



Universidade Federal do Ceará

Centro de Ciências

Departamento de Computação

Mestrado e Doutorado em Ciência da Computação

**GERNU: UMA ABORDAGEM PARA GERENCIAMENTO DE NUVENS BASEADA
NOS REQUISITOS DO USUÁRIO, CRIAÇÃO DINÂMICA DOS SERVIÇOS E
DEFINIÇÃO DE ATRIBUTOS DE QUALIDADE**

Hélder Pereira Borges

FORTALEZA - CE

Fevereiro - 2013

HÉLDER PEREIRA BORGES

**GERNU: UMA ABORDAGEM PARA AQUISIÇÃO DE SERVIÇOS EM NUVENS,
BASEADA NOS REQUISITOS DO USUÁRIO, CRIAÇÃO DINÂMICA DOS
SERVIÇOS E DEFINIÇÃO DE ATRIBUTOS DE QUALIDADE**

Tese submetida a Coordenação do Curso de Pós-Graduação em Ciência da Computação da Universidade Federal do Ceará como requisito parcial para obtenção do título de Doutor em Ciência da Computação.

Área de concentração: Ciência da Computação

Orientadores:

Prof. José Neuman de Sousa, Dr.

Prof. Bruno Schulze, Dr.

FORTALEZA - CE

Fevereiro - 2013

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca de Ciências e Tecnologia

-
- B731g Borges, Hélder Pereira.
Gernu: uma abordagem para aquisição de serviços em nuvens, baseada nos requisitos do usuário, criação dinâmica dos serviços e definição de atributos de qualidade / Hélder Pereira Borges. – 2013.
175 f. : il. color., enc. ; 30 cm.
- Tese (doutorado) – Universidade Federal do Ceará, Centro de Ciências, Departamento de Computação, Programa de Pós-Graduação em Ciências da Computação, Fortaleza, 2013.
Área de Concentração: Sistemas de Informação.
Orientação: Prof. Dr. José Neuman de Sousa.
Coorientação: Prof. Dr. Bruno Richard Schulze.
1. Computação em nuvem. 2. Serviços da web-controle de qualidade. I. Título.

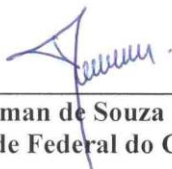
CDD 005

**GERNU: UMA ABORDAGEM PARA AQUISIÇÃO DE SERVIÇOS EM NUVENS,
BASEADA NOS REQUISITOS DO USUÁRIO, CRIAÇÃO DINÂMICA DOS
SERVIÇOS E DEFINIÇÃO DE ATRIBUTOS DE QUALIDADE**

Tese submetida a Coordenação do Curso de Pós-Graduação em Ciência da Computação da Universidade Federal do Ceará como requisito parcial para a obtenção do título de Doutor em Ciência da Computação. Área de concentração: Ciência da Computação.

Aprovada em 04/03/2013

BANCA EXAMINADORA



Prof. Dr. José Neuman de Souza (Orientador)
Universidade Federal do Ceará – UFC



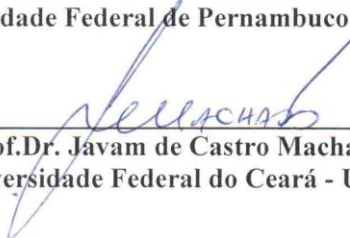
Prof. Dr. Antonio Roberto Mury
Laboratório Nacional de Computação Científica – LNCC



Prof. Dr. Bruno Richard Schulze (Coorientador)
Laboratório Nacional de Computação Científica – LNCC



Prof. Dr. Stênio Flávio de Lacerda Fernandes
Universidade Federal de Pernambuco - UFPE



Prof. Dr. Javam de Castro Machado
Universidade Federal do Ceará - UFC

Fortaleza, 04 de Março de 2013

A Deus. Meu criador e amigo. Sempre presente, pontual em cada necessidade e atento a todas elas. Obrigado Senhor, bendito és Tu que produz o querer e efetiva o realizar, sendo o motivo de Te honrar com este trabalho, foi muitas vezes o alento para continuar.

A minha amada e preciosa família. Obrigado por abdicarem o usufruto de muitas realidades para viverem comigo esta grande experiência.

“Sofremos muito com o pouco que nos falta e gozamos pouco o muito que temos.”

(William Shakespeare)

AGRADECIMENTOS

A Deus, por tudo, principalmente por toda provisão, discernimento e direcionamento durante esta longa jornada.

A minha querida esposa, Débora, uma deliciosa rapadura, doce e dura, incentivando, ajudando e socorrendo, sem perder a direção ou o objetivo do propósito maior. Obrigado por acreditar no que eu acredito.

Ao meu querido filho Hatus, cientista mirim, cheio de criatividade para quase todas as coisas, saudoso de todos, porém firme em não trocar a presença dos pais.

A minha querida filha Hadassa, farta em alegria que contagia a casa inteira, tão pequena e decidida a fazer tudo sozinha que chora e ri em um segundo.

Aos meus pais, Dalmo e Dulce, nem sempre fisicamente presentes, porém sempre amando e encorajando, acreditando também que existe um propósito maior que vale a pena pagar o preço pelo seu cumprimento.

A meu orientador na UFC Prof. José Neuman, pela parceria, ensinamentos, dedicação à pesquisa e confiança demonstrada em meu trabalho.

A meu orientador no LNCC Prof. Bruno Schulze, por viabilizar minha presença e participação no ComCidis, pelos constantes incentivos e confiança demonstrada em meu trabalho.

A meu amigo e tutor Antonio Mury, por toda disponibilidade, por todas as muitas conversas, elucidativas e inspiradoras, pela paciência, pelas incontáveis revisões de artigos e confiança demonstrada em meu trabalho.

Aos participantes do grupo ComCiDis, pelo acolhimento, encorajamento e múltiplas sugestões que serviram para o aperfeiçoamento deste trabalho e pelas inúmeras xícaras de café, regadas a descontração e boa prosa. Especial agradecimento ao Daniel Yokoyama, por todas as repetidas vezes que configuramos o servidor PXE em suas diversificadas versões e muitíssimos detalhes.

A Universidade Federal do Ceará, pela oportunidade de grande crescimento

profissional e intelectual. Aos professores e funcionários do Departamento de Computação e ao Orley pela simpatia e eficiência com que sempre me atendeu.

Ao Departamento Acadêmico de Informática do Instituto Federal do Maranhão (IFMA), por concordarem com o meu afastamento e conseqüentemente fazerem o trabalho que me era devido. Sou grato a todos os colegas que de alguma forma contribuíram com sua experiência e conhecimento durante este trabalho.

Agradeço também a Fundação de Amparo a Pesquisa no Estado do Maranhão (FAPEMA), pelo apoio financeiro dado a este trabalho durante quase toda a sua realização.

RESUMO

A Computação em Nuvem foi estabelecida nos últimos anos como uma importante área de pesquisa. Este paradigma tem se consolidado, pois, atualmente, tarefas como a obtenção, manipulação, compartilhamento e exploração de grandes quantidades de dados são cada vez mais comuns, exigindo muitos recursos computacionais. Dentro deste contexto, a computação em nuvem pode contribuir, pois pode fornecer estes recursos indefinidamente, incluindo memória, processamento, armazenamento entre outros, todos para uso imediato.

A provisão de serviços em nuvens envolve várias tarefas complexas. Para elevar a satisfação dos usuários, estas tarefas precisam considerar aspectos relevantes para eles, além das políticas particulares dos provedores. Desta forma, com o objetivo de definir um processo flexível para a especificação e aquisição de serviços nas nuvens, este trabalho propõe uma abordagem que utiliza um modelo de ambiente virtual contendo todas as informações relevantes para criação de um serviço personalizado a partir dos requisitos de hardware e software do usuário.

Outro desafio importante para a computação em nuvem está relacionado com o controle de qualidade dos serviços prestados, sendo que esta pode ser diretamente afetada pela utilização dos recursos. Sendo assim, a utilização de mecanismos automatizados para gerar serviços, monitorar os recursos e a qualidade dos serviços também contribui para incrementar o nível de satisfação dos clientes, além de criar um ambiente menos suscetível a erros para os administradores. Além dos aspectos de hardware e software, o modelo de ambiente conterá os atributos de qualidade do serviço, tendo sido definido um processo de negociação automatizado que ocorre concomitante ao processo de contratação do serviço. Assim, esta abordagem será capaz de entregar automaticamente infraestrutura, plataforma e software como serviço a partir dos requisitos específicos do usuário.

Palavras Chaves: Computação em Nuvem, IaaS, PaaS, SaaS, QoS, SLA, Especificação de Serviços, Modelos.

ABSTRACT

Cloud Computing has been established in recent years as an important area of research. This paradigm has been consolidated, since tasks such as obtaining, handling, sharing and exploitation of large amounts of data are becoming increasingly common, requiring many computing resources. Considering this context, cloud computing can help providing these resources indefinitely, including memory, processing, storage and others, all of them for immediate use.

The provision of cloud services involves several complex tasks. To increase the user satisfaction, these tasks need to consider issues relevant to them, in addition to the providers' policies. In order to define a flexible process for specification and procurement of services in clouds from the hardware and software requirements of users, this work proposes an approach that uses a virtual environment model containing all the relevant information to provide a personalized service.

Another key challenge for cloud computing is related to the services quality control. This aspect may be directly affected by resource utilization. Thus, the use of automated mechanisms to provide services, monitoring resources, as well the quality of services also helps to increase the level of customer satisfaction, creating an environment less susceptible to errors for the administrators. Besides the aspects of hardware and software, the virtual environment model will contain the quality of service attributes, having been set an automated negotiation process that occurs concomitantly with the procurement service process. Thus, this approach will be able to automatically deliver infrastructure, platform and software as a service from the user's specific requirements.

Keywords: Cloud Computing, IaaS, PaaS, SaaS, QoS, SLA, Service Specification, Models.

SUMÁRIO

RESUMO	vi
ABSTRACT	vii
SUMÁRIO.....	viii
LISTA DE ABREVIATURAS	xiii
LISTA DE FIGURAS	xv
LISTA DE TABELAS	xviii
1. Introdução.....	1
1.1 Motivação	1
1.1.1 A Computação em Nuvem	5
1.1.2 Acordos em Nível de Serviço (SLA)	7
1.1.3 Qualidade dos Serviços (QoS)	9
1.1.4 Abordagem Dirigida a Modelos	12
1.2 Definição do problema	13
1.3 Objetivos.....	18
1.3.1 Objetivos Específicos	18
1.4 Contribuições	19
1.5 Metodologia Adotada	21
1.6 Organização do Texto	22
2.Computação em Nuvem.....	24
2.1 Introdução	24
2.2 Conceitos Fundamentais	26
2.3 Características Essenciais	27
2.3.1 Virtualização de Recursos	28

2.3.2 Serviços sob Demanda	28
2.3.3 Independência de Localização e Amplo Acesso.....	28
2.3.4 Elasticidade e Escalabilidade	29
2.3.5 Medição dos Serviços	29
2.3.6 Repositório de Recursos	30
2.4 Modelos de Serviços	30
2.4.1 Infraestrutura como Serviço (IaaS).....	30
2.4.2 Plataforma como Serviço (PaaS).....	32
2.4.3 Software como Serviço (SaaS).....	32
2.4.4 Outros Serviços	33
2.5. Conclusão	33
3. Acordos em Nível de Serviço	34
3.1 Introdução	34
3.2 Conceitos Iniciais	37
3.3 Requisitos SLA	39
3.4 Funcionalidades de um SLA.....	42
3.4.1 Otimização da Seleção de Recursos	42
3.4.2 Monitoramento em Tempo Real.....	43
3.4.3 Negociação do Contrato.....	44
3.4.4 Publicação e Descoberta de Serviços	45
3.4.5 Criação de Modelos SLA.....	46
3.4.6 Contabilidade do SLA	47
3.6 Negociação do SLA.....	48
3.6.1 Estado da arte em negociação	49
3.6 Conclusão	53
4. Trabalhos Relacionados.....	54

4.1 Contratos em Nível de Serviços (SLA)	55
4.1.1 Comparativo de Atividades	61
4.1.2 Abordagem do GerNU	61
4.2 Qualidade de Serviço.....	62
4.2.1 Comparativo de Atividades	66
4.2.2 Abordagem do GerNU	66
4.3 Abordagem Dirigida a Modelos.....	67
4.3.1 Comparativo de Atividades	74
4.3.2 Abordagem do GerNU	74
4.4 Conclusão	75
5. GerNU, uma abordagem para especificação e provisionamento de serviços.....	77
5.1 Introdução	77
5.2 Visão Geral da Proposta.....	80
5.2.1 Processo para Especificação e Provisionamento de Serviços no GerNU	80
5.2.2 Cenário Proposto	84
5.2.2.1 Fase 1 - Especificação	85
5.2.2.2 Fase 2 - Negociação.....	88
5.2.2.3 Fase 3 - Implantação.....	90
5.2.2.4 Fase 4 - Monitoramento.....	91
5.2.2.5 Ciclo de Vida.....	92
5.2.3 Agentes do Sistema.....	93
5.2.3.1 Agente Negociador	94
5.2.3.2 Agente Extrator	95
5.2.3.3 Agente Criador	97
5.2.3.4 Agente Monitor	99
5.3 Visão Detalhada da Proposta	99

5.3.1 Abordagem Dirigida a Modelo	100
5.3.3 Infraestrutura de Hardware	102
5.3.4 Infraestrutura de Software.....	104
5.3.4.1 Preboot eXecution Environment (PXE)	105
5.3.5 Atributos de Qualidade	111
5.3.5.1 Introdução a Dependabilidade.....	112
5.3.5.2 Atributos da Dependabilidade	114
5.3.5.2 Atributos de QoS do GerNU	118
5.3.6 Processo de Negociação.....	122
5.3.6.1 Negociação Implícita	122
5.3.6.2 Negociação Explícita.....	124
5.3.6.3 Estratégia de Negociação.....	128
5.3.6.4 Finalização da Negociação	132
5.3.7 Gestão do SLA	133
5.5 Conclusão	135
6. Avaliação da Proposta	137
6.1 O GerNU envia o modelo do ambiente para um provedor disponibilizar o serviço .	137
6.2 O GerNU envia uma imagem virtual para o provedor disponibilizar o serviço	138
6.3 O GerNU provisiona o serviço	141
6.4 Utilização do GerNU.....	144
6.5 Avaliação do GerNU.....	148
6.6 Conclusão	152
7. Considerações Finais.....	153
7.1 Conclusões.....	154
7.2 Contribuições	156
7.3 Trabalhos Futuros.....	158

Referências.....	160
Apêndice A - Questionário de Avaliação.....	175

LISTA DE ABREVIATURAS

AHP	Analytic Hierarchy Process
API	Application Programming Interface
BPM	Business Process Manager
CT&I	Ciência, Tecnologia e Inovação
DHCP	Dynamic Host Configuration Protocol
DoE	Departamento de Energia Americano
ERP	Enterprise Resource Planning
HPC	High Performance Computing
IaaS	Infrastructure as a Service
ICNP	Iterated Contract Net Protocol
IU	Interface do Usuário
KPI	Key Performance Indicator
KQI	Key Quality Indicator
MDA	Model Driven Architecture
MTBF	Mean Time Between Failures
MTTR	Mean Time To Repair
MV	Máquina Virtual
PaaS	Platform as a Service
PXE	Preboot eXecution Environment
QoS	Quality of Service
REST	Representational State Transfer

SaaS	Software as a Service
SLA	Service Level Agreements
SLM	Service Level Management
SLO	Service Level Objective
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
TFTP	Trivial File Transfer Protocol
TI	Tecnologia da Informação
TIC	Tecnologia da Informação e Comunicação
VDI	VirtualBox Disk Image
VMDK	Virtual Machine Disk
XML	Extensible Markup Language

LISTA DE FIGURAS

Figura 1.1: Cenário do Problema	16
Figura 1.2: Processo de Virtualização	17
Figura 1.3: Metodologia para desenvolvimento da abordagem proposta	22
Figura 1.4: Organização da Proposta	23
Figura 2.1: Cenário da Computação em Nuvem	25
Figura 2.2: Modelo de Serviço	31
Figura 2.3: Exemplo de Serviços	31
Figura 3.1: Visão Geral do SLA	36
Figura 3.2: Requisitos de um SLA.....	40
Figura 5.1: Visão geral do desenvolvimento do GerNU	81
Figura 5.2: Processo usual para provisionamento de serviços em nuvem	81
Figura 5.3 Processo proposto para provisionamento de serviços em nuvem.....	82
Figura 5.4: Cenário de Aplicação do GerNU	84
Figura 5.5: Especificação de um IaaS	85
Figura 5.6: Especificação de um PaaS	86
Figura 5.7: Especificação de um SaaS	87
Figura 5.9: O GerNU atendendo o processo para aquisição de serviços	93
Figura 5.10: Ciclo de Vida do GerNU	93
Figura 5.11: Agentes do GerNU	94
Figura 5.12: Agente Negociador.....	96
Figura 5.13: Agente Extrator	97
Figura 5.14: Agente Criador	98
Figura 5.15: Agente Monitor	100

