece um ambiente no qual programadores podem desenvolver aplicações sem ter conhecimento da infraestrutura, linguagem e biblioteca em que ela será executada, permitindo assim que aplicações sejam facilmente implantadas, devido a abstração do serviço.

Como exemplo de um serviço PaaS pode-se destacar o *Google AppEngine* (GOOGLE, 2013a), que é um ambiente escalável em que desenvolvedores podem implementar e hospedar aplicações web e o *Microsoft Azure* (MICROSOFT, 2015), que possui um ambiente bem flexível para desenvolvimento.

- **Software-as-a-Service (SaaS):** Esta classe de serviço oferece através de uma interface web acessível de um navegador ou dispositivos móveis, aplicações como por exemplo um processador de texto ou planilhas, editores de imagens, vídeo, entre outros. Nesta classe de serviço o usuário compra o serviço de um produto, mas não o produto. É um modelo flexível, que lhe permite controlar o que você necessita, pagando apenas pelo que utiliza em determinado período ou situação pontual.

Como exemplo de um serviço SaaS pode-se destacar o *Google Docs* (GOOGLE, 2013b) que oferece um grande variedade de aplicações, como por exemplo editores de texto e de imagens, onde se utiliza o serviço do produto, sem a necessidade de ter o produto.

2.2 Simuladores de computação em nuvem

Nesta seção serão detalhadas as principais funcionalidades e características dos simuladores de computação em nuvem.

Atualmente existe uma quantidade relativamente pequena de simuladores para computação em nuvem, alguns apresentam documentação escassa ou inexistente. Nesta seção serão abordados três simuladores, são eles: o CloudSim (BUYYA; RANJAN; CALHEIROS, 2009), iCanCloud (CASTANE; NUNEZ; CARRETERO, 2012) e o GreenCloud (KLIASOVICH et al., 2010), que apresentam maior popularidade e documentação estabelecida. A descrição de cada um deles é apresentada nas próximas subseções.

2.2.1 CloudSim

A ferramenta CloudSim (CLOUDS, 2013), é desenvolvida pelo grupo CLOUDS (*The Cloud Computing and Distributed Systems*) da Universidade de Melborn, na Austrália, sob a coordenação de Rajkumar Buyya. O principal objetivo do CloudSim é permitir a modelagem e simulação de ambientes de computação em nuvem, sem a necessidade do usuário se preocupar com os detalhes de baixo nível.

O CloudSim se encontra na versão 3.03 e sua linguagem de desenvolvimento é o Java, que exige que o usuário possua conhecimento prévio em programação para tal linguagem e consequentemente conheça o paradigma de programação orientado a objetos.

A principal desvantagem desta ferramenta é a ausência de interface gráfica, portanto todo processo de modelagem e simulação é feito através de programação.

As principais funcionalidades do CloudSim são:

- Suporte para modelagem e simulação de grande escala de dados de centros de computação em nuvem;
 - Suporte para modelagem e simulação dos servidores virtualizados, com políticas personalizáveis para proporcionar recursos do host para máquinas virtuais;
 - Suporte para modelagem e simulação de recursos computacionais com consumo consciente de energia;
 - Suporte para modelagem e simulação de topologias de rede de dados e aplicações de troca de mensagens;
 - Suporte para inserção dinâmica de elementos de simulação, parada e recuperação de estado de simulação;
 - Suporte para modelagem e simulação de nuvens;
 - Suporte para políticas definidas pelo usuário para alocação de máquinas para máquinas virtuais e políticas para alocação de recursos de máquinas físicas para máquinas virtuais;
- Destaca-se que o CloudSim é base de vários projetos que estendem suas funcionalidades como RealCloudSim, CloudAnalist, CloudSimEx, entre outros (CLOUDS, 2013).

2.2.2 iCanCloud

O simulador iCanCloud (ARCOS, 2013) é desenvolvido pelo grupo ARCOS (*Grupo de Arquitectura de Computadores, Comunicaciones y Sistemas*) da universidade Carlos III de Madrid, na Espanha, sob coordenação de Jesús Carretero Pérez.

O iCanCloud é desenvolvido através dos frameworks OMNeT++ e INET (OMNET++, 2013) e utiliza-se dos mesmos para modelagem. Apresenta uma interface gráfica própria, mas necessita que o usuário tenha conhecimento prévio das ferramentas utilizadas.

O objetivo do ICanCloud é prever os *trade-offs* entre custo e desempenho de um determinado conjunto de aplicativos executados em um hardware específico, e em seguida, fornecer aos usuários informações úteis sobre esses custos. Tendo o foco na simulação da classe de serviços de IaaS.

As características mais notáveis do ICanCloud é fornecer métodos para a obtenção do consumo de energia de cada componente de hardware em sistemas de computação em nuvem. E permitir que o usuário seja capaz de projetar políticas de provisionamento de recursos modelo para sistemas de nuvem para equilibrar os *trade-offs* entre desempenho e consumo de energia. Outra característica importante é a personalização das VMs para simular sistemas *uni-core* ou *multi-core*.

2.2.3 GreenCloud

O simulador GreenCloud (GREENCLOUD, 2013) é desenvolvido pela Universidade de Luxemburgo, sob a coordenação de Dzmitry Kliazovich.

O GreenCloud consiste em uma extensão do simulador de rede NS2, possui cerca de 80 por cento de código implementado em C++ e os restantes 20 por cento na forma de linguagem de comando (*TCL - Tool comand language*) *scripts* e encontra-se na versão 2.0.3. O simulador apresenta uma interface gráfica, mas esta não é nativa, e deve ser instalada como uma extensão.

O objetivo do GreenCloud é completamente voltado para o desempenho energético. Ele pode ser utilizado para desenvolver novas soluções em monitoramento, alocação de recursos, carga de trabalho de programação, e até mesmo otimização de protocolos de comunicação e infraestruturas de rede. Pode simular datacenters existentes, bem como ajudar a projetar futuras instalações deles.(KLIAZOVICH et al., 2010).

Enfim o Greencloud é um simulador voltado para projetos de computação que seguem os conceitos de computação verde (BELOGLAZOV; BUYYA, 2012), termo empregado para designar projeto com foco em consumo consciente de energia e recursos, com o propósito de reduzir o uso de energia e, conseqüentemente, os custos operacionais.

2.2.4 Comparativo entre os simuladores

Nesta subseção será realizado um comparativo entre os simuladores apresentados anteriormente, destacando os pontos positivos e negativos na utilização de cada um deles. Os pontos que são considerados nesta comparação serão apresentados a seguir:

- **Framework base de desenvolvimento do simulador**, baseia-se no desempenho e na dificuldade de utilização da ferramenta.
- **Linguagem de programação**, baseia-se principalmente no desempenho e portabilidade da linguagem de programação utilizada.
- **Licença de utilização**, reflete na disponibilidade para uso e disponibilidade de seu código-fonte.
- **Suporte à interface gráfica**, baseia-se na disponibilidade de interface gráfica na ferramenta, que simplifique e melhore a experiência final do usuário, principalmente na modelagem.
- **Capacidade de simular virtualização**, baseia-se na ferramenta que possui a capacidade de simular virtualização, que reflete em uma melhor caracterização da arquitetura de nuvem.

Para melhor entendimento das diferenças de cada ferramenta apresenta-se a tabela 1 com o comparativo entre os simuladores estudados (SILVA, 2013) (SILVA, 2015).

	CloudSim	iCanCloud	GreenCloud
Framework	—	OMNeT++, MPI	NS2
Linguagem de programação	Java	C++	C++, Tlc
Licença de uso	<i>Open Source</i>	<i>Open Source</i>	<i>Open Source</i>
Suporte a interface gráfica	Limitado (através do uso de extensões)	Existente	Limitado (Através do uso de uma extensão)
Suporte à virtualização	Sim	Sim	Não

Tabela 1 – Tabela comparativa entre simuladores.

Analisando-se em relação ao *framework* de desenvolvimento, os simuladores iCanCloud e GreenCloud possuem desvantagem, com ponto de vista ao usuário, devido a necessidade de possuir conhecimentos adicionais sobre os *frameworks* para utilizá-los.

Quanto a linguagem de programação utilizada, destaca-se o simulador CloudSim em relação aos outros, por utilizar no desenvolvimento a linguagem de programação Java, garantindo maior portabilidade da ferramenta.

Com base na interface gráfica, destaca-se o simulador iCanCloud, que possui uma interface gráfica de auxílio própria, enquanto CloudSim necessita conhecimento de programação para modelagem e o GreenCloud necessita a instalação de uma expansão.

Por fim, analisando a capacidade de simular virtualização, destaca-se os simuladores IcanCloud e CloudSim, pois o GreenCloud não apresenta a modelagem desta característica.

2.3 iSPD

O iSPD (MANACERO et al., 2012) é um simulador de grades computacionais desenvolvido pelo GSPD (Grupo de Sistemas Paralelos e Distribuídos) da Unesp. Foi desenvolvido utilizando a linguagem de programação Java, devido a facilidade em elaborar uma interface icônica na mesma, e da grande capacidade de portabilidade das aplicações.

O objetivo do iSPD é facilitar o uso de simuladores, tornando o processo de modelagem computacional uma etapa simples e intuitiva através de sua interface icônica, permitindo que um usuário que não possua amplo conhecimento em programação, consiga fazer um modelo de simulação sem a necessidade de utilizar linhas de código.

A arquitetura da ferramenta é dividida em cinco módulos, cuja descrição de cada um deles é dada a seguir:

- **Interface icônica:** Módulo responsável pela modelagem computacional e pela configuração da carga de trabalho. A modelagem é feita através de ícones que representam os elementos que constituem a grade a ser simulada.
- **Interpretador de Modelos internos:** Módulo que tem a função de interpretar o modelo icônico gerado, convertendo-o para um modelo simulável, utilizado posteriormente pelo motor de simulação.
- **Interpretador de modelos externos e exportador de modelos internos:** Interpretador de modelos externos têm a função de interpretar modelos gerados por outros simuladores para o modelo icônico do iSPD, atualmente interpreta os modelos dos simuladores GridSim e SimGrid. O exportador de modelos internos tem a função de exportar modelos icônicos do iSPD, gerando modelos para o simulador GridSim.
- **Gerador e Gerenciador de Escalonadores:** Módulos que possuem a função de gerenciar os escalonadores disponíveis e facilitar a criação de novas políticas de escalonamento de maneira simplificada.
- **Motor de simulação:** O motor é o módulo que se realiza a simulação propriamente dita, nele é interpretado o modelo de grade convertendo-o em um modelo de filas para que seja realizado a simulação, no final da simulação é exibido ao usuário o resultado através de uma interface.

Para um melhor entendimento, a forma como os módulos apresentados interagem está representada na figura 1.

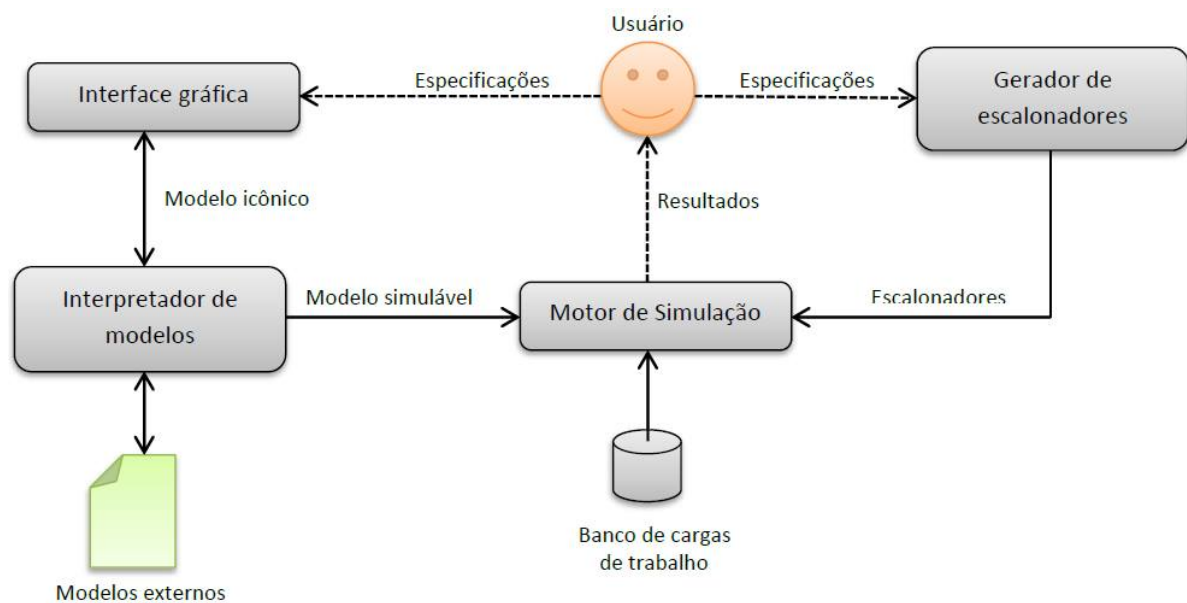


Figura 1 – Representação dos módulos do iSPD

2.4 Considerações finais

Neste capítulo apresentou-se primeiramente um estudo sobre computação em nuvem, apresentando suas origens, conceitos e principais características. Apresentou-se também um estudo dos principais simuladores de computação em nuvem, realizando no final um comparativo entre eles. No fim apresentou-se o iSPD, simulador no qual será feita a expansão para simulação em nuvem. No capítulo 3 serão apresentados as principais adições na ferramenta iSPD para modelagem e simulação em nuvem.

3 Modelagem e implementação em computação em nuvem

Neste capítulo, especifica-se a modelagem da classe de serviço que será simulada, além de apresentar o funcionamento do motor de simulação e suas políticas de escalonamento e de alocação das máquinas virtuais.

3.1 Modelagem da classe de serviço para o iSPD

Nesta seção aborda-se a classe de serviço que modelada e como foi adicionada essa modelagem na interface icônica no iSPD.

Optou-se por simular a classe de serviço IaaS (*Infrastructure-as-a-Service*). Nesta classe de serviço deve-se configurar a infraestrutura que será virtualizada, como as máquinas virtuais na janela de configuração das máquinas virtuais, configurando suas principais características, como a quantidade de núcleos de processamento alocado, e quantidade de memória. As políticas de escalonamento das tarefas e as políticas de alocação das máquinas virtuais no VMM (*Virtual Machine Monitor*).

O VMM é uma máquina mestre que irá gerenciar o escalonamento das tarefas que serão executadas e a alocação das máquinas virtuais nas máquinas físicas, seguindo as políticas de escalonamento e de alocação escolhidas.

3.2 Caracterização de tarefas em nuvem

Antes de se iniciar a modelagem e a simulação em nuvem deve-se primeiramente selecionar a caracterização adequada dos recursos e das aplicações. A seguir são apresentadas as principais características que compõem cada componente presente na simulação em nuvem.

Os custos presentes nas características, são a quantidade de cada recurso da máquina que foi alocado pelas máquinas virtuais na simulação.

Caracterização dos recursos:

- Recurso Máquina (*Computing Unit*)

Label (ID)

Proprietário

Número de núcleos de processamento

Poder computacional por núcleo (GFlops)

Memória Primária - *primary storage* (GB)

Memória Secundária - *secondary storage* (GB)

Custo de Processamento

Custo de Memória

Custo de Disco

Recurso ou VMM (Caso a máquina seja um VMM, ela possuirá uma política de alocação de máquinas virtuais e de escalonamento)

- *Cluster* (Conjunto de máquinas homogêneo)

Label (*Cluster ID*)

Proprietário

Número de elementos (máquinas)

Número de núcleos (*cores*) por elemento

Quantidade de RAM disponível por elemento

Quantidade de Disco (GB) disponível por elemento

Banda de comunicação entre os nós

Latência de transmissão entre os nós

Políticas de escalonamento e alocação da VMM

- Enlace e Internet

Label (ID)

Banda de passagem (Mbps)

Latência de transmissão (segundos)

fator de carga (porcentagem utilizada)

- Máquina Virtual (VM)

Label (ID)

ID da máquina hospedeira

ID do usuário contratante

Processamento alocado (número de núcleos)

Quantidade de memória alocada (GB)

Quantidade de disco alocado (GB)

Sistema operacional instalado

O iSPD salva os modelos icônicos gerados em arquivos XML. Assim quando o usuário construir um modelo icônico e salvá-lo, será gerado um arquivo XML, em um formato IMS (*Iconic Model of Simulation*) que contém todas as informações sobre os ícones modelados, a carga de trabalho e suas características.

A validação dos modelos é realizada através de um arquivo DTD (*Document Type Definition*), verificando-se todas as informações disponíveis e a integridade dessas informações. A função do DTD é definir o padrão que deve ser seguido na simulação em nuvem. Adicionou-se ao DTD as características citadas anteriormente para ser possível a validação dos modelos gerados.