Figura 5.16: Modelo do Ambiente Virtual	101
Figura 5.17: Infraestrutura de Software	105
Figura 5.18: Um modelo PXE	107
Figura 5.20: Processo para criação de uma máquina virtual.....	109
Figura 5.18: Arvore da Dependabilidade, adaptado de [Laprie et al., 1992]	113
Figura 5.19: Proposta Inicial	123
Figura 5.20: Processo de negociação do GerNU	132
Figura 5.21: Gerenciamento do SLA no GerNU	135
Figura 6.1: Provedor cria imagem e disponibiliza o serviço	137
Figura 6.2: GerNU cria imagem e envia para o provedor disponibilizar o serviço.....	138
Figura 6.3: Modelo do ambiente virtual.....	141
Figura 6.4: Neblina exibindo os serviços disponíveis	142
Figura 6.5: Neblina disponibilizando remotamente o serviço para o usuário	142
Figura 6.6: O GerNU cria a imagem e disponibiliza o serviço	143
Figura 6.7: Criação e provisão do serviço a partir do GerNU.....	144
Figura 6.8: Especificação de software	145
Figura 6.9: Especificação de hardware	146
Figura 6.10: Especificação dos parâmetros de qualidade	146
Figura 6.11: Especificação da prioridade dos requisitos	147
Figura 6.12: Login e cadastro de usuários.....	147
Figura 6.13: Composição de custos	148
Figura 6.14: Avaliação de usabilidade	149
Figura 6.15: Avaliação da corretude	150
Figura 6.16: Avaliação do tempo de resposta.....	150
Figura 6.17: Avaliação do nível de flexibilidade para especificação do serviço	151
Figura 6.18: Avaliação da metodologia de negociação.....	151

Figura 6.19: Comparativo de valores durante a negociação 152

LISTA DE TABELAS

Tabela 4.1: Atividades dos artigos sobre SLA.....	61
Tabela 4.2: Atividades dos artigos sobre QoS	67
Tabela 4.3: Atividades dos artigos sobre Modelos	75
Tabela 5.1: Disponibilidade.....	119
Tabela 5.2: Políticas de Backup.....	120
Tabela 5.3: Monitoramento do Ambiente Virtual	121
Tabela 5.4: Variabilidade	122
Tabela 5.5: Configurações de hardware – Padrão do GerNU	124
Tabela 5.6: Configurações de hardware - Padrão da Amazon.....	124
Tabela 5.7: Escala de [Saaty, 1986] para o AHP.....	127
Tabela 5.8 : Matriz exemplo de uma tabela AHP	127
Tabela 5.9: Critério AHP do GerNU	129
Tabela 5.10: Matriz de decisão	130
Tabela 5.11: Matriz Normalizada	130
Tabela 5.12: Cálculo autovalor	131
Tabela 6.1: Perfil dos usuário	149

1. INTRODUÇÃO

Esta tese apresenta uma abordagem para especificação e aquisição de serviços em nuvens computacionais que inicialmente contempla a definição de um processo para aquisição de serviços, levando em consideração os requisitos específicos que o usuário deseja contratar. Neste processo, a especificação de *hardware*, *software* e atributos de qualidade poderá ser feita de forma flexível, para atender exatamente as reais necessidades do cliente. Após a especificação do serviço, toda a infraestrutura necessária para o serviço ser disponibilizado será dinamicamente construída, sendo o serviço automaticamente instanciado com o objetivo de que todo o processo aconteça com mínima intervenção humana.

Posteriormente a disponibilização do serviço, o objetivo de um provedor deve ser garantir a qualidade do mesmo a partir de uma eficiente utilização dos recursos disponíveis e principalmente monitorando parâmetros de qualidade relevantes para o contexto de nuvens.

Neste capítulo serão apresentados: a motivação para o desenvolvimento deste trabalho, a descrição do problema abordado e a definição específica dos objetivos. Além destes, as contribuições pretendidas ao longo de todo o processo de pesquisa. Ao final do capítulo, será descrito como está organizada o restante desta tese.

1.1 MOTIVAÇÃO

A partir da Estratégia Nacional de Ciência, Tecnologia e Inovação (CT&I) 2012 - 2015 [MCTI, 2011] são enumerados um conjunto de desafios a serem superados pelo Brasil na área de CT&I e as ações a serem empreendidas para sua superação.

Entre os desafios citados, destaca-se a dificuldade do Brasil em transformar novos conhecimentos em inovações, visando a melhoria de bens e serviços, considerando-o como um dos grandes óbices para a redução da defasagem científica e tecnológica, que nos separa das demais nações desenvolvidas. Este

documento estipula, como objetivo a ser atingido na área de Tecnologia da Informação e Comunicação (TIC), a necessidade de “se promover uma ampla integração das tecnologias da informação e a modernização da indústria brasileira”; estabelecendo como estratégias associadas ao fortalecimento do setor nacional de TICs, os pontos a seguir:

- O desenvolvimento de infraestrutura avançada de TIC, visando ampliar a capacidade de armazenamento e acesso a serviços de computação em nuvem (cloud computing), para universidades e centros de pesquisa no país; e
- O desenvolvimento de um projeto para ampliar a capacidade instalada no país de computação de alto desempenho, supercomputação, expandindo o uso das tecnologias relacionadas para os diversos setores da pesquisa científica e de aplicações industriais avançadas.

Estas duas estratégias tratam de tecnologias, que hoje estão intimamente ligadas, na medida em que diversos artigos e trabalhos têm se dedicado a analisar como os recursos existentes em um ambiente de nuvem podem complementar e contribuir para as necessidades específicas da computação de alto desempenho.

Alem das estratégias específicas, é ressaltada a necessidade relativa a formação de recursos humanos de alto nível, colocando-a como requisito indispensável ao reforço e garantia da base de conhecimento existente e inovações futuras.

Dentro deste contexto podemos destacar o estudo encomendado pelo *National Center for Manufacturing Sciences* [Nacional, 2011] a *Intersec360 Research* [Snell et al., 2011a] identificando que apenas 8% das empresas de manufatura, com menos de 100 empregados, utilizam algum tipo de recurso de *High Performance Computing* (HPC) em apoio a seus processos de produção.

Este mesmo percentual foi percebido em empresas na faixa de 100 a 500 empregados. O estudo também identificou que 72% das empresas que utilizam alguma ferramenta computacional de modelagem, vislumbram a perspectiva de que o uso de recursos e ferramentas mais complexas aumentaria a sua produtividade e competitividade, sendo os custos dos recursos e a qualificação técnica necessária

para este uso, os principais fatores limitadores.

Outro ponto importante a ser destacado neste cenário, é a mudança da base da infraestrutura de comunicação de Ethernet para tecnologia de camada *Infiniband* [Snell et al., 2011b], motivada em parte pela redução do custo, mas, sobretudo pela necessidade crescente do aumento da largura de banda, da redução da latência e da capacidade de gerenciamento dos pacotes.

Um estudo do Departamento de Energia Americano (DoE), no período de 2009 a 2011, junto aos seus dois maiores centros de HPC, o *Argonne Leadership Computing Facility* no *Argonne National Laboratory* e o *National Energy Research Scientific Computing Center (NERSC)* no *Lawrence Berkeley National Laboratory*, teve por objetivo determinar a contribuição do uso da computação em nuvem em apoio a pesquisa científica, dentro dos recursos do DoE.

Dentre as conclusões e resultados constantes em seu relatório final, “The Magellan Report on Cloud Computing for Science” [Energy Department U.S., 2011]; cita o levantamento realizado entre seus usuários, onde lhes foi perguntado quais fatores consideravam ser os mais atrativos para o uso de recursos em nuvem.

Esta pesquisa levantou que 79% dos usuários viam a possibilidade de acesso a recursos adicionais como o elemento mais atrativo, 59% consideravam ser a possibilidade que o ambiente lhes dava para controle dos recursos, 52% consideravam a possibilidade de compartilhar os recursos com seus colaboradores e o mesmo percentual pela facilidade de operação quando comparado a um cluster do DoE (foi permitida mais de uma opção na pesquisa).

Este mesmo relatório, entretanto, relata a necessidade de uma maior maturidade nas aplicações de suporte a este ambiente, e no caso do DoE, a atual infraestrutura já comportava o suporte e os serviços propostos pelas nuvens computacionais, mas não descartava a possibilidade dele vir a servir no futuro.

Já o projeto *Helix-Nebula - Scientific Cloud Computing Infrastructure for Europe* [Cern-Esa, 2011], segue a possibilidade aberta no relatório *Magellan*, na medida em que estabelece como objetivo: a criação de uma infraestrutura e ambiente de nuvem, capaz de oferecer suporte a múltiplas organizações, baseado

nas necessidades específicas da comunidade de pesquisa Europeia e agências espaciais.

Este projeto baseia-se numa forte colaboração das iniciativas públicas e privada, que juntas verificam as possibilidades do uso de computação em nuvem em apoio as instituições e projetos de pesquisa. Para tanto, firmou como um dos compromissos iniciais, a identificação de um pequeno número de projetos chaves para servirem de modelo a estas parcerias, sendo o de Astronomia o primeiro a ser sugerido.

O objetivo a partir do projeto é: desenvolver, explorar e utilizar a infraestrutura de nuvem, inicialmente com base nas necessidades das organizações de pesquisa Europeias, permitindo a inclusão de outras organizações (governo, negócios e sociedade) a medida que surgirem os requisitos.

Observa-se, portanto neste projeto, como principais benefícios aos seus usuários, a sua independência em relação aos provedores de nuvem comerciais, garantindo a interoperabilidade e a garantia de serviços, além da segurança e transparência. Quanto aos provedores de serviços e infraestrutura, que se associarem, estes terão a possibilidade de estar em contato com um amplo ambiente de produção, teste, desenvolvimento tecnológico avançado, recursos governamentais e a garantia de um número inicial mínimo de usuários.

Nota-se que apesar das limitações ainda existentes para do uso do ambiente de nuvem em apoio a aplicações científicas e as demais atividades relacionadas a aplicações de alto desempenho, um ambiente de nuvem pode trazer benefícios a uma grande gama de usuários, na forma de acesso a recursos sob demanda, acesso remoto e principalmente a consolidação e otimização do uso destes recursos, considerando-se a possibilidade de compartilhamento pelos diversos usuários.

Considerando o contexto de nuvens computacionais, existem vários desafios carentes de pesquisa. Um destes desafios, de grande importância neste cenário, está relacionado com o controle da qualidade dos serviços disponibilizados, podendo o resultado qualitativo, bom ou ruim, ser reflexo direto da utilização dos recursos da nuvem.

A utilização de mecanismos automáticos com fácil usabilidade para gerar os serviços, monitorar os recursos da nuvem e a qualidade dos serviços podem gerar um maior nível de satisfação para os clientes e ainda criar um ambiente mais facilmente gerenciado e menos suscetível a erros para os administradores.

Outro desafio que pode ser comumente observado no âmbito da computação em nuvem é a inexistência de um padrão para o processo de especificação e aquisição de serviços. Cada provedor, conforme sua conveniência define como este processo deve acontecer, sendo importante para o paradigma de nuvens a definição de soluções para acesso automatizado aos serviços na nuvem.

Desta forma, contribuições relacionadas a estes aspectos podem promover um enriquecimento deste paradigma. A seguir são relacionados alguns destes desafios, embora sejam delimitados pelo escopo do problema abordado neste trabalho que será devidamente descrito posteriormente.

1.1.1 A Computação em Nuvem

A proposta da computação em nuvem é extremamente desafiadora porque representa uma grande quebra de paradigma, visto que até muito pouco tempo atrás, empresas e pessoas físicas utilizavam exclusivamente os recursos computacionais de forma proprietária, sendo os donos responsáveis pela gestão, manutenção e atualização dos recursos computacionais que dispõem.

Modelos semelhantes ao proposto pelas companhias de energia elétrica, que tem como objetivo fornecer energia de acordo com a demanda dos usuários é utilizado pela maioria dos serviços de utilidade pública, como água, luz e telefone. Embora sejam serviços fundamentais, tenham grande relevância para o dia a dia da sociedade, apresentem grande frequência de utilização e estejam teoricamente disponíveis a qualquer momento, o modelo de cobrança utilizado é baseado na quantidade consumida e não simplesmente por estarem disponíveis.

Analogamente, a infraestrutura de uma nuvem deve permitir que serviços sejam disponibilizados a qualquer momento e em qualquer lugar. Os sistemas de gerenciamento da nuvem devem controlar o uso destes recursos por meio de

medições que consideram cada tipo de serviço provido. Além disto, a ideia da cobrança pelo uso também foi absorvida pela computação em nuvem [Vaquero et al., 2009], representando uma característica muito importante para o crescimento deste paradigma.

A utilização de especialistas como provedores de recursos computacionais corresponde a proposta básica da computação em nuvem. Neste contexto, o fornecimento dos recursos é abstraído de forma que somente especialistas sejam responsáveis por seu gerenciamento e manutenção, além dos mesmos serem disponibilizados como serviços [Carr, 2008]. Desta forma, a provisão de recursos precisa ser vista em várias camadas, onde cada camada representa um gênero específico de recursos que podem ser providos de diferentes formas.

Uma nuvem também pode ser considerada uma metáfora para a internet, sendo baseada em abstrações que ocultam a complexidade de infraestruturas, onde cada parte é disponibilizada como serviço e hospedada em centros de dados que utilizam *hardware* compartilhado para computação e armazenamento [Buyya et al., 2009].

A computação em nuvem ainda pode ser vista como um paradigma com a ilusão de recursos infinitos, que estarão disponíveis sempre que houver necessidade. Para sua utilização são necessários poucos investimentos iniciais em infraestrutura e existe a possibilidade de contratação de recursos computacionais por prazos curtos e específicos [Armbrust, 2009].

Atualmente, a computação em nuvem, se apresenta como o cerne de um movimento que propõe profundas transformações no mundo da tecnologia, cujo objetivo é prover serviços sob demanda com o pagamento baseado em uso [Buyya et al., 2008]. Esta abordagem propõe alcançar os mais variados tipos de usuários através da disponibilização de diversas formas de serviços computacionais.

Um dos grandes desafios no ambiente de nuvens é controlar de forma eficiente o uso dos recursos computacionais, além de certificar-se que aquilo foi acordado com os clientes está sendo de fato cumprido. A utilização de Acordos em Nível de Serviço ou *Service Level Agreements* (SLA) pode delimitar de forma clara a responsabilidade de cada parte, bem como definir, de forma inequívoca, os

requisitos que estão sendo contratados, para que em avaliações periódicas, seja possível verificar-se o cumprimento dos contratos e a eficiência dos recursos.

Para uma eficiente utilização de um SLA, se faz necessário uma adequada identificação de parâmetros que possam determinar aspectos qualitativos do ambiente, que possam ser utilizados durante o processo de monitoramento e para determinar a Qualidade dos Serviços ou *Quality of Service* (QoS).

1.1.2 Acordos em Nível de Serviço (SLA)

Percebe-se que é fundamental identificar parâmetros que possam ser adequadamente utilizados para qualificar o ambiente e os serviços de uma nuvem computacional. Estes parâmetros devem estar descritos em um SLA, bem como os valores limítrofes, tanto o mínimo quanto o máximo, que sejam adequados para atender as restrições dos clientes.

Desta forma, como o objetivo de transformar as vantagens prometidas pela computação em nuvem em benefícios reais, os serviços devem ser acompanhados por definições exatas do que propõe, além da especificação de suas condições de uso. Estas condições podem ser especificadas com a utilização de acordos de nível de serviço.

Observa-se que cada vez mais empresas estão incorporando serviços de tecnologia da informação (TI) baseados em nuvem, por este motivo, confiabilidade e segurança tem se tornado fatores cruciais na gestão dos negócios. Essas condições devem ser entendidas pelos prestadores e usuários dos serviços, devendo ser negociáveis, para que os serviços possam atender aos requisitos dos clientes e sejam gerenciados a partir da perspectiva do provedor de serviços.

Normalmente, os SLAs são especificados apenas em alto nível, entre o prestador de serviço e o usuário. Estas especificações podem ser usadas para verificação se um serviço está em conformidade com os termos do SLA. Havendo violações, multas ou indenizações podem ser aplicadas. Entretanto, geralmente, elas não dão subsídios para que os prestadores planejem suas infraestruturas de TI e também não fornecem a compreensão de por que uma dada violação do SLA pode

ter ocorrido.

Este fato acontece porque os termos de garantia do SLA, em grande parte, não estão explicitamente relacionados com métricas de desempenho real ou parâmetros de configuração. Por isto é difícil para um prestador de serviço obter parâmetros de configuração adequados e métricas de monitoramento para um SLA em nuvem. Esta ligação que falta entre SLAs e as métricas e parâmetros é um grande obstáculo para o gerenciamento em termos de planejamento, previsão ou reconfiguração de processos [Butler et al., 2011].

Outro aspecto importante está relacionado com o fato de que uma solução em nuvem deve fornecer garantias para os serviços, por exemplo, de desempenho e disponibilidade. Sendo também necessário flexibilidade para adaptação diante de diferentes demandas, fato este comum devido a infraestrutura da nuvem dar suporte a múltiplos usuários. Este contexto dificulta a elaboração de estimativas e o fornecimento de garantias de QoS. Assim, estimar métricas de qualidade é um grande desafio nesses ambientes.

O provedor do serviço e o usuário devem definir as garantias de qualidade do serviço, podendo expressá-las por meio de um acordo de nível de serviço. Estes contratos devem claramente especificar um limiar que deve ser atendido para cada um dos requisitos identificados. As penalidades também devem ser especificadas para o caso de falhas. Os aspectos relacionados a qualidade de serviço podem ser abordados de diversas maneiras, por exemplo, considerando-se restrições de desempenho ou ainda a seleção dinâmica de um serviço que atenda determinadas especificações [Entrialgo et al., 2011].

De forma clara, existem riscos e desafios que precisam ser resolvidos. Do ponto de vista do usuário, não existem formas padronizadas para expressão e negociação de condições não funcionais. Na perspectiva do provedor de serviços, criar ofertas de serviços personalizados, negociar individualmente com os clientes, e ainda transformar requisitos de negócios em provisionamento específico consome tempo e recursos valiosos.

Sendo assim, percebe-se a necessidade de uma abordagem global de gestão de SLA, que possa ser facilmente utilizado em diferentes cenários e domínios, sendo

que as características de um SLA podem abranger vários domínios não funcionais, como segurança, disponibilidade, desempenho e confiabilidade, dentre outros.

Apesar de não haver grande complexidade no gerenciamento de um SLA com poucos serviços, isto se torna um problema complexo quando é necessário executar SLAs para milhares de aplicações que utilizam recursos compartilhados, pois a utilização de recursos dedicados normalmente não é economicamente praticável [Yang et al., 2009].

1.1.3 Qualidade dos Serviços (QoS)

Uma das razões o sucesso da computação em nuvem é que dado a sua natureza, sua aplicabilidade pode ser voltada tanto para o contexto acadêmico quanto para o contexto de negócios, podendo ser verificada sua eficiência em problemas reais.

Entretanto, normalmente as propostas para provisionamento de recursos têm sido baseadas em políticas de melhor esforço, onde é feito o possível para atender uma requisição, porém se não houver possibilidade, a mesma será descartada, logo não são levados em conta acordos em nível de serviço e a qualidade dos serviços [Distefano et al., 2011].

Existem muitas aplicações distribuídas na nuvem sendo executadas na internet, sendo que os clientes e os provedores usualmente estão dispersos globalmente. Muitas dessas aplicações requerem determinados níveis de QoS para garantir a satisfação dos seus usuários. Entretanto, garantir ou prever QoS em redes globais diversificadas, que suportam serviços complexos, é uma tarefa que ainda precisa de refinamento para ser resolvida, certamente atrelado ao amadurecimento da computação em nuvem [Pedersen et al., 2011].

Esta crescente utilização da nuvem para construção de aplicações que empregam componentes distribuídos revela a necessidade urgente de uma forma eficiente de avaliação da qualidade dos componentes da nuvem [Zhang et al., 2011].

Por este motivo, abordagens que avaliam e classificam a qualidade dos componentes de uma nuvem são importantes para auxiliar durante o processo de

seleção de componentes, isto considerando um conjunto de componentes com funcionalidades equivalentes. Além disto, a classificação dos componentes ajuda na identificação de componentes com baixo desempenho, importante para aplicações complexas que normalmente incluem grande quantidade de componentes [Zheng et al., 2010].

Ainda considerando sistemas distribuídos, [Vinek et al., 2011] afirma que antes da aplicação de qualquer estratégia para otimização da seleção de serviços e fontes de dados, é preciso haver uma análise em termos de métricas de QoS, de forma que seja possível realizar uma comparação com as estatísticas que descrevem os requisitos da aplicação.

O fato é que, devido a natureza dinâmica e heterogênea da computação em nuvem, alcançar níveis satisfatórios de QoS é um grande e relevante desafio, tanto para satisfazer as obrigações dos provedores quanto para as necessidades dos usuários [Chen and Zang, 2009], [Chen and Yang, 2008]. Ainda precisando-se levar em conta que requisitos de QoS podem possuir múltiplas dimensões, sendo então necessário o gerenciamento de QoS que considere componentes diversificados [Liu et al., 2011]. De forma geral, as principais dimensões de QoS em um ambiente de nuvem incluem: desempenho, confiabilidade, segurança, disponibilidade, acessibilidade, fidelidade, integridade, custos, dentre outros [IBM, 2011], [Yu and Buyya, 2005].

Considerando o aspecto de disponibilidade, como uma evolução natural dos sistemas distribuídos, a computação em nuvem pretende atender as transformações nos domínios das aplicações e nos requisitos de sistemas. Sendo assim, a virtualização de recursos é uma das chaves de sucesso do paradigma de nuvens, normalmente utilizado para hospedar diversas aplicações dos usuários em servidores multicore, porém, também apresenta novos desafios a serem investigados, incluindo alguns no contexto de QoS, sendo alguns semelhantes aos enfrentados na Computação em Grade [Armstrong et al., 2009].

As tecnologias de virtualização utilizadas pelos provedores de computação em nuvem podem introduzir interferências no desempenho das cargas de trabalho, de forma a afetar significativamente os parâmetros de QoS [Nathuji et al., 2010].

A qualidade dos serviços também pode ser considerada um requisito não funcional, podendo ser utilizada como base para seleção de serviços na nuvem, objetivando elevar o nível dos mesmos e ainda influenciar em sua precisão e confiabilidade [Tao et al., 2010]. Desta forma, a disponibilidade de um monitoramento de QoS seguro e confiável é importante para o crescimento da computação em nuvem, uma vez que será possível garantir que os usuários recebam integralmente os serviços contratados na nuvem. Este fato aumenta o nível de confiança dos serviços na nuvem, garantindo o crescimento deste paradigma [Romano et al., 2011].

Visto que, uma nuvem disponibiliza serviços para diversos usuários ao mesmo tempo, e que muitos destes usuários possuem diferentes requisitos de QoS, estratégias que consideram múltiplas restrições de QoS são profundamente relevantes para nuvens [Xu et al., 2009]. Neste sentido, uma vez que na prática, os recursos computacionais são limitados, é muito importante que os provedores de nuvem gerenciem eficientemente os seus recursos. Desta forma, um modelo eficiente de reconfiguração e alocação de recursos que satisfaça as solicitações de vários usuários é muito importante, sendo possível coletar e analisar informações de confiabilidade, que normalmente são baseadas nos históricos dos servidores da nuvem. Assim, com a utilização de políticas adequadas e a partir dos recursos disponíveis, poderá ser possível atender as solicitações de serviço [Kim et al., 2010].

Normalmente, uma aplicação pode ser hospedada em uma plataforma de execução construída a partir de recursos reais e/ou virtuais em uma nuvem. Neste contexto, os requisitos de QoS podem ser especificados em um SLA que estabelece as restrições da aplicação em relação a sua plataforma de hospedagem.

Através do monitoramento dos parâmetros de QoS o SLA poderá ser verificado, havendo alguma violação que degrade algum dos atributos de qualidade, a plataforma poderá ser dinamicamente re-configurada para incorporar recursos adicionais, e de forma inversa, se o SLA estiver sendo honrado e recursos da plataforma estiverem sendo desperdiçados por falta de utilização, também poderá ocorrer uma reconfiguração na plataforma para liberar estes recursos [Ferretti et al., 2010].

Para a computação em nuvem alcançar um sucesso pleno, faz-se necessário que, tanto os clientes quanto os fornecedores tenham certeza de que os acordos em nível de serviço assinados estão efetivamente apoiando da melhor forma suas respectivas atividades de negócio. Atualmente, conforme [Goiri et al., 2010] a utilização de SLAs não tem estabelecido bons níveis de confiança, especialmente quando os provedores terceirizam recursos para outros provedores. Estes provedores de recursos tipicamente suportam métricas muito simples ou métricas que dificultam a exploração eficiente dos seus recursos.

1.1.4 Abordagem Dirigida a Modelos

A utilização de modelos no contexto computacional é bastante difundida, e observando-se outras áreas da computação, foi possível perceber diversos benefícios advindos da utilização de modelos, tais como a padronização de processos e intercâmbio de informações, viabilizada através da realização do mapeamento entre estruturas distintas, dentre outros. Por isto, esta abordagem será utilizada durante o desenvolvimento deste trabalho.

Modelos elevam o nível de abstração do desenvolvimento de sistemas, ajudando no planejamento e entendimento dos mesmos, sendo que a importância do uso de modelos no desenvolvimento de software é um fato comprovado [Selic, 2003] [Fowler, 2004]. Uma abordagem que cresceu bastante a partir dos anos 2000, foi a da geração automática de código fonte a partir do modelo do sistema, e o desenvolvimento dirigido a modelos segue esta proposta, promovendo o modelo a artefato principal do desenvolvimento de software em detrimento do código.

Considerando-se o contexto de desenvolvimento de software, pôde-se observar a exigência de uma série de qualidades para garantir a longevidade de um sistema. Uma destas qualidades está relacionada com a premissa sempre presente no mundo do desenvolvimento de software, as mudanças, que especificamente estão relacionadas com a adaptabilidade, tanto em relação aos requisitos quanto ao ambiente em que a aplicação está ou será inserida.

Esta necessidade de adaptação é comumente observada em um ambiente de nuvens computacionais, por isto, observou-se que um dos paradigmas que possui

várias contribuições na assistência ao desenvolvimento de *software*, os modelos, podem ser utilizados para ajudar a resolver alguns dos desafios em uma nuvem.

No contexto deste trabalho, a utilização de modelos será muito relevante devido a flexibilidade para o armazenamento e manipulação de informações tais como dados dos usuários, especificação dos requisitos, dentre outros. Os modelos podem ser utilizados para automatizar processos complexos no ambiente de nuvens computacionais, além de agregar flexibilidade com a utilização de mecanismos de importação e exportação dos componentes de sua estrutura, permitindo então que um ambiente de trabalho possa ser migrado de um provedor para outro.

1.2 DEFINIÇÃO DO PROBLEMA

Considerando o contexto de nuvens computacionais e a partir da percepção que o processo de aquisição e provisionamento de serviços comumente encontrado nos grandes provedores de serviço atuais é inflexível, sendo impossibilitado ao usuário a manifestação de sua exata opinião em relação aos requisitos que deseja contratar para seu serviço, representando este cenário uma grande desafio que não possui uma solução adequada, e obviamente desejando criar uma perspectiva realística, se define-se o escopo para este trabalho destacando-se os três subproblemas como objetivos principais:

a) Padronização do Processo de Especificação e Aquisição dos Serviços

O que pode ser comumente observado no âmbito da computação em nuvem é a inexistência de um padrão para a especificação e aquisição de serviços. Cada provedor, conforme sua conveniência define como este processo deve acontecer. Inclusive, a *National Science Foundation* (NSF) dos Estados Unidos da América, confirma esta realidade através do programa *Strategic Technology for Cyber Infrastructure* (STCI) [NSF, 2012], onde convoca iniciativas com propostas para padronizar e automatizar o acesso aos serviços na nuvem.

Certamente por motivos comerciais, não existe um formato de arquivo de imagem virtual largamente adotado por provedores de nuvem. Sendo assim, também é importante uma iniciativa para estabelecer um modelo que especifica os requisitos que o provedor precisa atender para criar um determinado serviço. Para os usuários, poderá servir como um guia para contratação de serviços. Desta forma, um serviço especificado para um determinado provedor poderá facilmente ser mapeado com o objetivo de ser implantado em qualquer outro.

b) Disponibilização Automática dos Serviços

A geração automática de serviços, com mínima ou nenhuma intervenção humana acelera e agrega qualidade ao processo de disponibilização de serviços. Além disto, a geração de um padrão estrutural para a disponibilização dos serviços cria a possibilidade de portabilidade de um serviço de um provedor para outro.

O ambiente heterogêneo e distribuído de uma nuvem computacional agrega uma série de tarefas para a disponibilização dos serviços, muitas delas com grande complexidade. A instanciação automática de serviços possibilitará uma menor intervenção humana, desejando-se chegar a nenhuma, assim, esta automação conduzirá a execução destes procedimentos com um menor índice de erros e em tempo reduzido.

c) Definição de Parâmetros para a Qualidade dos Serviços

Os serviços disponibilizados em nuvem precisam estar dentro dos parâmetros contratados pelo usuário. Por este motivo se faz necessário identificar parâmetros relevantes para este contexto, devendo ser realizado um monitoramento contínuo, de forma a avaliar se os parâmetros de qualidade dos serviços estão entre os limiares aceitáveis.

O fato é que a necessidade dos usuários não termina com a obtenção de um determinado serviço, é preciso que sejam estabelecidas

métricas que possibilitem a avaliação da qualidade dos serviços disponibilizados. Ou seja, não é suficiente resolver um problema, é preciso resolvê-lo considerando determinadas exigências especificadas pelo usuário. Este aspecto diz respeito ao gerenciamento de qualidade.

A proposta deste trabalho envolve o desenvolvimento de uma abordagem que apresenta uma solução para estes três problemas. A solução proposta viabilizará uma especificação de serviços baseada nos requisitos de *hardware*, *software* e parâmetros de qualidade, representando exatamente a necessidade do cliente.

O serviço personalizado será gerado a partir de um processo de negociação do SLA, fase onde serão gerados / extraídos as informações necessárias para a criação de todos os artefatos que darão o suporte indispensável para a criação dos serviços. Todo o processo de criação da máquina virtual que dará suporte ao serviço acontece de forma dinâmica e automática, sendo toda a infraestrutura criada do zero, visto que não existem imagens pré-fabricadas no ambiente da solução.

Após a disponibilização do serviço, o provedor deverá monitorar a qualidade dos mesmos a partir de parâmetro de qualidade propostos por nossa solução, sendo responsabilidade do provedor disponibilizar mecanismos com este objetivo.

A figura 1.1 exibe o cenário genérico do problema abordado neste trabalho, podendo ser observado, incluso na nuvem, o **GerNU**, produto deste trabalho que intenciona resolver eficientemente estas três importantes questões relacionadas ao ambiente de nuvens, tendo seu contexto de limitado a estas questões. Além disto, a figura 1.1 também retrata um cenário cotidiano do ambiente de nuvens computacionais, onde os usuários podem se conectar a nuvem através da internet, para então solicitarem os serviços desejados. Os recursos computacionais de suporte para estes serviços, tais como cpu, memória e armazenamento, normalmente são disponibilizados transparentemente através de técnicas de virtualização, disponibilizando-se então máquinas virtuais que irão efetivamente executar os serviços.

A demanda dos serviços não pode ser prevista com exatidão, pois é impossível determinar a quantidade de usuários e suas cargas de trabalho a cada momento, por isto, considerando a qualidade dos serviços, o gerente da nuvem

poderá auxiliar na determinação da possibilidade ou não de atender uma requisição e com o auxílio de um escalonador, identificar em qual *HOST* (que é máquina hospedeira, equipamento real que tem seus recursos virtualizados) o serviço será disponibilizado.