Na figura 2 apresenta-se um trecho do DTD que especifica o elemento "*characteristic*", que contém as características dos elementos modelados.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<!DOCTYPE system SYSTEM "iSPD.dtd">
<system version="2.2">
  <owner id="user1"/>
  <machine id="mac4" load="0.0" owner="user1" power="30000.0">
    <position x="189" y="183"/>
    <icon_id global="4" local="4"/>
    <characteristic>
      <process number="4" power="30000.0"/>
      <memory size="4.0"/>
      <hard_disk size="100.0"/>
      <cost cost_disk="1.0" cost_mem="3.0" cost_proc="5.0"/>
    </characteristic>
  </machine>
  <machine id="mac6" load="0.0" owner="user1" power="30000.0">
    <position x="307" y="142"/>
    <icon_id global="6" local="6"/>
    <characteristic>
      <process number="4" power="30000.0"/>
      <memory size="4.0"/>
      <hard_disk size="100.0"/>
      <cost cost_disk="1.0" cost_mem="3.0" cost_proc="5.0"/>
    </characteristic>
  </machine>
  <machine id="mac5" load="0.0" owner="user1" power="30000.0">
    <position x="255" y="183"/>
    <icon_id global="5" local="5"/>
    <characteristic>
      <process number="4" power="30000.0"/>
      <memory size="4.0"/>
      <hard_disk size="100.0"/>
      <cost cost_disk="1.0" cost_mem="3.0" cost_proc="5.0"/>
    </characteristic>
  </machine>
  <machine id="mac1" load="0.0" owner="user1" power="30000.0">
```

Figura 2 – Representação do DTD das características que são salvas

3.2.1 Implementação das interfaces para a simulação de computação em nuvem

A interface icônica presente no iSPD tem o objetivo de facilitar a criação de um modelo de simulação computacional, em que através de ícones o usuário consegue facilmente criar uma grade computacional.

A interface sofreu alterações para a simulação em nuvem, foram criadas novas janelas de configuração e algumas já existentes foram modificadas.

Primeiramente dado que a ferramenta irá simular tanto modelo de grades computacionais quanto de computação em nuvem (IaaS), é necessário desenvolver uma janela de seleção de tipo de modelo que irá ser simulado. Esta janela está apresentada na figura 3.

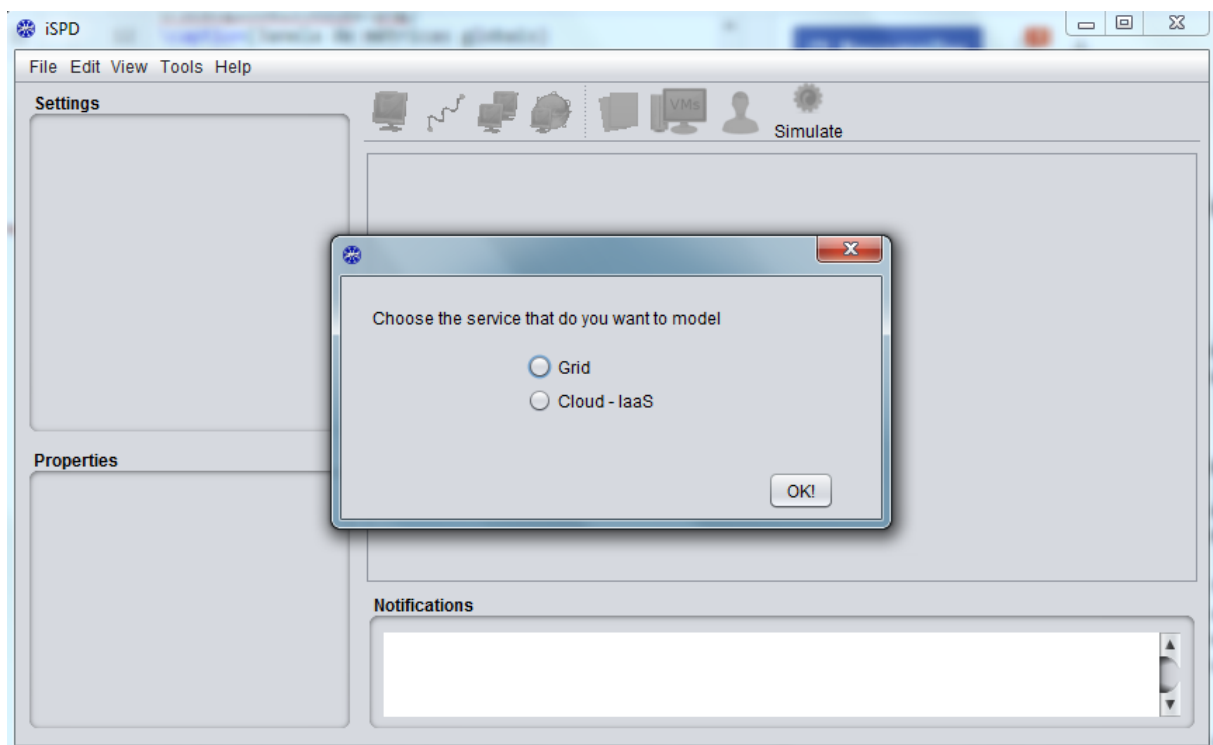


Figura 3 – Janela de escolha do modelo

Após selecionar a opção de computação de nuvem, o ambiente computacional, é modelado através dos ícones apresentados na área superior da interface. adicionando os ícones de máquinas individuais e *clusters* e interligando através dos ícones de enlace lógico e internet. Na figura 4 é apresentado um modelo de ambiente computacional onde à esquerda é exibido o painel de configuração dos atributos que constituem cada elemento da infraestrutura da nuvem. Abaixo do painel de configuração são exibidas as propriedades do elemento selecionado. Acima da área quadriculada (área de desenho) além dos ícones utilizado para a modelagem do ambiente, existem ainda os ícones responsáveis pela configuração de outras informações relevantes ao modelo, como o gerenciamento de usuários, configuração de carga de trabalho, gerenciador de máquinas virtuais e o último é o ícone de início de simulação.

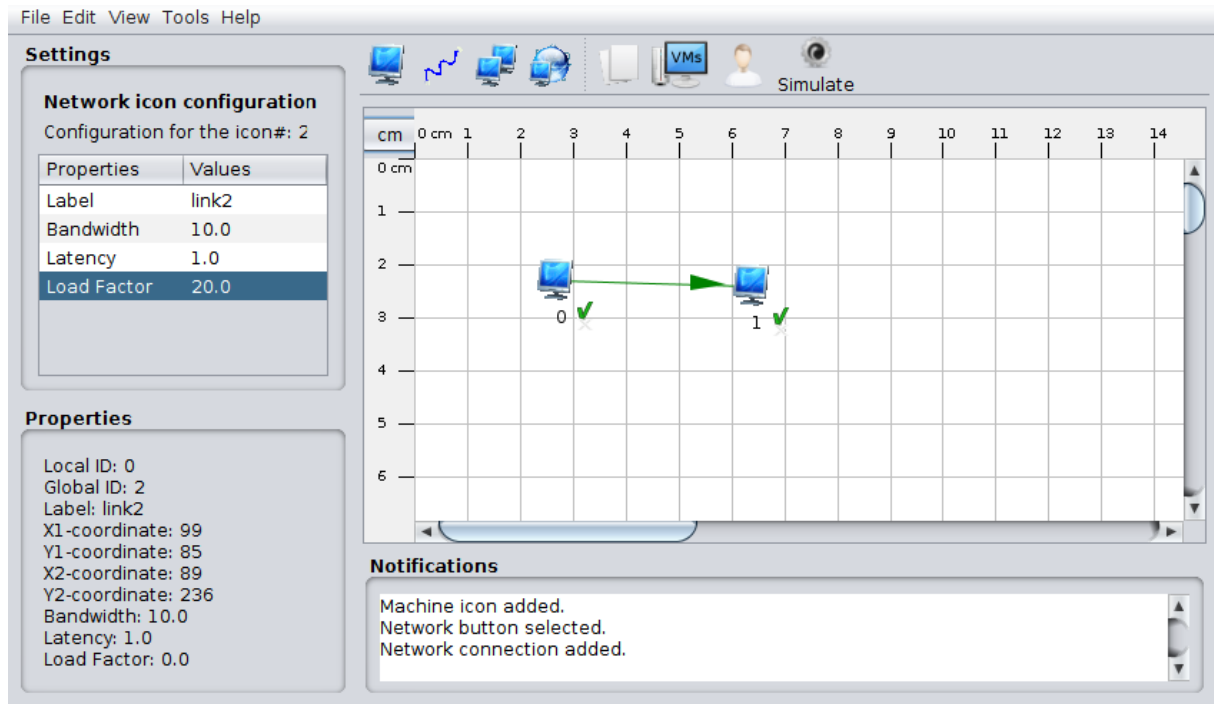


Figura 4 – Interface icônica do iSPD

Na janela de configuração das máquinas virtuais realiza-se a adição, remoção e a configuração das máquinas virtuais que serão utilizadas no modelo, para fazer essa configuração deve-se criar ao menos uma VMM e uma máquina física. Na janela de configuração é feito a configuração das características da máquina virtual, como por exemplo a quantidade de memória e a quantidade de processamento alocado. Ao terminar a configuração acionando o botão "add VM" a máquina virtual configurada será adicionada à lista de máquinas virtuais. Essa lista possui todas as máquinas virtuais adicionadas.

Nesta janela também é feita a remoção de uma máquina virtual caso seja necessário, selecionando a máquina e acionando o botão "remove VM". Na figura 5 apresenta-se a janela de configuração das máquinas virtuais, com as características que serão configuradas e a lista das máquinas virtuais.

As características são a qual usuário a máquina virtual pertence, a VMM que irá gerenciar essa máquina, a quantidade de núcleos de processamento alocados, quantidade de memória e disco e o sistema operacional instalado.