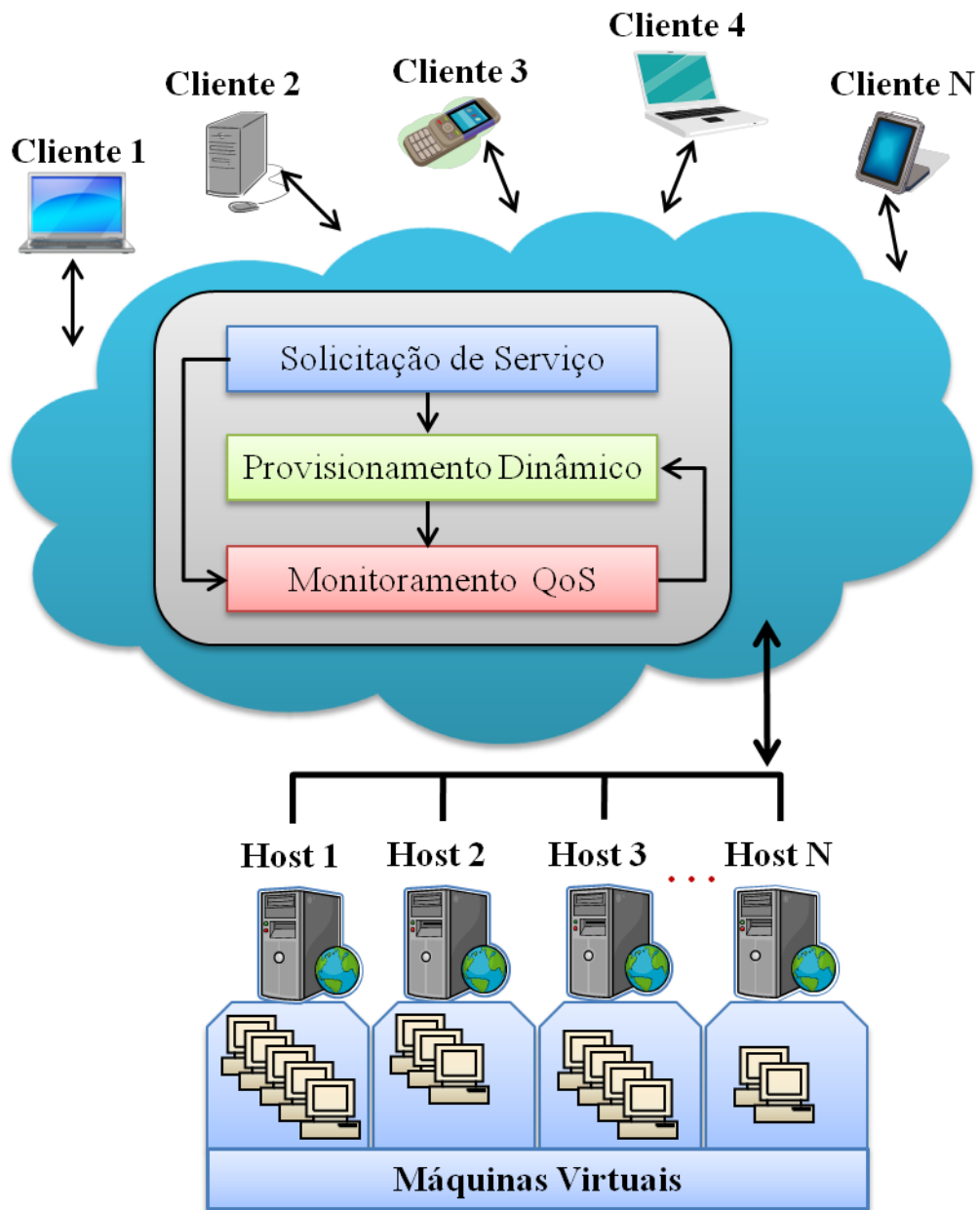


Figura 1.1: Cenário do Problema

A figura 1.2 exibe um exemplo do processo de virtualização, onde a partir dos recursos físicos do *HOST* foram criadas duas máquinas virtuais, cada uma delas com uma configuração específica de *hardware* que possibilite atender

eficientemente o serviço para o qual é destinada.

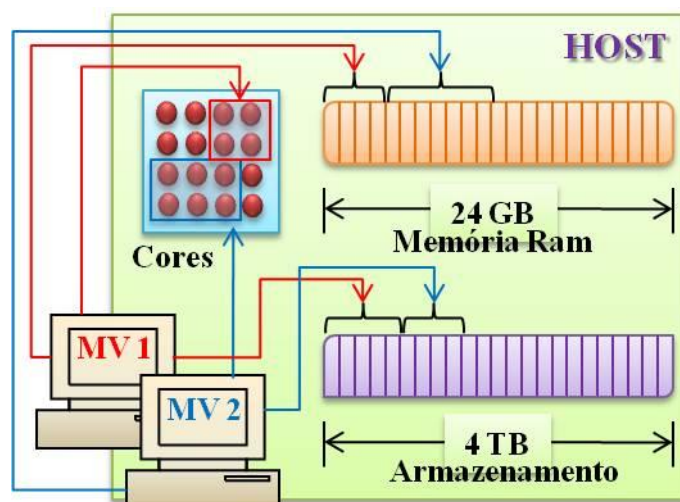


Figura 1.2: Processo de Virtualização

Havendo disponibilidade de recursos, será iniciado um processo automatizado para criação e disponibilização da máquina virtual, bem como todo suporte de software, tanto o sistema operacional quanto aplicações. Neste procedimento, a intervenção humana será mínima ou inexistente.

A atuação do sistema proposto, a partir de mecanismos do provedor de nuvem, continua através do monitoramento contínuo dos serviços que foram disponibilizados. Objetiva-se com isto garantir a qualidade dos mesmos através da verificação da dependabilidade do ambiente de trabalho virtual, fator este que engloba uma série de atributos que podem ser utilizados como parâmetros para qualificar os serviços e o ambiente da nuvem.

Desta forma, como propósito para este trabalho, pretende-se que o GerNU defina um processo para especificação e aquisição de serviços na nuvem que seja baseado nos requisitos dos usuários, considerando a QoS e utilizando modelos para o ambiente virtual, sendo então possível intercambiar serviços entre provedores mediante o mapeamento do modelo proposto e suas estruturas proprietárias.

Assim sendo, este trabalho foi desenvolvido baseado na seguinte hipótese: A definição de um processo para especificação e aquisição de serviços em nuvens computacionais baseado nos requisitos do usuário, permitindo que um serviço seja dinamicamente criado e ainda tenha sua qualidade monitorada, aumentará a

qualidade dos serviços disponibilizados em nuvens, incrementando o nível de confiança e a satisfação dos usuários na corretude do provisionamento de serviços, contribuindo para o amadurecimento deste paradigma.

Como forma de tratar esta hipótese, esta tese de doutorado objetiva fornecer um processo flexível para especificação e aquisição de serviços para nuvens, implementado através de uma aplicação denominada GerNU, cujo conjunto de funcionalidades propiciará o atendimento destes desafios. Construir o GerNU é uma tarefa desafiadora devido a complexidade inerente a um ambiente de nuvem e das exigências previstas para a solução.

1.3 OBJETIVOS

O objetivo geral deste trabalho é desenvolver uma abordagem para especificação e aquisição de serviços em nuvens computacionais, sendo que ela está focada em usabilidade e flexibilidade para especificação do serviço que será criado e disponibilizado automaticamente. Além destes, pretende-se identificar eficientemente parâmetros que irão garantir a qualidade dos serviços disponibilizados na nuvem, através da utilização de um modelo do ambiente para controlar e definir as atividades no sistema.

Nós propomos atributos advindos da dependabilidade, como, disponibilidade, integridade, segurança, dentre outros, que serão utilizados para garantir os níveis de qualidade oferecidos em cada SLA e que ainda podem identificar um serviço como comprovadamente confiável.

Essa proposta intenciona realizar a negociação de garantias de SLA durante o processo de aquisição de serviços, servindo o produto desta negociação como base para o modelo de ambiente que será utilizado para a alocação de recursos e utilizado para a verificação dos requisitos de forma a garantir as especificações do SLA.

1.3.1 Objetivos Específicos

Para atingir o objetivo principal proposto, foram definidos os seguintes

objetivos específicos:

1. Definir um processo flexível para a especificação e aquisição de serviços em nuvem baseado nos requisitos do usuário.
2. Identificar e propor parâmetros de qualidade que sejam relevantes para o contexto de nuvens computacionais.
3. Definir um processo de negociação que considere os aspectos mais relevantes para o usuário durante sua execução.
4. Definir um modelo de ambientes para nuvens, através do qual seja possível mapear os requisitos de *hardware* e *software* do usuário estabelecendo parâmetros que devem ser atendidos por um provedor.
5. Definir uma abordagem para o gerenciamento de SLA.
6. Definir um mecanismo para automação da construção e entrega de serviços na nuvem.
7. Projetar e implementar um *framework* que contemple todos os itens descritos anteriormente.
8. Analisar a eficiência da abordagem proposta através da realização de experimentos práticos, onde usuários poderão testar o ambiente e expressar sua opinião a respeito de aspectos como usabilidade, corretude, flexibilidade, dentre outros.

1.4 CONTRIBUIÇÕES

Este trabalho é inédito por considerar a perspectiva do usuário na definição das funcionalidades relacionadas ao provisionamento de serviços em nuvens. Além disto, não foi possível encontrar na literatura referências semelhantes ao processo automatizado de construção e disponibilização do serviço utilizado na proposta deste trabalho.

Assim sendo, este trabalho pretende contribuir com o desenvolvimento do paradigma de nuvens computacionais, criando mecanismos que validem e consolidem a utilização do mesmo, podendo ser destacados os pontos a seguir.

1. Uma abordagem flexível para especificação de serviços, que inicialmente poderá ser utilizada em qualquer infraestrutura de nuvem privada, podendo inclusive trabalhar em conjunto com um escalonador de recursos, de forma a atender de forma mais eficiente as requisições de serviços, onde a negociação das garantias do SLA acontece durante o processo de aquisição dos serviços.
2. Uma abordagem para a geração dinâmica de serviços em uma nuvem, que auxiliará tanto o provedor em relação as complexas tarefas relacionadas a provisão de serviços, quanto os usuários na identificação de requisitos e funcionalidades que poderão exigir do provedor, auxiliando inclusive na definição de qual provedor é mais adequado para atender suas necessidades.
3. A implementação da arquitetura e das abordagens propostas, sendo as mesmas baseadas na utilização de modelos.
4. Com a padronização da estrutura dos serviços, cria-se um padrão para o processo de aquisição de serviços, alavancando mecanismos que viabilizem a portabilidade de um provedor de nuvem para outro.
5. Revisão bibliográfica que descreva o estado da arte relacionado com os problemas abordados.

Além destas contribuições diretas, podemos destacar alguns benefícios na utilização da proposta deste trabalho, tanto do ponto de vista do usuário quando do provedor.

a) Benefícios para o Usuário

1. Ambiente de fácil manipulação, que não exige conhecimentos de como uma nuvem funciona ou aspectos relacionados com a manutenção de infraestruturas.
2. Especificação dos serviços com uma simples seleção de parâmetros sem a necessidade de instalação de *hardware* robusto no ambiente local do usuário.

3. Tempo reduzido para a utilização dos serviços.
4. Geração clara e objetiva de parâmetros relacionados aos custos dos serviços, possibilitando aos usuários fazerem uma previsão real de gastos a fim de avaliar com precisão a relação de custo/benefício da contratação dos serviços.
5. Garantia de qualidade dos serviços.

b) Benefícios para o Provedor

1. Ambiente com reduzida intervenção humana.
2. Tempo reduzido para a disponibilização dos serviços.
3. Otimização da utilização dos recursos.
4. Controle da qualidade dos serviços disponíveis para garantir a satisfação dos usuários.

1.5 METODOLOGIA ADOTADA

A metodologia utilizada para o desenvolvimento da abordagem proposta é apresentada na figura 1.3 e descrita a seguir.

Após uma extensa pesquisa bibliográfica nos temas relacionados com esta tese de doutorado, o primeiro passo consiste em definir um processo para aquisição de serviços em nuvem. Este processo tem a finalidade de atender a real necessidade dos usuários quanto a especificação dos requisitos de *hardware*, *software* e parâmetros de qualidade. O resultado do levantamento bibliográfico realizado está descrito nos capítulos 2, 3 e 4 e a definição do processo pode ser encontrada no capítulo 5.

O segundo passo consiste em definir um processo interativo de negociação do SLA que considere os fatores mais relevantes para o usuário na construção das propostas de valores. Será analisado o estado da arte em negociação e definido um método para construir o processo de negociação utilizado na aplicação. O processo de negociação proposto está documentado no capítulo 5, sendo a estado da arte encontrado no capítulo 3.

Uma vez formalizado o processo de negociação, o ambiente virtual do serviço deverá ser construído, constituindo-se como terceiro passo da metodologia. Para tanto se faz necessário definir uma arquitetura de *hardware* e *software* que dará subsídio a solução proposta. Todo o processo de automação da criação e entrega do serviço está descrito no capítulo 5.

O quarto passo consiste em validar a abordagem proposta. Para isto foram formuladas perspectivas reais de aplicação da proposta, sendo levantados benefícios e desafios na sua utilização, além da correção na provisão dos serviços especificados no ambiente da aplicação. A avaliação realizada está descrita no capítulo 6.

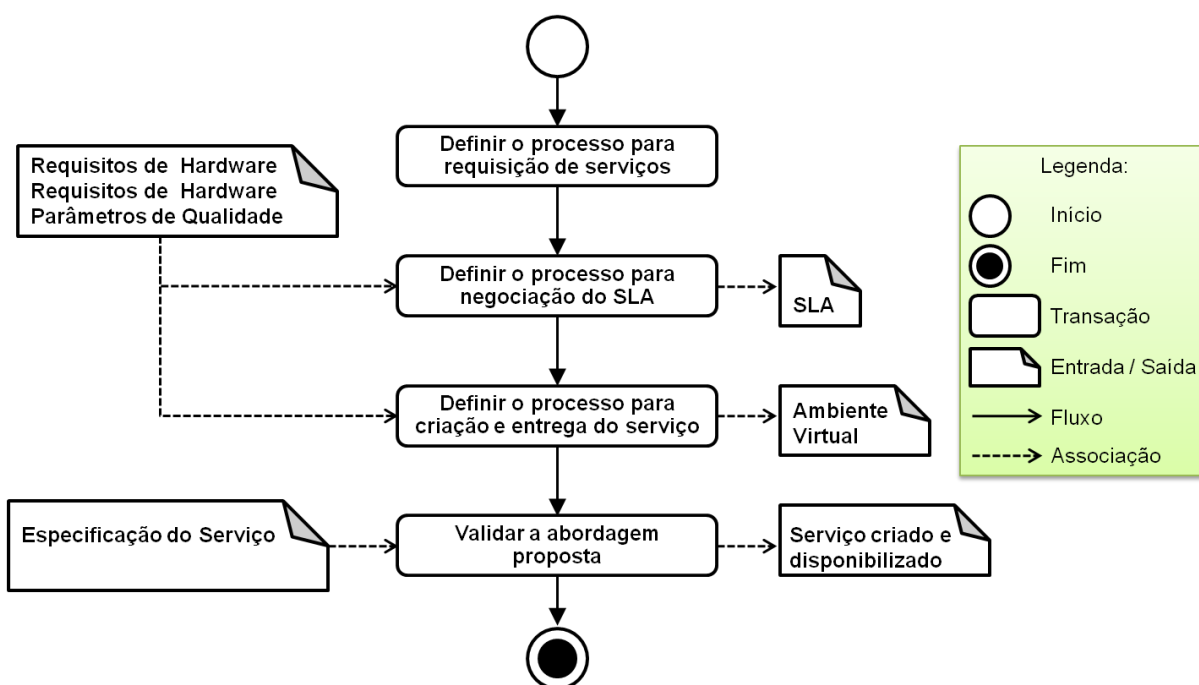


Figura 1.3: Metodologia para desenvolvimento da abordagem proposta

1.6 ORGANIZAÇÃO DO TEXTO

Os estudos e análises realizados no desenvolvimento desta pesquisa estão estruturados em quatro partes organizados em capítulos ilustrados na figura 1.4 descrita a seguir.

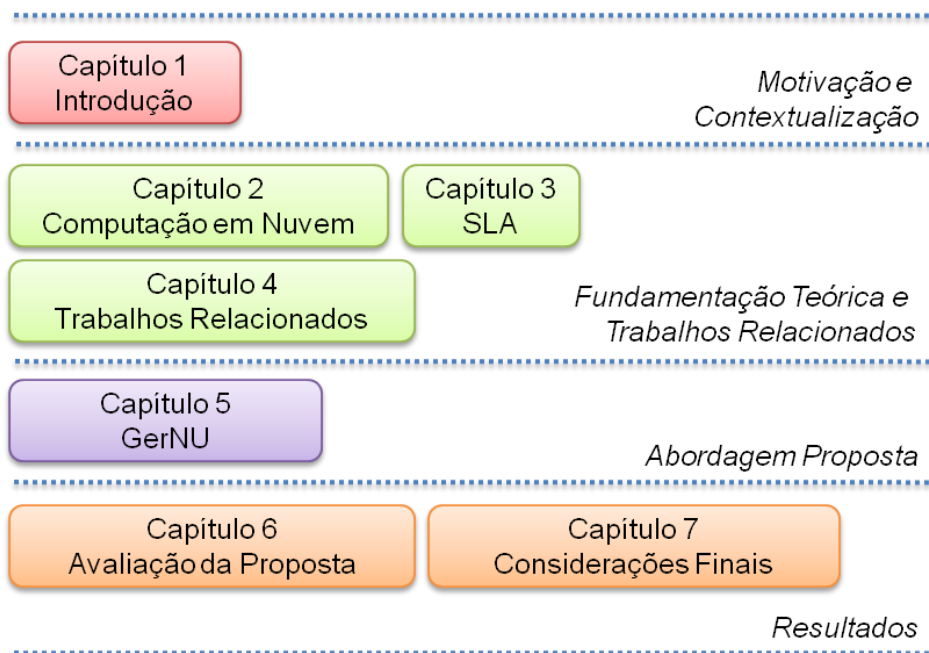


Figura 1.4: Organização da Proposta

O ambiente no qual a proposta deste trabalho está inserido é descrito no capítulo 2, sendo destacados aspectos relevantes relacionados ao paradigma da computação em nuvem.

No capítulo 3 foi descrito aspectos referentes aos contratos em nível de serviços, indispensáveis a um bom gerenciamento de uma nuvem computacional.

Alguns trabalhos que se relacionam com aspectos deste trabalho e subsidiaram as abordagens utilizadas no mesmo estão descritos no capítulo 4, provendo uma revisão bibliográfica sobre estes temas.

O capítulo 5 apresenta a abordagem proposta descrevendo sua arquitetura e funcionalidades e políticas de gestão, sendo o capítulo 6 utilizado para apresentar os testes e resultados alcançados com este trabalho.

Finalmente, o capítulo 7 resume o trabalho proposto, discute as contribuições alcançadas e indica algumas direções em que a pesquisa apresentada pode ser estendida.

2.COMPUTAÇÃO EM NUVEM

Este capítulo apresenta um *overview* sobre a computação em nuvens, abordando diversos aspectos inerentes a este paradigma com o objetivo de descrever sua história, conceitos, estrutura e principais modelos de serviço e implantação. Além disto, são apresentados alguns sistemas para computação em nuvem que estão disponíveis na literatura.

2.1 INTRODUÇÃO

A computação em nuvem tem se estabelecido nos últimos anos como uma importante plataforma de pesquisa que apresenta uma série vantagens e de estimulantes desafios. Tarefas como obtenção, compartilhamento, manipulação e exploração de grandes quantidades de dados são absolutamente comuns no cenário atual, porém a execução das mesmas demanda um grande volume de recursos. A computação em nuvem pode contribuir com este cenário a medida que pode disponibilizar de forma indefinida recursos de processamento, memória, armazenamento, dentre outros, para utilização imediata.

A disponibilidade sob demanda destes recursos agrega uma série de vantagens para as organizações e usuários, visto que preocupações com complexas instalações e manutenções de infraestruturas passam a ser de exclusiva responsabilidade dos provedores de serviços, que por sua vez podem compartilhar o custo com diversos usuários. Além disto, geram a possibilidade para que os usuários se concentrem exclusivamente nas regras dos negócios que lhes são pertinentes.

Este cenário de escalabilidade de serviços, processos e infraestrutura quase ilimitados não possui precedente e efetivamente melhora a flexibilidade relacionada a estruturas de tecnologia de informação, bem como pode diminuir o custo total dos negócios pelo provimento de serviços sob demanda, sendo que a proposta básica da computação em nuvem consiste na provisão de recursos computacionais sob a responsabilidade de empresas especializadas ou que seja abstraído o fornecimento deles em níveis que apenas especialistas venham a se preocupar em gerenciá-los e

mantê-los, e ainda que os mesmos sejam disponibilizados como serviços [Carr, 2008], representando uma quebra de paradigma em relação aos modelos existentes.

Os serviços devem ser concebidos considerando-se camadas distintas para cada tipo de serviço, onde abstrações ocultam a complexidade das infraestruturas, sendo estes comumente hospedados em centrais de processamento de dados que utilizam hardware compartilhado para a computação e armazenamento [Buyya et. al, 2009].

A figura 2.1 revela um ambiente comum da computação em nuvem, onde se pode observar diversos dispositivos interagindo, usando os recursos da nuvem, enviando e recebendo dados, além de exibir as camadas da arquitetura de uma nuvem computacional.

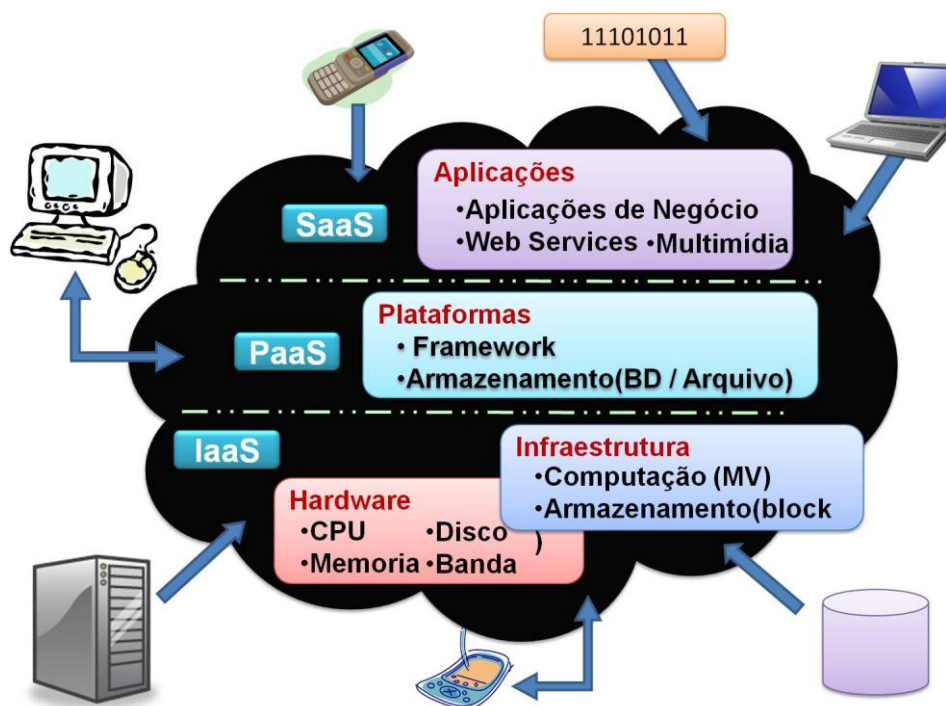


Figura 2.1: Cenário da Computação em Nuvem

Com o advento deste novo paradigma, surgem novos desafios e oportunidades pertinentes ao fornecimento e utilização deste modelo para que seja possível a transição do modelo tradicional. Além dos novos desafios, problemas já resolvidos podem emergir devido o novo cenário.

2.2 CONCEITOS FUNDAMENTAIS

Uma definição para Computação em Nuvem que seja consenso entre os diversos participantes do processo ainda não foi encontrada, portanto, a seguir são descritas algumas definições interessantes e pertinentes encontradas na literatura, iniciando com a definição absorvida por esta monografia. No contexto deste trabalho, a computação em nuvem pode ser definida como um paradigma que tem como objetivo a abstração no provimento de serviços, tais como Infraestrutura, Plataforma e *Software*, entre outros. A sua estrutura de suporte é mantida por um grande conjunto de recursos que podem ser indefinidamente escalados para suportar a demanda dos serviços disponibilizados.

Um modelo de computação onde as capacidades relacionadas a tecnologias da informação são escaláveis e elásticas, sendo que as mesmas são providas como serviços para os usuários finais através da internet, esta é uma definição disponibilizada pelo grupo Gartner em [Cearley, 2009].

Um ponto de vista que considera as características de hardware é fornecido em [Armbrust 2009], sendo a computação em nuvem definida como um paradigma com a ilusão de recursos infinitos, que estarão disponíveis sempre que houver necessidade. Ressalta-se que poucos investimentos iniciais em infraestrutura são necessários e existe a possibilidade de contratação de recursos computacionais por prazos curtos e específicos.

O trabalho [Buyya et al., 2008] considera que uma nuvem é um tipo de sistema paralelo e distribuído, consistindo de uma coleção de computadores virtualizados e interconectados que são provisionados de forma dinâmica e apresentados como um ou mais recursos computacionais unificados. Estes recursos são disponibilizados e controlados através de acordos relacionados aos serviços que são estabelecidos entre um prestador e um consumidor sendo definidos a partir de negociações entre as partes.

O termo computação em nuvem, segundo [Taurion, 2009], surgiu em 2006 em uma palestra de Eric Schmidt, da Google, sobre como sua empresa gerenciava seus datacenters. Hoje, computação em nuvem, se apresenta como o cerne de um

movimento de profundas transformações do mundo da tecnologia.

A nuvem é uma representação para a internet ou uma infraestrutura de comunicação entre componentes arquiteturais, baseada em uma abstração que oculta à complexidade da infraestrutura. Cada parte desta infraestrutura é provida como um serviço, e estes serviços são normalmente alocados em datacenters, utilizando hardware compartilhado para computação e armazenamento, [Sousa et al., 2009].

Após um estudo onde foram consideradas várias definições distintas para o conceito de computação na nuvem, os autores de [Vaquero et al., 2009] chegaram a definição de que nuvens são grandes repositórios de recursos virtualizados, tais como hardware, plataformas de desenvolvimento e software, que são facilmente acessíveis. Além disto, estes recursos podem ser configurados dinamicamente de modo a ajustar-se a diferentes cargas de trabalho com a intenção de otimizar sua utilização. O modelo de cobrança utilizado para a exploração destes repositórios está baseado no pagamento pelo uso.

Encerrando, porém longe de esgotar as definições disponíveis, apresenta-se a definição do *National Institute of Standards and Technology* (NIST) dos Estados Unidos, onde a computação em nuvem representa um conveniente modelo de acesso, sempre que for necessário, a um conjunto compartilhado de recursos computacionais configuráveis, tais como, redes, servidores, armazenamento, aplicações e serviços, que podem ser disponibilizados rapidamente, e para isto o esforço de gerenciamento e interação com o provedor dos serviços é mínimo ou nenhum.

2.3 CARACTERÍSTICAS ESSENCIAIS

O NIST definiu algumas características que descrevem o modelo de computação em nuvem, porém dado o amadurecimento e enriquecimento do mesmo, para este trabalho, outras informações e valores foram agregados. Estas características representam algumas das vantagens deste paradigma e servem também para melhor identificar e distinguir a computação em nuvem de outros

paradigmas.

2.3.1 Virtualização de Recursos

Existem muitas tecnologias já amadurecidas que proporcionam a virtualização de recursos computacionais, dentre elas, pode-se citar as máquinas virtuais, virtualização de redes, de memória e de armazenamento de dados. Graças a este mecanismo, possibilita-se uma separação dos serviços de infraestrutura dos recursos físicos como hardware ou redes, sendo então possível, por exemplo, tratar em uma camada inferior os aspectos relativos a localização de recursos, tornando então transparente este contexto para as demais camadas na estrutura da nuvem. Com esta abstração, os recursos podem ser disponibilizados e utilizados como serviços utilitários, sem a necessidade de uma manipulação direta do hardware.

2.3.2 Serviços sob Demanda

O cliente pode unilateralmente, conforme sua necessidade, requerer maior ou menor quantidade de recursos computacionais, tais como, capacidade de processamento, armazenamento ou largura de banda, estes recursos devem ser disponibilizados de forma automática, sem a necessidade de interação humana com o provedor de cada serviço. Um provedor de recursos computacionais idealmente deve atender vários consumidores através de um modelo multi-clientes, utilizando diferentes recursos físicos e virtuais que podem ser atribuídos e reconfigurados dinamicamente de acordo com a demanda dos consumidores.

2.3.3 Independência de Localização e Amplo Acesso

Os recursos devem estar disponíveis através da rede e internet, estando acessíveis por meio de dispositivos computacionais padrões, promovendo sua utilização por plataformas heterogêneas, como por exemplo, telefones celulares, laptops, PDAs, etc. Os softwares clientes instalados para acessar a nuvem normalmente são muito leves como um navegador de internet. Desta forma, a nuvem, aparentemente, seria um ponto de acesso centralizado para as

necessidades computacionais dos seus usuários, estando disponível o tempo todo e em qualquer lugar.

2.3.4 Elasticidade e Escalabilidade

A elasticidade é uma característica inovadora da computação em nuvem. É a capacidade de disponibilizar e remover recursos computacionais em tempo de execução, independente da quantidade solicitada. Dentro deste contexto, temos a definição de escalabilidade, que está relacionada com o requisito de aumento da capacidade de trabalho através da adição proporcional de recursos.

Um prestador de serviços não pode prever como seus clientes usarão os serviços disponíveis, visto que, por exemplo, um cliente pode usar um serviço somente algumas vezes por ano, em épocas de pico, enquanto que outro pode usá-lo como uma plataforma de desenvolvimento principal para todas as suas aplicações. Desta forma, o serviço precisa estar disponíveis sete dias por semana, 24 horas por dia, além de ter sido concebido para escalar para cima em períodos de alta demanda e para baixo quando a demanda cai, podendo ser utilizado também quando usuários são adicionados ou quando as especificações do aplicativo mudam.

A capacidade de escalar é alcançada mediante a característica de elasticidade dos serviços da computação em nuvem. Enfim, para os usuários, os recursos parecem ser ilimitados e podem ser adquiridos em qualquer quantidade, ou seja, a demanda do usuário deve determinar a liberação e aquisição dos recursos e isto deve ser executado de forma rápida, transparente e sem intervenção humana.

2.3.5 Medição dos Serviços

Os serviços de utilidade pública, como água, luz, telefone, devido sua importância e frequência de utilização no dia-a-dia, devem estar disponíveis a qualquer momento, porém os consumidores pagam aos provedores destes serviços apenas a quantidade consumida durante um determinado período. Analogamente, os sistemas de gerenciamento da nuvem, controlam e aperfeiçoam o uso dos

recursos por meio de medições que consideram cada tipo de serviço provido.

O monitoramento agrega transparência tanto para o provedor quanto para o cliente, sendo que normalmente são utilizados contratos referentes aos serviços, para especificar as características dos serviços, parâmetros de qualidade e determinar os valores que serão cobrados. Estes contratos definem os níveis de disponibilidade, funcionalidade, desempenho e outros atributos relativos aos serviços incluindo inclusive penalidades para o caso de violação das regras por qualquer uma das partes.

2.3.6 Repositório de Recursos

Os provedores de recursos computacionais são organizados para atender múltiplos usuários através de um modelo multi-clientes. Para isto são utilizados diferentes recursos físicos e virtuais que podem ser atribuídos e configurados dinamicamente de acordo com a demanda de cada cliente. O usuário não conhece a localização física dos recursos computacionais, porém pode ser possível especificar sua prioridade de localização com relação ao país e ao centro de dados através do contrato.

2.4 MODELOS DE SERVIÇOS

O modelo conceitual encontrado com maior frequência na literatura é composto por três camadas, este modelo define um padrão arquitetural para soluções em computação em nuvem e pode ser visto na figura 2.2, que também exibe uma breve especificação de serviços compatíveis com cada camada. Na figura 2.3, pode ser observado exemplos de aplicações destacadas por tipo de camada.

2.4.1 Infraestrutura como Serviço (IaaS)

Representa a camada inferior do modelo conceitual, sua base, ela é composta por plataformas para o desenvolvimento, teste, implantação e execução de aplicações proprietárias. Segundo [Sousa et al., 2009], seu principal objetivo é tornar mais fácil e acessível o fornecimento de recursos, como servidores, redes,

armazenamento e outros que são fundamentais na construção de um ambiente sob-demanda podendo ser tanto sistemas operacionais quanto aplicativos. A infraestrutura é baseada na virtualização dos recursos computacionais que pode ser dinamicamente escalada para aumentar ou diminuir os recursos de acordo com as necessidades das aplicações. Em resumo, IaaS relaciona-se com a capacidade que um provedor tem de oferecer uma infraestrutura de processamento e armazenamento de forma transparente.