3.3 Motor de simulação

Esta seção será abordado a implementação feita no motor de simulação do iSPD, esta seção foi dividida em duas subseções para um melhor entendimento, a primeira abordará as implementações adicionadas ao motor, e a segunda abordará as políticas de alocação das máquinas virtuais adotadas.

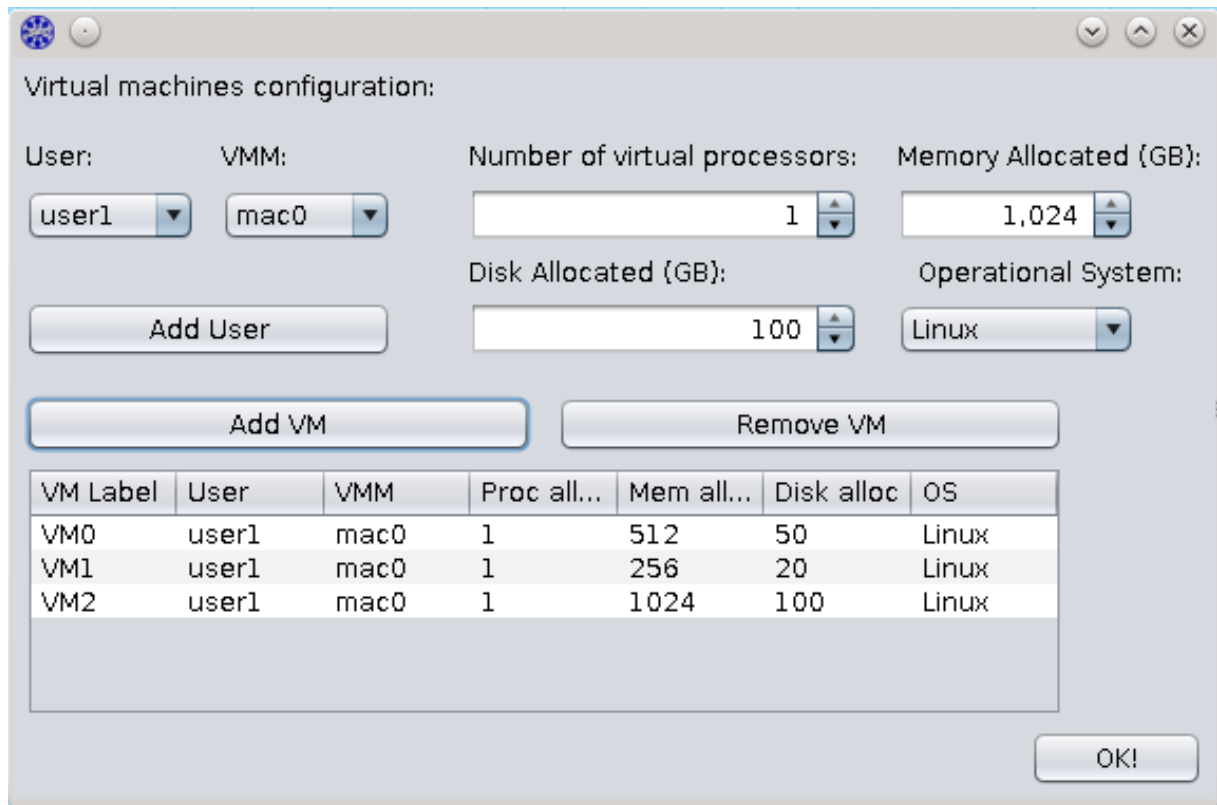


Figura 5 – Janela de configuração das máquinas virtuais

3.3.1 Implementação do motor de simulação

O motor de simulação contempla o local da realização da simulação do modelo gerado. Nesta subseção será detalhado o funcionamento desse motor.

O motor de simulação é composto pelos centros de serviço, rede de filas, políticas de escalonamento e de alocação das máquinas virtuais, além das métricas de simulação que serão detalhadas no próximo capítulo.

Os centros de serviço possuem a função de executar os processos, ou encaminhar uma tarefa para outro serviço. Como centro de serviço tem-se por exemplo as VMs, máquinas físicas, e VMM.

A política de escalonamento em nuvem implementada foi o *RoundRobin*, e adicionou-se um gerenciador de escalonadores em nuvem na interface principal, na opção *Tools*, onde é possível gerar novas políticas. Na figura 6 apresenta-se a janela do gerenciador de escalonadores.

A simulação começa no centro de serviço VMM, que gerenciará a fase de alocação e de escalonamento no motor.

Na fase de alocação a VMM possui a lista das máquinas físicas e virtuais adicionadas e a política de alocação adotada. A VMM irá enviar as VMs para as máquinas físicas, através dos centros de serviços de conexões, nas máquinas físicas as VMs são adicionadas em um lista de

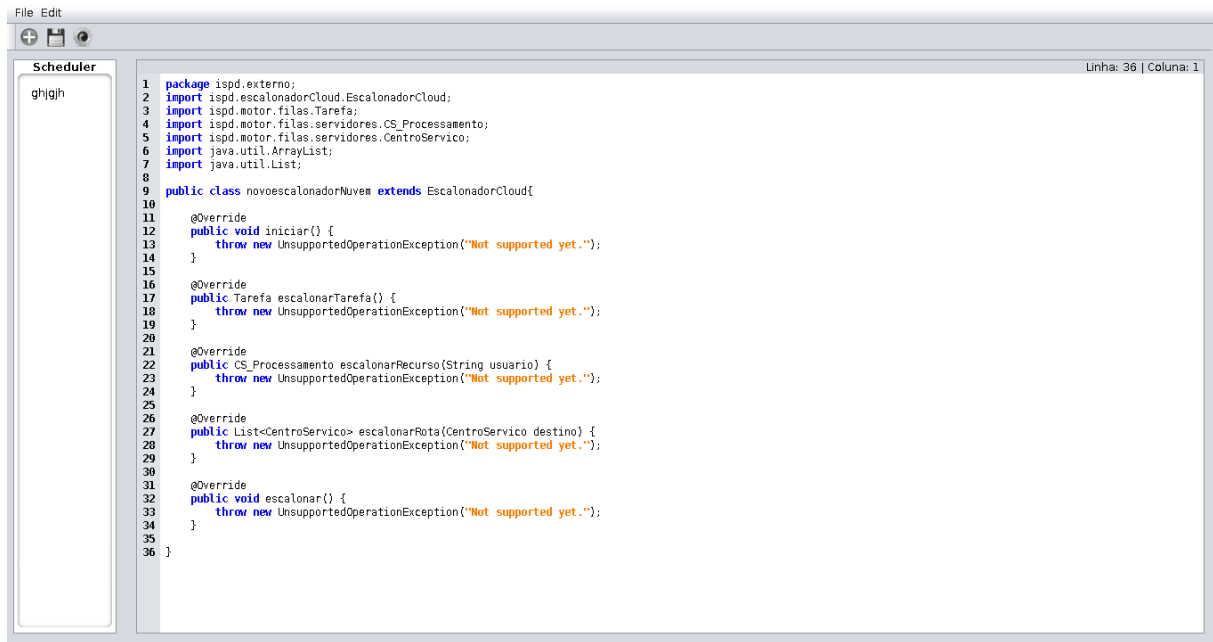


Figura 6 – Janela do gerenciador de escalonadores

VMs, caso a VM seja alocada.

Na fase de escalonamento a VMM possui a lista das máquinas físicas, lista de tarefas e a política de escalonamento adotada. Nesta fase as tarefas são enviadas para as máquinas físicas, que encaminharão essas tarefas para as VMs alocadas executarem.

O centro de serviço de máquina física têm a função de alocar as VMs e encaminhar as tarefas para elas executarem as tarefas na fase de escalonamento.

O centro de serviço de máquina virtual têm a função de executar as tarefas que são encaminhadas pelas máquinas físicas hospedeiras. Na figura 7 apresenta-se como os centros de serviços interagem no motor de simulação.

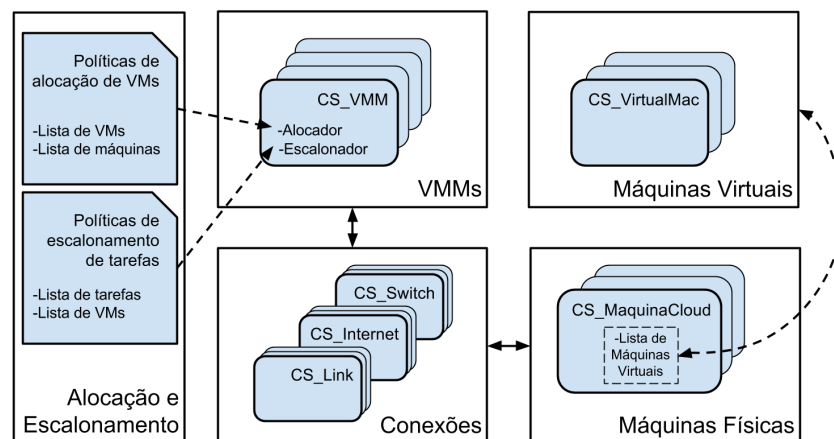


Figura 7 – Diagrama das classes de serviço do motor e suas funcionalidades

3.3.2 Política de alocação das máquinas virtuais

A política de alocação das máquinas virtuais tem a função de alocar as máquinas virtuais nas máquinas físicas. No iSPD foi implementado duas políticas, *RoundRobin* e o *Immediate*.

A política de alocação *RoundRobin* pega uma lista de máquinas virtuais com uma ou mais máquinas, e tenta alocar na primeira máquina física da lista, caso não seja possível, segue para a próxima máquina da lista. Ao alocar uma VM, a próxima VM da lista tentará ser alocada na próxima máquina física da lista depois da última máquina que recebeu uma VM na última alocação. Caso a VM não seja aceita em nenhuma máquina ela será adicionada a uma lista de máquinas rejeitadas.

A política de alocação *Immediate*, segue o mesmo princípio do *RoundRobin*, mas ao alocar uma VM em uma máquina, a próxima VM da lista tentará ser alocada nessa mesma máquina, até que a máquina não suporte mais uma VM, assim passará para a próxima máquina da lista, caso a VM não seja aceita em nenhuma máquina ela será adicionada a uma lista de máquinas rejeitadas.

Foi adicionado um gerenciador de alocadores, onde é possível editar novas políticas de alocação, esta janela está apresentada na figura 8.

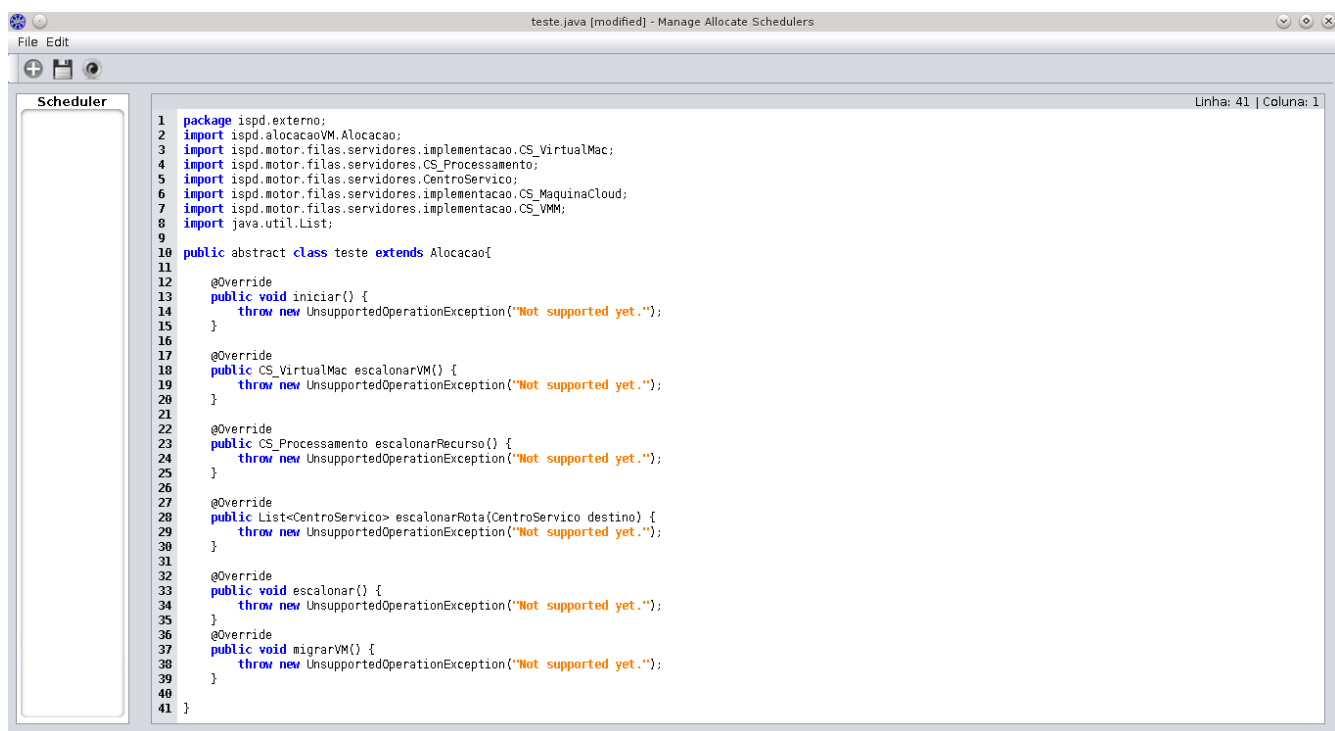


Figura 8 – Janela de gerenciamento dos alocadores

3.4 Considerações finais

Neste capítulo apresentou-se a modelagem em nuvem, o funcionamento do motor de simulação e suas políticas de alocação e de escalonamento.

No próximo capítulo, será apresentado os resultados obtidos até o presente momento deste trabalho, comparando o iSPD com o simulador CloudSim e apresentando as métricas utilizadas nos resultados.

4 Resultados obtidos

Apresentam-se nesse capítulo os resultados obtidos nos testes realizados e como foi o procedimento para a realização dos mesmos.

4.1 Procedimento para realização dos testes

Antes de se iniciar os testes foi adicionado uma nova métrica no motor de simulação, que utiliza características exclusivas de computação em nuvem, que são as métricas de custo, em que se calcula o custo total de disco (GB), processamento (custo por núcleo de processamento) e memória (GB) de todas as máquinas, este custo se refere a quantidade alocada para as máquinas virtuais, para cada unidade existe um custo que é configurado na janela de configuração das máquinas físicas. No fim da simulação é apresentado na janela de métricas globais este custo. Na figura 9 apresenta-se a janela de métricas globais com a adição das métricas de custo.

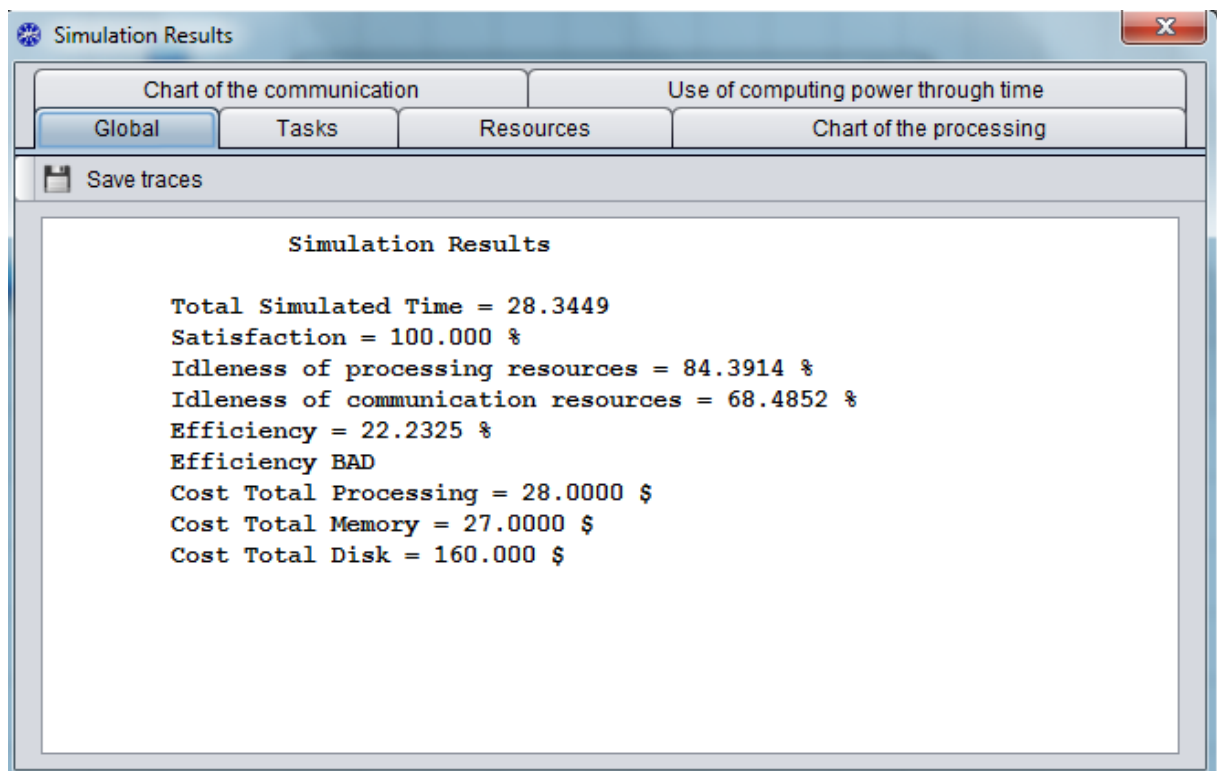


Figura 9 – Janela de métricas globais com as métricas de custo

A janela de configuração das cargas de trabalho é onde serão adicionadas as tarefas que serão escalonadas para as máquinas virtuais e posteriormente executadas.

A alteração do número de tarefas executadas nos testes foram configuradas nesta janela. Existem três possibilidades de configuração: configuração para cada nó, aleatoriamente ou por

traces. Na figura 10 apresenta-se a janela de configuração das cargas de trabalho.

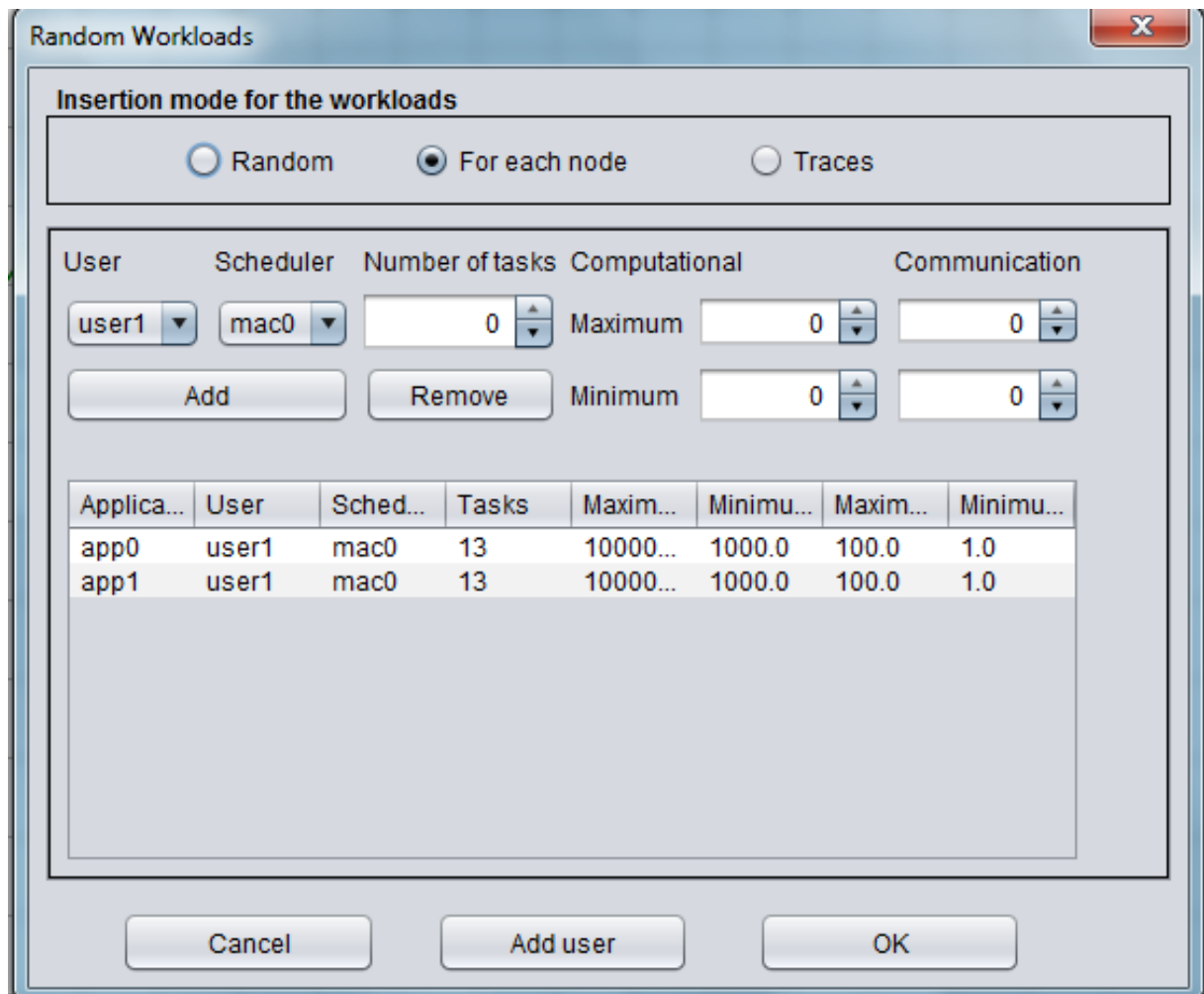


Figura 10 – Janela de configuração das tarefas

Na próxima seção será apresentado os testes realizados e suas métricas.

4.2 Resultados dos testes realizados

Nesta seção apresenta-se os testes feito com o iSPD em comparação ao simulador de nuvem CloudSim.

Utilizou-se o simulador CloudSim (CALHEIROS et al., 2011) especificado na seção 2.3.1 para teste de comparação, pois atualmente é um dos principais simuladores quando o conceito é simulação em nuvem, e um dos mais referenciados.

4.2.1 Testes comparativos com o CloudSim

Para a realização dos testes foi gerado um modelo com uma VMM, 8 máquinas físicas com 4 núcleos de 30000 MFlops de processamento, 4 GB de RAM, 100 GB de armazenamento.

Foram adicionadas 30 máquinas virtuais para serem alocadas com 1 GB de RAM, 10 GB de armazenamento e 1 núcleo de processamento.

Na figura 11 apresenta-se o modelo icônico gerado para os testes no iSPD. Com a interface icônica economizou-se centenas de linhas de código que foram necessárias para gerar o mesmo modelo de simulação no CloudSim.

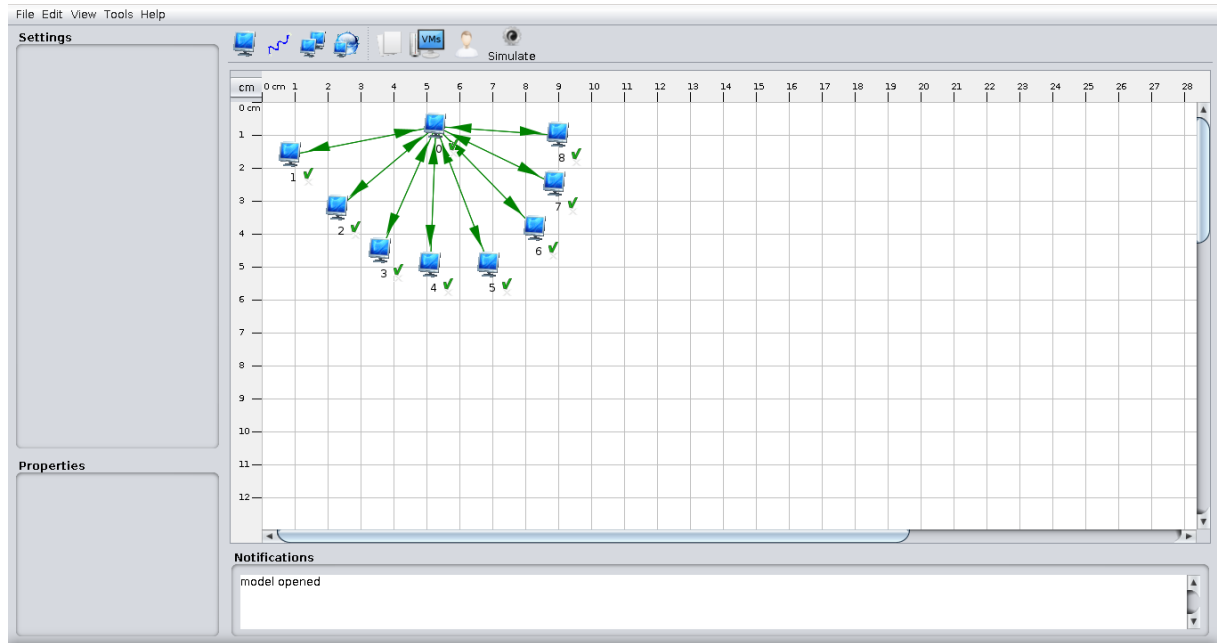


Figura 11 – Janela do modelo gerado para os testes

Foram feitos testes de 1000 à 20000 tarefas de tamanho de comunicação igual 300 bytes e 400000 MFlops de tamanho de processamento, simulando no CloudSim, e no iSPD utilizando as políticas de alocação *RoundRobin* e *Immediate*, e a política de escalonamento *RoundRobin* nos dois simuladores.

No gráfico apresentado na figura 12 é interessante notar que os tempos do CloudSim em comparação aos tempos do iSPD utilizando a política de alocação *RoundRobin* são semelhantes, devido a semelhança da política de alocação e da política de escalonamento utilizada no CloudSim. Já utilizando o iSPD com a política *Immediate*, nota-se uma maior diferença, pois é uma política totalmente diferente da adotada.

Na Tabela 2 está o tempo em segundos dos testes realizados.

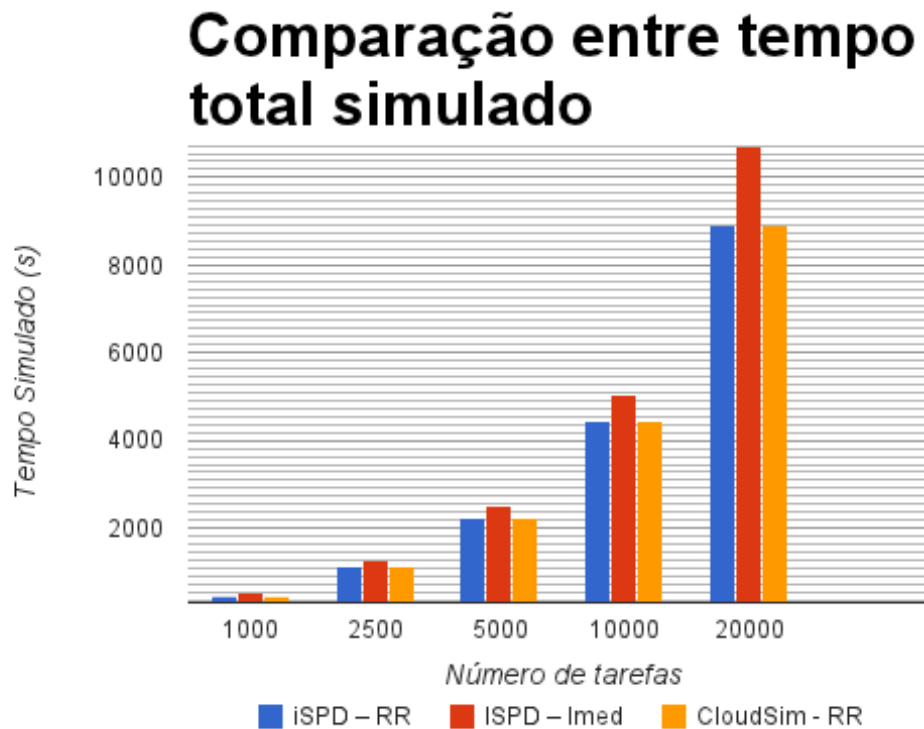


Figura 12 – Gráfico dos testes realizados

Núm.Tarefas	ISPD-RR	IRSPD-IM	CloudSim
1000	459,7	523,4	453,4
2500	1120,1	1283,4	1120,1
5000	2236	2523,4	2226,7
10000	4429,73	5030,07	4453,3
20000	8902,7	10700,1	8893,4

Tabela 2 – Tabela de dados dos testes realizados com o tempo de simulação em segundos.

Na apresentação dos resultados o iSPD apresenta janelas com várias métricas dos resultados, como gráficos de processamento das tarefas nas máquinas virtuais, métricas de comunicação, uso das tarefas durante o tempo de execução, uso dos recursos. Já o CloudSim apresenta apenas um log com o tempo e as tarefas executadas, dificultando assim o estudo do resultado da simulação.

Na figura 13 apresenta-se a janela do gráfico de processamento do teste com 2000 tarefas, em que se pode observar quanto cada VM adicionada processou durante a simulação. Nesse teste o desempenho das VMs foram semelhantes, pois todas possuíam a mesma configuração.