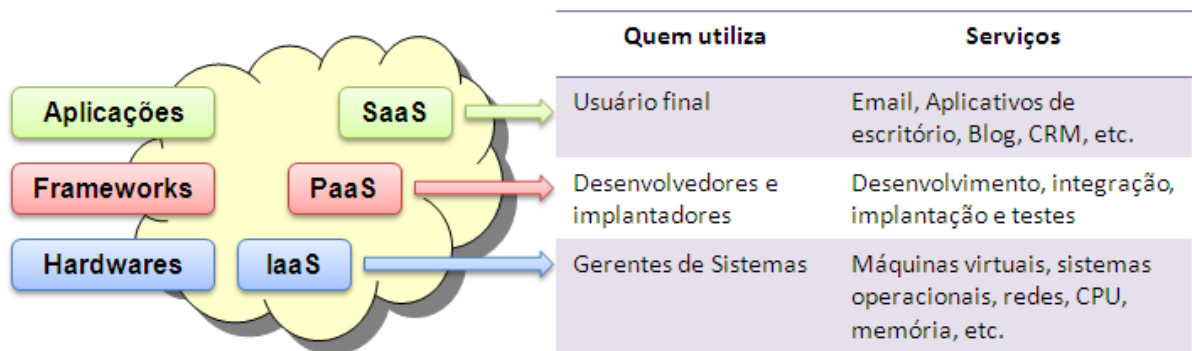


Figura 2.2: Modelo de Serviço

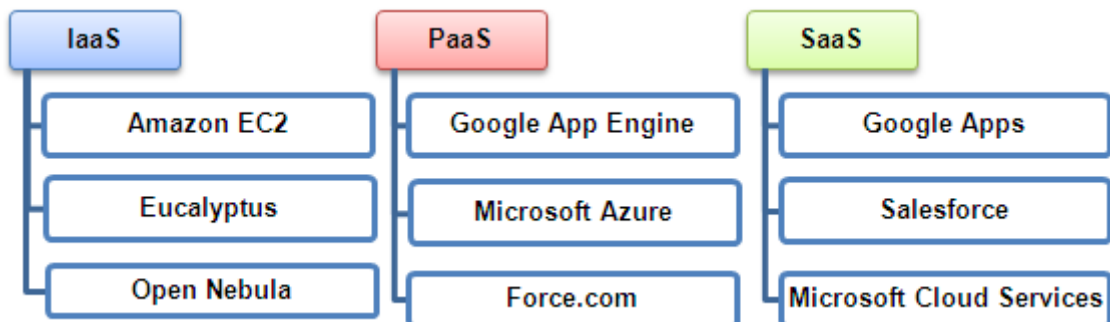


Figura 2.3: Exemplo de Serviços

Alguns exemplos de IaaS são o *Amazon Elastic Cloud Computing - EC2*, o *Elastic Utility Computing Architecture Linking Your Programs To Useful Systems - Eucalyptus* e o *Open Nebula*. É muito importante visualizar uma infraestrutura de tecnologia da informação como um conjunto, o *hardware*, a rede, o *software* básico, os links e tantos outros componentes e conceitos que compõem um parque tecnológico. Ou seja, infraestrutura como um serviço, representa um ambiente completo e gerenciado.

O modelo de entrega IaaS consiste de vários conceitos e componentes que foram desenvolvidos e amadurecidos ao longo dos últimos anos, entretanto utilizar estes componentes juntos, em um ambiente terceirizado e compartilhado revela vários desafios. Dentre eles, destacam-se a segurança e a privacidade, como os mais significativos. [Dawoud, 2010] descreveu uma série de componentes e conceitos fundamentais ao contexto de uma IaaS, sendo estes e outros apresentados a seguir.

2.4.2 Plataforma como Serviço (PaaS)

A provisão de serviços que possibilitem o desenvolvimento, testes, implantação, hospedagem e gerenciamento de aplicações objetivando suportar o ciclo de vida do desenvolvimento de aplicações é a proposta básica do modelo PaaS. É a camada intermediária do modelo conceitual, sendo composta por hardware virtual disponibilizado como serviço. Oferece tipos específicos de serviços como sistemas operacionais, banco de dados, serviços de mensagens, serviços de armazenamento de dados e etc. Uma PaaS fornece ambientes de desenvolvimento de software e facilita a implantação de aplicações sem os custos e complexidades relativas a compra e gerenciamento do hardware e de software adjacentes que são necessários ao ambiente de desenvolvimento.

Muitos serviços podem ser oferecidos através de uma PaaS, facilidades para o projeto e desenvolvimento de aplicações, testes, implantação, hospedagem, integração de serviços web, segurança, integração de banco de dados, persistência, etc. Todos estes serviços também podem ser configurados como uma solução integrada, oferecida através da internet. Como exemplo de PaaS pode-se citar a Google AppEngine, Aneka [Vecchiola et al., 2009] e o Microsoft Azure.

2.4.3 Software como Serviço (SaaS)

O fornecimento de software como serviço é um conceito atrativo para os profissionais de TI que se defrontam com intermináveis atualizações, correções e disputas de licenças de software. Correspondendo a camada mais externa do modelo conceitual, ela é composta por aplicativos que são executados no ambiente

da nuvem. Podem ser aplicações completas ou conjuntos de aplicações, cujo uso é regulado por modelos de negócios que permitem customização. Os sistemas de software devem estar disponíveis na internet através de uma interface com um navegador web, logo devem ser acessíveis de qualquer lugar a partir dos diversos dispositivos dos usuários. Desta forma, novos recursos podem ser adicionados aos sistemas de forma transparente aos usuários, possibilitando assim, que as tarefas de manutenção e evolução dos sistemas sejam mais simples. A aquisição de licenças para uso é dispensada para a utilização do SaaS, reduzindo-se então custos operacionais.

Exemplos de SaaS são o Google Docs, Facebook e Microsoft SharePoint.

2.4.4 Outros Serviços

Existem muitos conceitos derivados no contexto da prestação de serviços. Estes conceitos são normalmente utilizados para diferenciar um determinado tipo de serviço, dentro os quais podemos citar, banco de dados como serviço (DaaS), testes como serviço (TaaS), segurança, simulação, comunicação, dentre outros, todos sendo oferecidos como serviço.

2.5. CONCLUSÃO

A computação como um serviço é um fato em ebulição e tem se tornando cada vez mais reconhecido, por isto, diversas empresas que já perceberam suas vantagens têm apoiado e incentivado o seu desenvolvimento. O meio científico também tem demonstrado muito interesse no desenvolvimento e adoção deste paradigma.

Este capítulo apresentou a computação em nuvens, sendo discutidos aspectos conceituais da estrutura deste paradigma. No próximo capítulo serão descritos os conceitos e especificações relacionadas com acordos em nível de serviço, bem como seus requisitos comuns e sua arquitetura.

3. ACORDOS EM NÍVEL DE SERVIÇO

A computação em nuvem, de forma um tanto quanto generalizada, utiliza o conceito de acordos em nível de serviço ou *Service Level Agreements* (SLA) para controlar o uso de recursos computacionais advindos de um provedor e definir as responsabilidades tanto para os clientes quanto para provedores. O objetivo deste capítulo é descrever as características deste tipo de acordo de negócio, relatando alguns aspectos relevantes e que servirão de fundamento para elaboração da proposta deste trabalho.

3.1 INTRODUÇÃO

Historicamente, o SLA foi desenvolvido como uma generalização e aperfeiçoamento dos serviços web padrão XML: WS-Agreement (*Web Services Agreement*) [Andrieux et al., 2007], WSLA (*Web Services Level Agreement*) [Keller et al., 2003] e WSDL (*Web Services Description Language*), [Christensen et al., 2001]. Em vez de serviços web, os modelos de SLA negociam serviços em geral, e em vez de utilizar especificamente XML, propõe uma independência de linguagem.

As propostas de SLA podem apresentar muitas diferenças, porém, segundo [Schnjakin, 2010], é possível identificar uma estrutura geral para um SLA, que apresenta informações sobre as partes envolvidas, os parâmetros do SLA, as métricas para calcular os parâmetros do SLA, o algoritmo para calcular os parâmetros do SLA, objetivo de nível de serviço ou *Service Level Objective* (SLO) e ações a serem tomadas em caso de violação do acordo.

Normalmente as estratégias para gerenciamento de SLA estão baseadas no entendimento de duas fases distintas, onde uma se relaciona com a negociação do contrato e a outra está focada na monitoração em tempo real de sua execução. Desta forma, pode-se dizer que um gerenciamento de SLA deve incluir a definição do contrato (criado a partir de um esquema básico com parâmetros de QoS), a negociação, o acompanhamento e ainda a execução do SLA, isto tudo de acordo com uma política definida entre as partes. Neste contexto, o objetivo principal é

construir uma nova camada acima da nuvem computacional que seja capaz de criar um mecanismo de negociação entre provedores e consumidores de serviços.

Pode-se observar na literatura a existência de muitos modelos para qualidade de serviços em nuvens, tais como [Fito et al., 2010] e [Malkowski et al., 2010], entretanto, estes modelos são muito gerais e não contemplam alguns aspectos específicos que agregariam maior flexibilidade e usabilidade quando forem adotados.

A tarefa de definir um SLA não é trivial, principalmente considerando que estes acordos devem refletir de forma objetiva e inequívoca aspectos relacionados a valores monetários, bem como as exigências dos usuários para seus serviços. Agregado a isto, os SLAs também precisam descrever as condições de negócio, tais como parâmetros de desempenho e disponibilidade, métricas de avaliação, contabilidade e questões jurídicas e ainda os prazos do contrato [Malkowski et al., 2010].

Segundo [Entrialgo, 2011], questões de qualidade de serviço podem ser abordadas sob vários pontos de vista, tais como, prestar um serviço sujeito a restrições de desempenho ou como descobrir e selecionar dinamicamente um serviço com requisitos de desempenho, portanto medições, monitoramento e relatórios de desempenho da nuvem são baseados na capacidade dos usuários consumirem os recursos disponíveis. Sendo que é importante que os provedores também ofereçam SLAs baseados em desempenho para os usuários [Schad, 2010].

A execução e o acompanhamento de um SLA também possuem uma função de supervisão. A intenção é verificar se as condições definidas no contrato para todos os serviços em execução estão sendo cumpridas. Podem ser utilizados mecanismos de notificação, avisando que uma situação anormal está acontecendo e ainda ativar ações corretivas, de acordo com as políticas definidas.

Dado a natureza complexa do ambiente de nuvens, outro aspecto que representa um grande desafio é a determinação da causa da interrupção de um serviço, importante para preservação da QoS. Além disto, a negociação automática de atributos de QoS ainda não está devidamente equacionada no contexto comercial. Vários esquemas de SLA e protocolos de negociação já foram propostos,

dentre eles, o WS-Agreement, [Andrieux, 2007], e o WSLA [Keller, 2003]. Porém, estes mecanismos, de modo geral, tratam apenas a negociação de um serviço simples com a básica troca de mensagens. A negociação automática de múltiplos serviços e em várias etapas ainda carece de amadurecimento.

A figura 3.1 apresenta uma visão geral de como um processo sistemático de gestão do SLA pode parecer. Pode-se perceber que o processo global de gestão SLA pode incluir diferentes *stakeholders*, dentre eles, clientes, prestadores de serviços e provedores de infraestrutura, além disto, podem ser identificado diversas atividades de negócios, tais como a avaliação do negócio, a negociação, a contratação e etc.

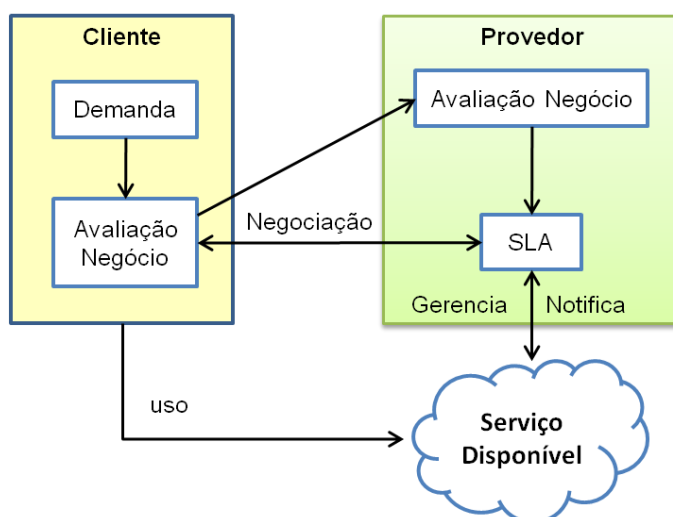


Figura 3.1: Visão Geral do SLA

Esta visão geral foi intencionalmente simplificada, pretendendo demonstrar um ciclo impulsionado pelo aumento da demanda nas atividades de um cliente. Dada esta constatação, se faz necessário aumentar a oferta dos serviços prestados sendo portanto necessário avaliar mecanismos para suprir esta demanda (Avaliação do Negócio - Cliente).

O provedor também precisa conscientemente avaliar se ele tem condições de atender a solicitação do cliente baseado em sua infraestrutura (Avaliação do Negócio - Provedor). Realizadas estas avaliações iniciais e restritivas, é então iniciado o processo de negociação dos termos do serviço com a intenção de estabelecer um SLA, para então o serviço ser disponibilizado.

3.2 CONCEITOS INICIAIS

A orientação a serviços tem se tornando um princípio básico no qual arquiteturas e estruturas de negócios tem sido organizadas, incluindo a computação em nuvem, porém para que as promessas deste paradigma gerem benefícios reais é preciso que os serviços sejam acompanhados de uma especificação exata de como devem ser utilizados. Estas condições podem ser especificadas através de um SLA, que permite um gerenciamento consistente entre as distintas camadas de uma nuvem. Um modelo de SLA deve fornecer uma especificação do conteúdo do SLA com um alto nível de detalhes, que é tanto ricamente expressivo quanto intrinsecamente extensível, suportando uma personalização controlada para domínios arbitrários que possuem requisitos específicos.

A transformação de uma economia orientada a produtos em uma economia orientada a serviços elevou grandemente a relevância das tecnologias de informação para suporte a serviços de provisionamento em diversos domínios. Entretanto, inicialmente estes serviços possuíam uma natureza estática onde os serviços têm uma configuração pré-definida e um contexto específico.

Para alavancar o contexto da orientação a serviços é preciso atender a exigência dos usuários, no sentido de que os serviços deveriam ser providos dinamicamente, com características específicas dependendo das necessidades de cada usuário, ou seja, sob demanda e apoiado por um processo de negócio transparente.

A computação em nuvem vem ganhando significativa atenção em vários cenários de negócios e isto destaca os grandes desafios no provisionamento de tecnologias dirigidas a serviço, sendo que, como mais empresas incorporam serviços de TI baseados em nuvem como parte de sua cadeia de valor, a confiabilidade e segurança tornam-se um fator crucial na gestão de negócios [Butler et al., 2011].

Além disto, a internet, sob a metáfora de “internet das coisas”, tem como objetivo oferecer acesso integrado às pessoas, aos meios de comunicação, serviços e coisas, fornecidos por uma plataforma base, permitindo novas formas de

interações sociais e econômicas em uma escala sem precedentes, oferecendo flexibilidade e qualidade além de prover características como independência de localidade para o acesso, interoperabilidade, escalabilidade, segurança e eficiência de acesso a um conjunto coordenado de serviços [Butler et al., 2011].

Entretanto, uma visão tão ampla exige uma sólida e bem definida abordagem para o seu gerenciamento. Esta abordagem deve ser compatível com ambientes com diversas e camadas, levando em consideração tanto questões que devem ser tratadas localmente quanto questões que necessitem ser geridas de uma forma mais integrada.

Considerando a orientação a serviços, computação em nuvem e internet das coisas, é possível perceber que o sucesso e maturidade acontecerão se e somente se o provisionamento de serviços for feito sob condições claramente definidas. Tais condições devem ser entendidas pelos fornecedores e usuários dos serviços. Devem ser negociáveis, para que os serviços possam atender aos requisitos dos clientes, e também devem ser gerenciáveis do ponto de vista do provedor de serviços.

Existem grandes riscos e desafios a serem resolvidos. Do ponto de vista do usuário dos serviços, o consumo destes pode impor riscos elevados. Geralmente, não existem SLAs formalmente acordados que especifiquem a qualidade e as condições de entrega de um serviço. Além disso, não existem formas padronizadas para os clientes expressarem e negociarem requisitos não funcionais. Na perspectiva do provedor de serviços, criar ofertas personalizadas de serviços, negociar individualmente com os clientes, e converter requisitos de negócios em provisionamento interno, consome tempo e recursos valiosos.