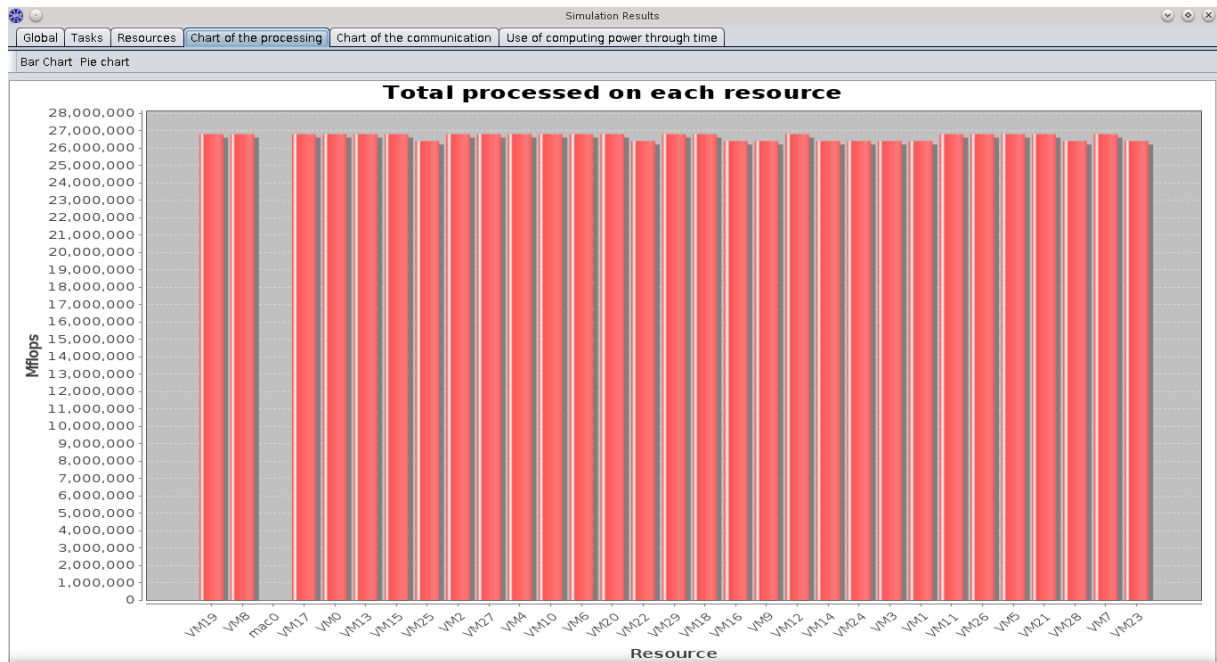


Figura 13 – Gráfico de processamento no iSPD

Na figura 14 apresenta-se as métricas globais em que se analisa-se o tempo de simulação, o tempo ocioso das máquinas, custo e a eficiência que permite analisar de modo geral o comportamento do sistema durante a simulação.

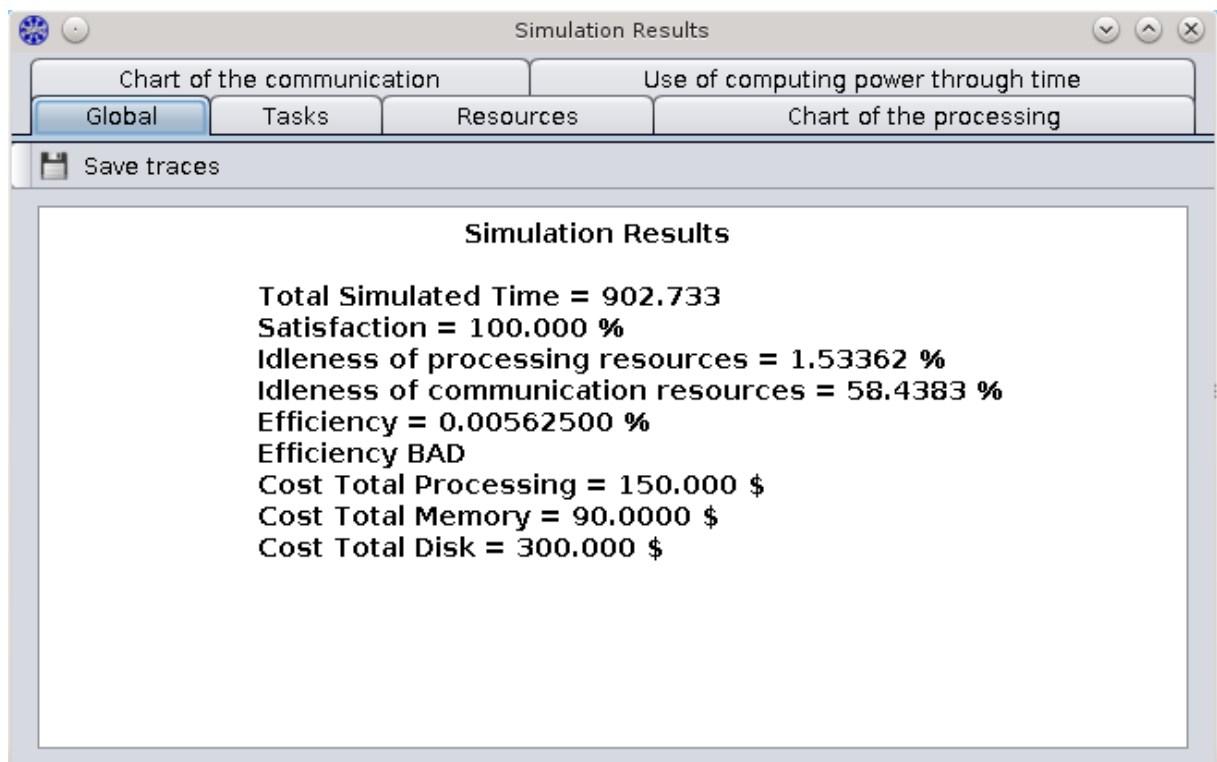
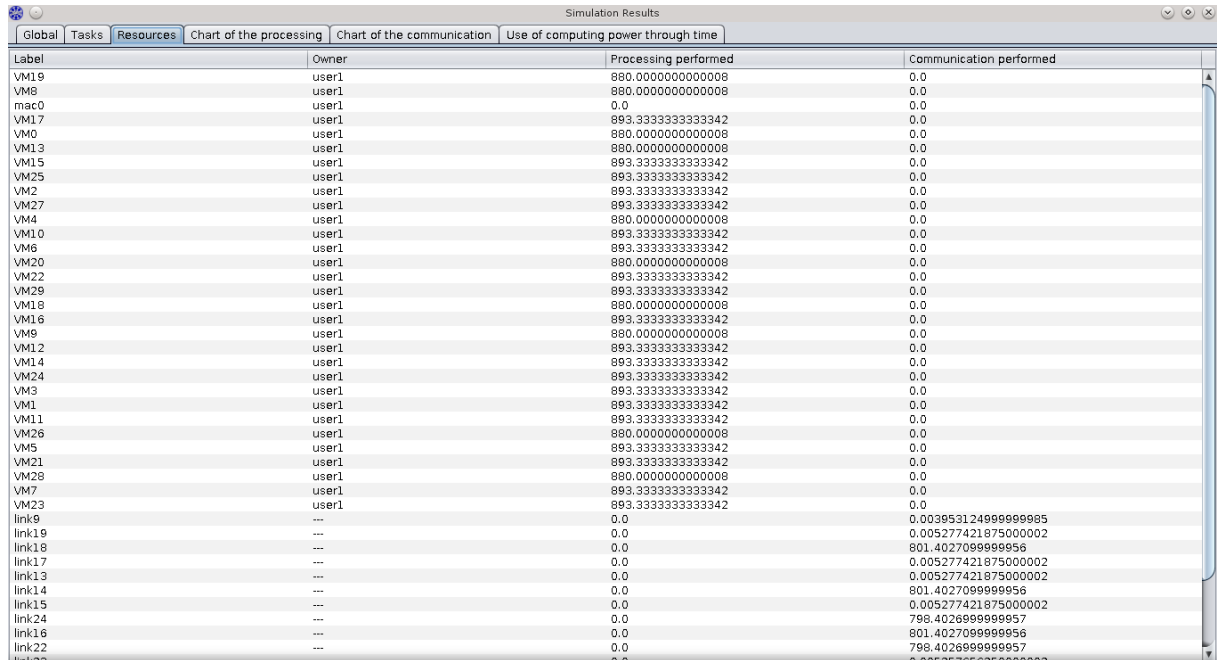


Figura 14 – Métricas globais no iSPD

Na figura 15 apresenta-se as métricas dos recursos, que permite analisar quanto em processamento cada VM executou durante a simulação e quanto cada link processou de informação, tendo assim uma análise da utilização dos recursos de todos os componentes.



The screenshot shows the 'Simulation Results' window with the 'Resources' tab selected. The table displays metrics for various VMs and links. The columns are 'Label', 'Owner', 'Processing performed', and 'Communication performed'.

Label	Owner	Processing performed	Communication performed
VM19	user1	880.0000000000008	0.0
VM8	user1	880.0000000000008	0.0
mac0	user1	0.0	0.0
VM17	user1	893.3333333333342	0.0
VM0	user1	880.0000000000008	0.0
VM13	user1	880.0000000000008	0.0
VM15	user1	893.3333333333342	0.0
VM25	user1	893.3333333333342	0.0
VM2	user1	893.3333333333342	0.0
VM27	user1	893.3333333333342	0.0
VM4	user1	880.0000000000008	0.0
VM10	user1	893.3333333333342	0.0
VM6	user1	893.3333333333342	0.0
VM20	user1	880.0000000000008	0.0
VM22	user1	893.3333333333342	0.0
VM29	user1	893.3333333333342	0.0
VM18	user1	880.0000000000008	0.0
VM16	user1	893.3333333333342	0.0
VM9	user1	880.0000000000008	0.0
VM12	user1	893.3333333333342	0.0
VM14	user1	893.3333333333342	0.0
VM24	user1	893.3333333333342	0.0
VM3	user1	893.3333333333342	0.0
VM1	user1	893.3333333333342	0.0
VM11	user1	893.3333333333342	0.0
VM26	user1	880.0000000000008	0.0
VM5	user1	893.3333333333342	0.0
VM21	user1	893.3333333333342	0.0
VM28	user1	880.0000000000008	0.0
VM7	user1	893.3333333333342	0.0
VM23	user1	893.3333333333342	0.0
link9	---	0.0	0.00395312499999995
link19	---	0.0	0.00527742187500002
link18	---	0.0	801.4027099999956
link17	---	0.0	0.00527742187500002
link13	---	0.0	0.00527742187500002
link14	---	0.0	801.4027099999956
link15	---	0.0	0.00527742187500002
link24	---	0.0	798.4026999999957
link16	---	0.0	801.4027099999956
link22	---	0.0	798.4026999999957
link23	---	0.0	0.00535768635000002

Figura 15 – Métricas de recursos no iSPD

Na figura 16 apresenta-se o log do mesmo teste de 2000 tarefas no CloudSim, em que se aparece o tempo que cada tarefa demorou para executar e em qual VM ela executou. A única informação de avaliação do resultado da simulação é este log, que se torna pouco eficiente, quando o teste possui muitas tarefas como este, sendo difícil analisar de modo geral o desempenho do sistema simulado.

Portanto a geração de métricas e resultados gráficos do iSPD facilitam o estudo da simulação, sendo possível analisar recursos, tarefas e processamento através de tabelas e gráficos, ao contrário do CloudSim, onde é necessário procurar informações em um *log* extenso, principalmente quando o número de tarefas é elevado. Outro fator é a difícil modelagem, devido a ausência de interface gráfica, que dificulta o processo de geração do modelo.

Saída - cloudsim (run)

Simulation completed.
Simulation completed.

===== OUTPUT =====

Cloudlet ID	STATUS	Data center ID	VM ID	Time	Start Time	Finish Time
24	SUCCESS	2	24	880	0,1	880,1
54	SUCCESS	2	24	880	0,1	880,1
84	SUCCESS	2	24	880	0,1	880,1
114	SUCCESS	2	24	880	0,1	880,1
144	SUCCESS	2	24	880	0,1	880,1
174	SUCCESS	2	24	880	0,1	880,1
204	SUCCESS	2	24	880	0,1	880,1
234	SUCCESS	2	24	880	0,1	880,1
264	SUCCESS	2	24	880	0,1	880,1
294	SUCCESS	2	24	880	0,1	880,1
324	SUCCESS	2	24	880	0,1	880,1
354	SUCCESS	2	24	880	0,1	880,1
384	SUCCESS	2	24	880	0,1	880,1
414	SUCCESS	2	24	880	0,1	880,1
444	SUCCESS	2	24	880	0,1	880,1
474	SUCCESS	2	24	880	0,1	880,1
504	SUCCESS	2	24	880	0,1	880,1
534	SUCCESS	2	24	880	0,1	880,1
564	SUCCESS	2	24	880	0,1	880,1
594	SUCCESS	2	24	880	0,1	880,1
624	SUCCESS	2	24	880	0,1	880,1
654	SUCCESS	2	24	880	0,1	880,1
684	SUCCESS	2	24	880	0,1	880,1
714	SUCCESS	2	24	880	0,1	880,1
744	SUCCESS	2	24	880	0,1	880,1
774	SUCCESS	2	24	880	0,1	880,1
804	SUCCESS	2	24	880	0,1	880,1
834	SUCCESS	2	24	880	0,1	880,1
864	SUCCESS	2	24	880	0,1	880,1
894	SUCCESS	2	24	880	0,1	880,1
924	SUCCESS	2	24	880	0,1	880,1
954	SUCCESS	2	24	880	0,1	880,1
984	SUCCESS	2	24	880	0,1	880,1
1014	SUCCESS	2	24	880	0,1	880,1
1044	SUCCESS	2	24	880	0,1	880,1
1074	SUCCESS	2	24	880	0,1	880,1

Figura 16 – Log gerado no CloudSim

4.3 Considerações Finais

Nesta seção apresentam-se os procedimentos e os testes realizados no simulador iSPD com adição das implementações de simulação em nuvem em comparação ao simulador de computação em nuvem CloudSim, se diferenciando por sua interface icônica que facilita a configuração de todos os componentes do modelo, apresentando a opção de criação de novas políticas e de escolha de mais de uma política de alocação.

Por fim apresentando métricas de resultados que facilitam o estudo da simulação do modelo gerado.

No próximo capítulo apresenta-se as conclusões e direções futuras.

5 Conclusões

Neste trabalho apresentou-se um estudo sobre computação em nuvem e suas origens, os simuladores de computação em nuvem e um comparativo entre eles. Apresentou-se também a especificação da caracterização dos recursos e carga de trabalho para a modelagem e simulação de computação em nuvem, as implementações adicionadas ao motor de simulação, como as políticas de alocação e de escalonamento. E por fim as métricas de simulação desenvolvidas, e o teste comparativo com a ferramenta CloudSim.

Conclui-se que neste trabalho procurou-se deixar o iSPD uma ferramenta de simulação capaz de simular um ambiente em nuvem de maneira prática e intuitiva, de modo que facilite o acesso a simulação de ambientes de nuvem.

5.1 Dificuldades

Um dos principais desafios deste trabalho foi encontrar uma documentação teórica, pois como computação em nuvem é uma tecnologia recente e existem poucos simuladores baseados nela, portanto foi complicado encontrar referências adequadas.

Outra dificuldade foi simular a alocação das máquinas virtuais, pois foi necessário alterar o motor de simulação, criando novos centros de serviços, e alterando o modo que ele encaminha as tarefas na simulação.

E por fim uma grande dificuldade foram os testes realizados, pois houve a necessidade de estudar toda documentação do simulador comparado e codificar manualmente o modelo gerado.

5.2 Próximas ações

As próximas etapas deste trabalho contemplam as seguintes ações:

- **Adição de novas políticas de alocação e de escalonamento** : Consiste em adicionar novas políticas de alocação de máquinas virtuais e de escalonamento de tarefas, para aumentar a variedade de testes possíveis;
- **Modelagem da classe de serviço PaaS**: Consiste em estudar e implementar a classe de serviço PaaS, especificada na subseção 2.1.5, que se difere do IaaS, pois a virtualização não é da infraestrutura, o usuário ao invés de criar as máquinas virtuais, irá criar as aplicações que serão executadas, e o simulador irá criar a infraestrutura adequada;

- **Adição de novas métricas voltadas para computação em nuvem:** Consiste em implementar novas métricas de avaliação voltadas para computação em nuvem, para apresentar resultados mais completos, facilitando assim o estudo.

5.3 Publicações

Este trabalho foi publicado nos seguintes eventos:

Modelagem e implementação em computação em nuvem no simulador de grades computacionais iSPD. Apresentado no Congresso de Iniciação Científica de 2014 (Cic - Unesp).

Simulação de sistemas de computação em nuvem para o iSPD. Publicado no Volume 4, Número 1, 2015, da Revista Interciência Sociedade.

Referências

AMAZON. AWS - Amazon Web Services. Disponível em <<http://aws.amazon.com/pt/>>. 2015. Citado 2 vezes nas páginas 14 e 17.

ARCOS. What is icancloud? Disponível em <<http://www.arcos.inf.uc3m.es/icancloud/Home.html>>. 2013. Citado na página 19.

ARMBRUST, M. et al. *Above the Clouds: A Berkeley View of Cloud Computing*. [S.l.], 2009. Disponível em: <<http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html>>. Citado na página 15.