Uma solução para este problema pode ser alcançada abordando-se dois aspectos: Primeiro, a necessidade de expressar exata e claramente as condições em que os serviços devem ser entregues, podendo ser utilizado SLAs para este propósito. Segundo, é preciso que haja uma gestão sistemática dos SLAs, para que tanto os interesses dos clientes quanto dos provedor sejam negociados de forma transparente. Isso leva à necessidade de uma abordagem holística de gestão de SLA que pode ser facilmente utilizado em diferentes cenários e domínios.

Em um cenário de negócios a motivação para um prestador de serviços deve ser a capacidade de oferecer serviços com SLAs diferenciados, seguros e configuráveis, além de possibilitar uma forma automatizada para uma negociação concreta com os clientes. Além disto, as características de SLA devem abranger vários domínios não funcionais, incluindo segurança, disponibilidade, desempenho e confiabilidade.

Os sistemas de negócios normalmente consistem em complexos sistemas em camadas, desta forma SLAs em nível de usuário não podem ser diretamente mapeados para a infraestrutura física. Um serviço pode ser composto de outros serviços mais fundamentais, que por sua vez podem também serem fornecidos por terceiros. A consequência é que um mapeamento alto nível dos requisitos de SLA para níveis mais baixos e a agregação de capacidades/recursos de baixo nível para níveis mais altos, é crucial para o embasamento de um SLA em nível de usuário para uma infraestrutura. Este fluxo de informação vertical deve refletir cuidadosamente interdependências dos serviços, bem como o contexto do negócio original. Além disto, o fluxo de informações vertical também abrange acompanhamento, controle e contabilização de dados, e deve apoiar os processos de intermediação e negociação em cada camada [Future Internet, 2012].

Considerando este aspecto, normalmente acontece que os termos de garantia do SLA, não são explicitamente ou diretamente relacionados com as métricas de desempenho real ou parâmetros de configuração. Este fato dificulta a obtenção de parâmetros de configuração adequados a partir de SLAs de alto nível bem como a avaliação (baixo nível) de métricas de monitoramento. A ligação perdida entre SLAs de alto nível e as métricas e parâmetros de baixo nível é um grande obstáculo para o gerenciamento em termos dos processos de planejamento, previsão ou reconfiguração de acordo com SLAs.

3.3 REQUISITOS SLA

Na figura 3.2, pode-se observar um diagrama contendo alguns requisitos importantes para o cenário de SLA, bem como as suas relações. Neste diagrama são especificadas três fases distintas, a primeira, Fase de Definição do Contrato,

pode ser dividida na especificação do modelo, sua publicação e descobrimento. A Fase de Negociação do SLA envolve duas etapas, a negociação propriamente dita e a otimização do SLA. Por fim, a Fase de Execução do SLA é representada por quatro atividades, o acompanhamento, a avaliação, a renegociação e a contabilidade do SLA.

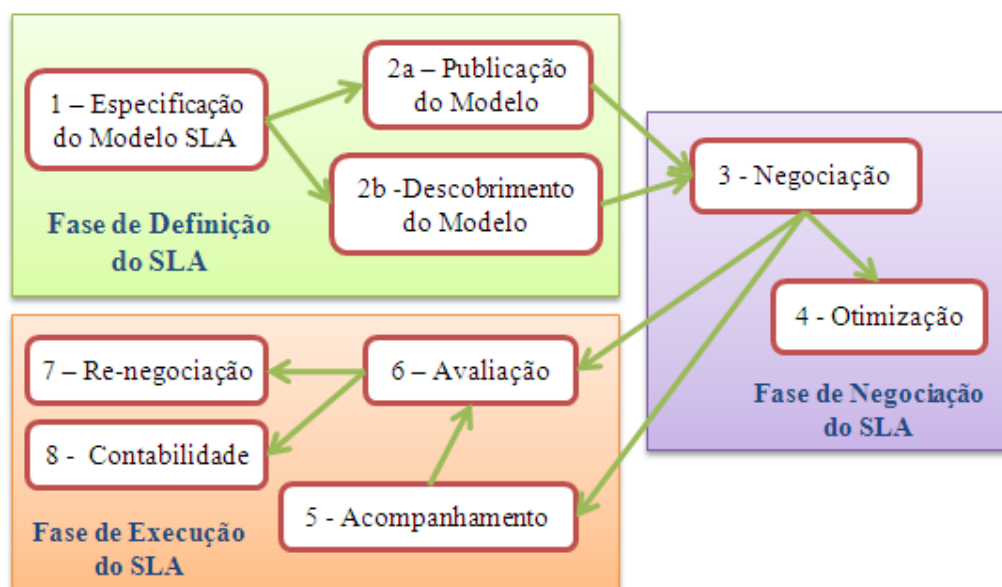


Figura 3.2: Requisitos de um SLA

As subseções a seguir, descrevem cada etapa das três fases dos requisitos de um SLA.

3.3.1 Especificação do Modelo SLA

Quando um novo recurso é disponibilizado por um provedor de serviço, ele deve incluir também um modelo de SLA. Este modelo deve descrever um tipo de contrato compatível com o uso dos recursos para regular sua utilização por todos os envolvidos no ambiente de negócio, incluindo, inclusive, penalidades pelo não cumprimento das regras. Pode ser difícil definir um modelo, porém este desafio pode ser facilitado com a utilização de adaptações realizadas em modelos utilizados previamente.

3.3.2 Publicação e Descoberta do Modelo SLA

Uma vez que um recurso foi disponibilizado, o ideal é que a maior quantidade possível de usuários tenha conhecimento deste fato. Este anúncio deve apresentar a

capacidade dos recursos oferecidos e deve ser encaminhado para um grande público. Um ambiente de publicação / descobrimento deve ser provido de um agente atuando como ponte entre os usuários potenciais e o provedor da oferta, a fim de permitir que possíveis parceiros estabeleçam comunicação.

3.3.3 Negociação

O cliente e o provedor de serviço, que são as partes envolvidas no processo, devem aceitar os termos do SLA que os vincula. Eles também devem detalhar as responsabilidades de cada parte, bem como as consequências advindas da violação das normas. Esta negociação pode consumir grande quantidade de tempo, representando um longo processo. A disponibilização de uma interface onde seja possível debater pontos conflitantes do SLA pode gerar um rápido acordo, beneficiando mais rapidamente a todos.

3.3.4 Otimização

Quando um provedor aceita uma requisição, ele precisa utilizar uma política de agendamento para definir a ordem que vai atender cada solicitação. Além disto, o provedor também precisa considerar como otimizar a utilização dos recursos e como preservar os parâmetros de QoS que são garantidos pelo SLA. Neste cenário, mesmo já estando executando outras tarefas, é muito importante considerar a possibilidade da chegada de novas requisições e suas respectivas prioridades, para atendê-las com os recursos que satisfaçam os requisitos da forma mais adequada.

3.3.5 Acompanhamento

Considerando o fato que um provedor iniciou a provisão de acesso aos recursos, ele deve monitorar a operacionalização e utilização destes recursos. As informações monitoradas podem ser utilizadas para verificar se os atributos de QoS definidos no SLA estão sendo respeitados e ainda otimizar a capacidade da infraestrutura do provedor.

3.3.6 Avaliação

Os gerentes e usuários, não possuem um interesse focado exclusivamente em saber apenas se uma tarefa está sendo executada corretamente. Outras informações como violação de contrato ou estatísticas globais também são

relevantes para a verificação do SLA. Uma avaliação consiste em um processo que analisa informações pré-determinadas sendo que estas foram monitoradas e registradas em momentos anteriores. A identificação de informações que são relevantes é importante para retratar com fidelidade o processo.

3.3.7 Renegociação

A qualquer momento, uma das partes do contrato pode querer alterar as políticas de uso dos recursos, normalmente para estar em conformidade com alguma exigência externa, advinda de mudanças no contexto operacional. Uma vez que a premissa de alterações sempre vai existir enquanto o sistema estiver executando, é importante considerar este aspecto, mas apesar do ambiente sofrer alterações o comportamento dos processos deve permanecer inalterado. Ou seja, é preciso garantir que a após qualquer migração, adição ou remoção de recursos o sistema continuará funcionando corretamente.

3.3.8 Contabilidade

A utilização dos recursos obviamente deve gerar uma lista descrevendo quais foram utilizados, em que medida e por quanto tempo, relacionando-os com os valores acertados pelo uso de cada um deles de acordo com as definições estabelecidas no SLA. Isto não representa o faturamento propriamente dito, mas é a base para elaboração do prospecto financeiro, que inclusive pode ser desfavorável ao provedor em caso do não cumprimento dos requisitos de QoS.

3.4 FUNCIONALIDADES DE UM SLA

Uma funcionalidade em um SLA está relacionada com a capacidade de atender a um ou vários dos requisitos descritos na seção 3.3.

3.4.1 Otimização da Seleção de Recursos

Este aspecto aborda a necessidade de um algoritmo de otimização integrado a um alocador de recursos, com a intenção de encontrar o tempo e o local para alocação mais adequado, considerando as definições do SLA. Esta funcionalidade

tem o propósito de aperfeiçoar a execução de uma instância de uma aplicação, tarefa ou serviço baseado nos parâmetros SLA de forma a maximizar a probabilidade de satisfação do SLA.

As requisições submetidas a um SLA devem ser processadas para selecionar o melhor host, entre todos os disponíveis. A definição do melhor host depende do estado de uma série de variáveis do ambiente, tais como, os recursos disponíveis, os recursos que são necessários para satisfazer os requisitos do SLA e ainda os objetivos de otimização definidos pelo administrador.

Alguns objetivos estão diretamente relacionados ao conhecimento dos requisitos em um SLA, como a minimização do tempo de conclusão de uma tarefa, a minimização dos custos, a maximização da probabilidade de sucesso, dentre outros. Além destes, outros objetivos estão relacionados com o estado do sistema, como por exemplo, o balanceamento da carga de trabalho entre os servidores.

3.4.2 Monitoramento em Tempo Real

Esta funcionalidade consiste em um processo de monitoramento da execução de uma instância de um serviço relacionando-a com as definições descritos no SLA de forma a verificar se o contrato está sendo respeitado ou não. Durante a execução do monitoramento, se alguns parâmetros associados aos SLOs do SLA, que são efetivamente os tópicos a serem medidos dentro do SLA, atingir um valor limite, identifica-se uma ameaça de violação e ações de recuperação podem ser ativadas a fim de preservar o SLA ou até mesmo minimizar as consequências de uma violação efetiva do contrato.

Entre as ações de recuperação, pode-se vislumbrar a realocação das tarefas de uma aplicação para os recursos temporariamente disponíveis ou ainda a aquisição de recursos adicionais. Uma opção interessante é a notificação do provedor através de alertas de ameaça, capacitando-lhe a tomar medidas para tentar impedir a violação. Outra opção para esta funcionalidade seria a possibilidade de oferecer informações para o usuário que assinou o SLA, informando-o sobre o status atual do SLA, tornando o processo mais transparente, isto porque quaisquer ferramentas de monitoramento estarão junto com os recursos, que estão em posse

do provedor.

Uma alternativa para dirimir qualquer suspeita quanto a veracidade na avaliação dos serviços, seria a utilização de um módulo de terceiros para monitorar o processo, porém isto pode ir contra as práticas de negócio do provedor.

3.4.3 Negociação do Contrato

O cliente e o provedor precisam concordar com as condições do SLA antes de assiná-lo, sendo que este ponto de comum acordo será baseado nas solicitações do usuário e nas políticas de gerenciamento do provedor. Esta funcionalidade está relacionada ao procedimento entre as partes, consumidores e provedores, concordando com os termos de um SLA que regerá o intercâmbio dos recursos. As partes tentam chegar a um acordo baseado em um consenso depois de compartilhar suas ideias e objetivos.

Um aspecto comum em acordos comerciais é que o provedor normalmente não publica ofertas, ele espera o consumidor enviar uma proposta. Para tanto o cliente precisa conhecer o que o provedor está pronto a oferecer, o que normalmente acontece através de propagandas genéricas feitas pelo provedor. Desta forma, uma proposta básica de serviços de um provedor consistiria na publicação de um modelo que descreve o serviço e os seus termos, incluindo parâmetros de QoS e as possíveis compensações em caso de violação. Estes modelos podem dispor de campos editáveis, sendo destinadas a relacionar as necessidades específicas do usuário. Em posse do modelo, o cliente preenche-o com valores que descrevem o seu planejamento para o uso dos recursos disponíveis. Novos termos podem ser adicionados, outros podem ser removidos ou alterados, gerando um novo documento que deve ser encaminhado ao provedor. Nesta ocasião, o provedor, considerando sua disponibilidade de recursos e políticas internas, envia ao cliente uma cotação com valores considerados, por ele, pertinentes para a provisão dos serviços conforme as especificidades relacionadas no documento enviado pelo cliente.

O cliente se estiver satisfeito com a cotação, assina o documento e envia para o provedor como uma proposta. Esta proposta representa um acordo que o usuário

está pronto para cumprir. O provedor é livre para rejeitar ou aceitar, no último caso, a proposta torna-se um SLA oficialmente assinado por ambas as partes, e começa a ser um documento legalmente válido. Caso o cliente não fique satisfeito com a cotação, o processo pode ser repetido, até chegar-se a um acordo ou então desconsiderar a contratação do serviço com aquele provedor. Durante a negociação as partes podem modificar livremente os termos em relação a taxas, parâmetros de QoS, disponibilidade de recurso, dentre outros.

Uma vez que o contrato foi concordado e assinado, ainda assim a necessidade de mudança precisa ser considerada, sendo necessária uma renegociação. Neste caso, o mesmo procedimento pode ser utilizado, embora se tenha que considerar a existência do primeiro contrato, que já existem recursos alocados e ainda que dependendo da alteração podem surgir dificuldades técnicas no lado do provedor.

3.4.4 Publicação e Descoberta de Serviços

Um espaço virtual compartilhado ou um mercado de serviços é necessário para que provedores e clientes possam se comunicar. Os provedores devem anunciar os seus serviços, utilizando de um mecanismo de publicação baseado em seus modelos de SLA, enquanto os clientes precisam encontrar um serviço que possa satisfazer as suas necessidades, possivelmente utilizando um mecanismo de busca.

Disponibilizar serviços para terceiros, obviamente envolve a desafiante tarefa de comunicar a sua existência, fornecendo meios para encontrá-los. Portanto, técnicas para a publicação de serviços, ou seja, anúncios para uma comunidade, precisam ser estabelecidas. Descobrir serviços representa um processo para localização de provedores que disponibilizam documentos com a descrição dos serviços que tenham sido disponibilizados. Para implementar uma funcionalidade relacionada com a publicação e descoberta de serviços, minimamente são necessários alguns componentes infraestruturais, tais como:

- a) um mecanismo de registro no ambiente virtual de serviços, onde um provedor de serviços publica o seu serviço juntamente com os preços de cada

um deles.

b) a criação e registro de corretores de recursos, que são responsáveis por descobrir os recursos e serviços com características específicas que atendem os consumidores com seus requisitos de QoS, também se faz necessário.

c) os consumidores de serviços também precisam de um mecanismo de registro neste ambiente virtual. Neste caso, os próprios clientes querer pesquisar em busca de serviços que atendam suas necessidades ou atribuir essa tarefa a um corretor de recursos que monitora o espaço.

Essas tarefas envolvem vários aspectos, todos alicerçados na existência do mercado de serviços, que pode reunir provedores e consumidores em potencial. Este mercado precisa estabelecer procedimentos para a publicidade dos serviços, como a definição de formatos comuns para modelos de SLA e descrição dos serviços, mas também deve prover ferramentas de mineração de dados para facilitar a pesquisa.

3.4.5 Criação de Modelos SLA

Esta funcionalidade está relacionada a criação e definição de modelos SLA que serão publicados por um provedor. Estes modelos são componentes muito importantes para negociação do SLA e para a descoberta dos serviços do provedor, tanto que os modelos são as ferramentas que dão suporte a publicidade dos serviços para o mundo exterior e a negociação sempre começa com um cliente solicitando um modelo de SLA para um provedor.

Em alguns casos os provedores de serviço podem ter dificuldade para descrever seus recursos de forma clara e segura, incluindo aspectos jurídicos e de usabilidade. De modo geral, um modelo SLA deve ser facilmente interpretado sem a necessidade do auxílio de outros mecanismos, porém sem perder sua expressividade, sendo que isto permitirá a qualquer cliente entender exatamente o que está especificado no modelo, além de facilitar a criação dos modelos pelo provedor.

Embora possa ser tentador adaptar os modelos de SLA existentes para atender às necessidades do cliente, é importante que sejam consideradas as limitações do provedor de serviços como as mais importantes. Para o caso de conflito, as restrições do prestador de serviços devem ter preferência sobre as necessidades do cliente de serviços.