BELOGLAZOV, A.; BUYYA, R. Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. *Concurr. Comput. : Pract. Exper.*, John Wiley and Sons Ltd., Chichester, UK, v. 24, n. 13, p. 1397–1420, set. 2012. ISSN 1532-0626. Disponível em: <<http://dx.doi.org/10.1002/cpe.1867>>. Citado na página 20.

BUYYA, R.; BROBERG, J.; GOSCINSKI, A. *Cloud Computing: Principles and Paradigms*. Wiley, 2011. (Wiley Series on Parallel and Distributed Computing). ISBN 9781118002209. Disponível em: <<http://books.google.com.br/books?id=S1NvRRd77rQC>>. Citado 2 vezes nas páginas 15 e 16.

BUYYA, R.; RANJAN, R.; CALHEIROS, R. N. Modeling and simulation of scalable cloud computing environments and the cloudsims toolkit: Challenges and opportunities. *CoRR*, abs/0907.4878, 2009. Citado na página 18.

CALHEIROS, R. N. et al. CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Softw. Pract. Exper.*, John Wiley & Sons, Inc., New York, NY, USA, v. 41, n. 1, p. 23–50, jan. 2011. ISSN 0038-0644. Disponível em: <<http://dx.doi.org/10.1002/spe.995>>. Citado na página 34.

CASTANE, G. G.; NUNEZ, A.; CARRETERO, J. icancloud: A brief architecture overview. In: *ISPA*. IEEE, 2012. p. 853–854. ISBN 978-1-4673-1631-6. Disponível em: <<http://dblp.uni-trier.de/db/conf/ispa/ispa2012.html#CastaneNC12>>. Citado na página 18.

CLOUDS. Cloudsim: A framework for modeling and simulation of cloud computing infrastructures and services. Disponível em <<http://www.cloudbus.org/cloudsim/>>. 2013. Citado 2 vezes nas páginas 18 e 19.

GOOGLE. Google cluster data. Disponível em <<http://code.google.com/p/googleclusterdata/>>. 2012. Citado na página 14.

GOOGLE. Google AppEngine, Power your business with google cloud. Disponível em <<https://developers.google.com/appengine/>>. 2013. Citado na página 18.

GOOGLE. Google docs. Disponível em <docs.google.com/?hl=pt-BR>. 2013. Citado na página 18.

GREENCLOUD. Greencloud - the green cloud simulator website. Disponível em <<http://greencloud.gforge.uni.lu/>>. 2013. Citado na página 20.

- KLIAZOVICH, D. et al. GreenCloud: A packet-level simulator of energy-aware cloud computing data centers. In: *Global Telecommunications Conference (GLOBECOM 2010)*, 2010 IEEE. [S.l.: s.n.], 2010. p. 1–5. ISSN 1930-529X. Citado 2 vezes nas páginas 18 e 20.
- MANACERO, A. et al. iSPD: an iconic-based modeling simulator for distributed grids. In: *Annals of 45th Annual Simulation Symposium*. Orlando, USA: [s.n.], 2012. (ANSS12, CDROM), p. 1–8. Citado na página 21.
- MELL, P.; GRANCE, T. *The NIST Definition of Cloud Computing*. Gaithersburg, MD, 2011. Disponível em: <<http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>>. Citado 3 vezes nas páginas 15, 16 e 17.
- MICROSOFT. Office web apps. Disponível em <<http://office.microsoft.com/pt-br/web-apps/>>. 2013. Citado na página 14.
- MICROSOFT. Microsoft azure. Disponível em <<http://azure.microsoft.com/pt-br/>>. 2015. Citado na página 18.
- OMNET++. Welcome to the omnet++ community! Disponível em <<http://www.omnetpp.org/>>. 2013. Citado na página 19.
- RITTINGHOUSE, J.; RANSOME, J. F. *Cloud Computing: Implementation, Management, and Security*. [S.l.]: CRC, 2009. Citado na página 14.
- SALESFORCE. Salesforce. Disponível em <<http://www.salesforce.com/>>. 2015. Citado na página 14.
- SILVA, D. T. da. Identificação de características definidoras de tarefas em ambientes de computação em nuvem. 2013. Citado na página 21.
- SILVA, D. T. da. Simulação de sistemas de computação em nuvem para o ispd. Disponível em <http://intercienciaesociedade.fmpfm.edu.br/colecao/online/v4_n1/086-093.pdf>. 2015. Citado na página 21.
- THOMSEN, E. *OLAP Solutions: Building Multidimensional Information Systems*. Wiley, 2002. (Wiley Computer Publishing). ISBN 9780471400301. Disponível em: <<http://books.google.com.br/books?id=vVZIYeVHOcwC>>. Citado na página 16.
- VAQUERO, L. M. et al. A break in the clouds: Towards a cloud definition. *SIGCOMM Comput. Commun. Rev.*, ACM, New York, NY, USA, v. 39, n. 1, p. 50–55, dez. 2008. ISSN 0146-4833. Disponível em: <<http://doi.acm.org/10.1145/1496091.1496100>>. Citado na página 14.
- VMWARE. VMware ESXi e ESX. Disponível em <<http://www.vmware.com/br/products/esxi-and-esx/overview.html>>. 2013. Citado na página 16.
- XEN. Why Xen Project? Disponível em <<http://www.xenproject.org/users/why-the-xen-project.html>>. 2013. Citado na página 16.