A criação de modelos SLA não é uma tarefa trivial e provedores de serviço sem experiência precisam de assistência para especificar os modelos SLA para os serviços que prestam, por isso não é suficiente fornecer somente a definição do que o modelo deve conter. Parece razoável fornecer componentes para aliviar o fardo de criação de um modelo SLA, sendo que isto pode ser realizado fornecendo-se uma interface simples para a criação de um framework para modelos SLA que podem então ser refinados pelo provedor. O uso de modelos SLA deve sempre estar em sintonia com os aspectos legais, o que é crítico para ser utilizado no domínio de negócios.

3.4.6 Contabilidade do SLA

Os provedores precisam fazer um resumo dos recursos usados em uma base de dados do usuário para compará-lo com as condições acordadas no SLA. Os dados monitorados devem ser utilizados e analisados, sendo a contabilidade feita regularmente, incluindo as possíveis sanções para as violações do SLA. Esta funcionalidade abrange o processo de cobrança dos consumidores pela utilização dos serviços do provedor de acordo com os preços acordados entre as duas partes que estão definidos no SLA. Para que a cobrança dos clientes seja feita através do SLA, o modelo de SLA deve ser concebido com este propósito. De forma mais específica, algumas métricas do SLA devem representar os termos dos valores, descrevendo como o consumidor vai ser cobrado para o uso de um recurso específico.

As métricas que representam condições de preço devem conter informações como o valor pelo uso da métrica, a moeda e o período pelo qual o preço é válido. O projeto do modelo de SLA deve ser suficientemente flexível para permitir diferentes tipos de métricas e encargos. Custos adicionais poderão ser cobrados, dado o uso

cumulativo de uma métrica x ou para o aumento da disponibilidade de uma métrica y ao longo do período de faturamento. O componente de contabilidade deve recolher periodicamente informações sobre o uso, que é fornecido pelo componente de acompanhamento, para então calcular as taxas que devem ser cobradas do cliente. A qualquer momento o cliente deve ser capaz de visualizar o uso de recursos que ele fez, juntamente com seus respectivos encargos.

Também é possível definir sanções que serão impostas ao consumidor, quando um de seus aplicativos causarem uma violação de SLA. Nesse caso, a taxa de penalização relacionada com a violação de uma restrição também deve ser incluída no modelo SLA como um dos seus termos. Sendo que o componente de acompanhamento deve enviar notificações para o módulo de contabilidade cada vez que uma violação ocorre, de modo que este possa calcular a taxa, com base no SLA, para então notificar o consumidor.

3.6 NEGOCIAÇÃO DO SLA

Em sistemas baseados em serviços, a negociação de acordos de nível de serviço ocupa um papel muito importante no ciclo de provisionamento do serviço. Durante as negociações as partes devem utilizar um mecanismo interativo que possibilite a troca de ofertas, portanto, a aquisição de software como serviço que tem sua configuração negociada, está ganhando popularidade devido várias razões tecnológicas e de negócios [Dubey et al., 2007], [Elfatry et al., 2004]. Estes modelos de negócios têm um potencial inerente para reduzir o Custo Total de Propriedade (TCO) e melhorar o retorno sobre o investimento (ROI).

Tradicionalmente, o software é comprado sob uma licença e utilizado em conformidade com a mesma. Considerando os modelos “*aaS*”, o usuário assina um software, sendo isto determinado por um processo de negociação que deverá convergir para um acordo entre o cliente e o prestador do serviço. Durante a negociação, o produto será adaptado / configurado às necessidades do consumidor e a capacidade do provedor. Assim, o paradigma “*as a service*” injeta flexibilidade para o contexto de utilização de software. Em um mercado aberto, como o previsto pela computação orientada a serviços, esta flexibilidade se torna uma necessidade,

sendo que aspectos de uso como custo e qualidade não podem ser fixados previamente, estando dependentes da situação atual de oferta e demanda no mercado [Bennett et al., 2000].

Um protocolo de negociação determina a cardinalidade das partes envolvidas, seus papéis, a visibilidade das ofertas trocadas, o gerenciamento de sessão, os limites para negociação, e assim por diante. Normalmente, uma máquina de software dedicado é necessária para executar os protocolos / interfaces de negociação. Para entrar no mercado, os fornecedores frequentemente anunciam seus serviços usando a publicação de *templates* que expressam as propriedades funcionais e não funcionais de um serviço, juntamente com as restrições necessárias para adequá-lo como uma oferta concreta. Um único serviço pode ser anunciados por meio de vários *templates*.

Negociações baseadas em *templates* foram implementadas pelo *framework Web Service Agreement (WSAG)* [Andrieux et al., 2007], [Waeldrich et al., 2007] e pelo *framework Web Services Level Agreement (WSLA)* da IBM [Ludwig et al., 2003]. Usando estes modelos, as ofertas e contraofertas são trocadas entre as partes negociadoras em uma sequência determinada pelo protocolo de negociação. Na melhor das hipóteses, um acordo será realizado e documentado como um SLA.

O cliente deve agir em conformidade com os níveis de utilização acordados e o provedor deve manter os níveis de qualidade de serviço também acordados. Em caso de violação, as penalidades devem ser aplicadas. Um SLA pode precisar ser renegociado caso os clientes experimentem uma mudança na demanda do serviço ou o provedor precise reajustar seus recursos.

3.6.1 Estado da arte em negociação

A negociação tem se tornado um grande desafio de pesquisa. Vários conceitos de economia, inteligência artificial e teoria de jogos foram combinadas para tratar as preocupações relacionadas com negociação através de abordagens interdisciplinares.

Uma das primeiras análises formais do processo de negociação foi realizada

por John Nash [Nash et al., 1950] em seu trabalho sobre negociação um-para-um e depois em jogos não cooperativos. Isto popularizou a teoria dos jogos, e posteriormente levou a sua absorção pela Ciência da Computação, especialmente entre os agentes inteligentes distribuídos [Weiss, 2000].

Vários *frameworks* e sistemas de negociação de suporte ou *Negotiation Support Systems* (NSS) têm sido propostos na literatura. O *framework* OPELIX [Hauswirth et al., 2001] é um projeto europeu que permite a um cliente e um provedor terem negociações bilaterais, totalmente automatizadas. A arquitetura OPELIX implementa todas as fases fundamentais de uma transação comercial: ofertas de produtos e de descoberta, processo de negociação, atividades de pagamento e a entrega do produto ao cliente. No entanto, não suporta os protocolos de negociação sofisticados, sendo restrito a negociações bilaterais.

Os três *frameworks* apresentados a seguir estão relacionados com projetos desenvolvidos pela Concordia *University* em conjunto com a Carleton *University*, ambas do Canadá. O Inspire [Kersten and Noronha, 1999] oferece suporte a operadores humanos na gestão de negociações bilaterais, gerenciando as ofertas e as contrapropostas apresentadas pelos participantes. As funções que norteiam a decisão de cada participante são confidenciais. Como evolução melhorada do Inspire surgiu o Aspire [Kersten et al., 2003], fornecendo apoio a negociação por meio de agentes inteligentes que fazem sugestões aos usuários sobre quais operações executar. Os agentes não automatizam completamente o processo de negociação, apenas prestam apoio a tomada de decisão, por estarem cientes do status das sessões de negociação, implementando uma estratégia de negociação específica, definida em termos de pesos sobre as variáveis de negociação e funções objetivas.

O projeto e-Agora [Chen et al., 2004], oferece um mercado de negócio complexo, no qual os usuários interagem através de agentes inteligentes autônomos. O sistema fornece um modelo de processo e suporta um conjunto de protocolos, sendo que o processo é definido como uma série de atividades e as fases e os protocolos são definidos por meio de regras e restrições as atividades de negociação.

No trabalho Kasbah [Chavez et al., 1997], é permitido que os potenciais compradores e vendedores criem seus próprios agentes, atribuindo-lhes algumas orientações estratégicas para então enviá-los para um mercado centralizado onde ocorrem as negociações. O apoio é limitado a negociações bilaterais. A única ação válida no protocolo de negociação distributiva é para a compra de agentes para oferecer uma oferta para os vendedores. Dado este protocolo, Kasbah fornece aos compradores várias estratégias de negociação que determinam uma função de lances crescentes para um produto ao longo do tempo.

Em AuctionBot [Wurman et al., 1998] é oferecido um versátil servidor de leilão *online*. Agentes de software são providenciados para conduzir os leilões com base em parâmetros específicos: a participação (número de participantes), bens discretos (lances só são permitidos em quantidades inteiras) e regras de licitação que determinam aceitabilidade, melhoria das ofertas e condições de fechamento.

O *framework* ASAPM [Chhtriet al., 2007] descreve um sistema multi-agente que permite negociações automatizadas utilizando o *Iterated Contract Net Protocol* (ICNP) da *Foundation for Intelligent Physical Agents* (FIPA). Neste *framework*, existem agentes que negociam a partir dos termos de qualidade de serviço e o ICNP acomoda esta funcionalidade permitindo várias rodadas de negociação.

No trabalho [Karaenke and Kirn, 2010] foi desenvolvido um *framework broker-based* para a realização de negociações de SLA. É utilizado um protocolo de negociação multicamadas que é também baseado no ICNP da FIPA. O escopo do protocolo é alargado para permitir interações de negociação entre cadeias de serviços diferentes.

O CAAT [Ncho and Aimeur, 2004] é um *framework* que pode ser usado para projetar sistemas multi-agente para negociações bilaterais e trilaterais automáticas. Este protocolo de negociação permite sequências válidas de interações usando mensagens construídas sobre o *Agent Communication Language* (ACL) da FIPA. Uma ontologia que define a semântica de comunicação foi desenvolvido e usado nas mensagens para transmitir uma determinada ação.

As abordagens apresentadas acima fazem avanços interessantes em relação a negociação automatizada, mas eles ainda não são flexíveis o suficiente para

projetar comportamentos interativos e personalizados ou para personalizar facilmente negociações em domínios de aplicações individuais.

Para este fim, SECSE [DiNitto et al., 2007] foi desenvolvido para fornecer uma infraestrutura flexível que pode ser adaptado em termos de multiplicidade, *workflow*, protocolo e modelo de decisão para atender a uma aplicação de domínio específico. A arquitetura do *framework* de negociação é composta de um mercado de negócio que abriga múltiplos agentes, onde cada agente está associado a um participante específico da negociação e a um componente negociador. Os negociadores interagem com os participantes humanos através de interfaces gráficas que permitem fazer ofertas e contraofertas. Além disso, um modelo de decisão interno ou modelo de decisão definido pelo usuário pode ser encapsulado ao *framework* para executar negociações automáticas. SECSE apoia negociações híbridas, onde alguns participantes são agentes automatizados, enquanto outros são seres humanos.

O WSAG [Andrieux et al., 2007] representa um esforço de padronização do *Open Grid Forum* (OGF) e oferece uma especificação para acordos baseados em serviços web. Uma linguagem foi desenvolvida e pode ser usada para especificar um modelo de acordo e ainda operações padrão para a gestão do ciclo de vida do serviço. Além disso, ele fornece um protocolo de negociação que permite o estilo *take-it-or-leave-it* de negociações bilaterais.

Por fim, analisando as abordagens de arquitetura e projetos propostos na literatura, diferentes padrões foram observados: 1) arquiteturas *broker-based*, onde um componente *broker* gerencia negociações Um-para-Um em nome das partes envolvidas; 2) arquiteturas *marketplace-based*, onde as partes envolvidas em negociações M-para-N são geridos por um mercado intermediário, exigindo que os participantes da negociação exponham suas preferências para o *framework* de negociação; 3) agentes independentes que negociam uns com os outros sem mediação.

Estes padrões competem ou cooperam livremente com base em racionalidade individual. Do ponto de vista dos protocolos descritos, observou-se abordagens baseadas em regras de negócio que regulam o processo de

negociação. O uso de ontologias e esquemas representam o conteúdo e semântica das mensagens e os protocolos de negociação são baseadas em parâmetros de configurações.

3.6 CONCLUSÃO

Neste capítulo foram discutidos conceitos sobre acordos em nível de serviço, ou como é comumente conhecido, SLA. Foram apresentados os requisitos para um SLA, especificados algumas funcionalidades e uma arquitetura e por fim descritos tópicos usuais que são comumente encontrados em um SLA.

A utilização de SLA tem como propósitos principais auxiliar no gerenciamento dos recursos disponíveis em uma nuvem computacional e garantir a satisfação dos requisitos contratados por cada cliente.

No capítulo seguinte serão descritos uma série de trabalhos que estão relacionados com o contexto desta proposta, sendo identificadas suas principais atividades e como a abordagem proposta atende cada um destes aspectos.

4. TRABALHOS RELACIONADOS

A computação em nuvem é o contexto de desenvolvimento deste trabalho, sendo considerado que a estrutura da nuvem é mantida com um grande e complexo conjunto de recursos que suportam os serviços sob demanda. Desta forma, considerando que o domínio em que o GerNU está inserido é muito amplo e complexo, e também que, dentro dos nossos mecanismos e possibilidades de verificação, não conseguimos identificar na literatura, trabalhos que sejam similares ao contexto geral da proposta do GerNU, no que diz respeito a atender os três objetivos especificados no Capítulo 1, desta forma, estaremos descrevendo trabalhos relevantes, porém baseado nos aspectos gerais que envolvem a abordagem da proposta, mas que em conjunto descrevem o objetivo do GerNU.

Os trabalhos considerados neste capítulo estão relacionados com a utilização dos contratos em nível de serviço, sendo que no Capítulo 4, destinado a detalhar este tema foi apresentado diversos trabalhos relacionados exclusivamente com o contexto de negociação. Além destes, relatamos a utilização de parâmetros de qualidade para garantir os requisitos contratados pelos usuários E por fim, a abordagem dirigida a modelos para criação de serviços. A diversidade e quantidade de trabalhos, com alto nível de qualidade, disponíveis na literatura sobre cada um desses assuntos, obviamente com especificidades que os tornam completamente distintos, servem para ressaltar a relevância destes temas como objeto de pesquisa, tendo-se ainda muitos desafios a serem resolvidos. Assim sendo, serão descritos uma série de trabalhos que, dado suas características, contribuíram diretamente ou indiretamente para o desenvolvimento das estruturas e políticas adotadas pelo GerNU e inclusive colaboraram na concepção deste projeto como um todo.

Ao final de cada seção será apresentada uma análise comparativa entre os trabalhos descritos, objetivando identificar com que tipo de atividade cada trabalho está relacionado, identificando aspectos comuns entre os mesmos. Além disto, também será brevemente descrito como as funcionalidades do GerNU tratam cada um destes aspectos.

4.1 CONTRATOS EM NÍVEL DE SERVIÇOS (SLA)

a) [Mahbub et al., 2011]

O trabalho [Mahbub et al., 2011] apresenta um processo para transformar as especificações de SLA em especificações operacionais que possam ser utilizadas por agentes de monitoramento do sistema. Este artigo aponta a necessidade de desenvolvimento de um *framework* para definição de SLAs, que permita interoperar com os mais diversos ambientes. Este *framework* deve também incorporar os requisitos em termos do ambiente, especificações e aplicações, não permitindo apenas as especificações necessárias para o monitoramento, mas também outras especificações que facilitem a utilização do ambiente pelo usuário.

b) [Yaqub et al., 2011]

Um importante aspecto é apresentado no trabalho [Yaqub et al., 2011], sendo descrito a necessidade de mecanismos que realizem o processo de negociação de forma mais interativa e automatizada, facilitando a negociação dos serviços e tendo os requisitos agregados. A automatização desse processo também envolve a definição de padrões de protocolos que não servem apenas para verificar os requisitos do contrato, mas também para definir os parâmetros de funcionamento dos recursos disponíveis. Também é apontado que a diversidade nas pesquisas devem revelar novos fatos sobre o processo de criação automática de SLA, especialmente em áreas com dependências aninhadas e que precisam de algoritmos de otimização eficientes, estratégias de negociação que possam rapidamente convergir em acordos, análise de tendências de mercado e a criação de perfis, produzindo desta forma resultados concretos que podem levar a uma bem-sucedida economia orientada a serviços. Este estudo e também [Mahbub et al., 2011] levantam questões importantes, tais como, SLAs que permitem a definição de aspectos como segurança, custo, dependabilidade e infraestrutura, adicionados a camada de negócios.

c) [Winkler and Gilani, 2011]

A proposta apresentada no artigo [Winkler and Gilani, 2011] propõe que o tratamento dos requisitos seja realizado através da utilização de uma abordagem

baseada em modelos, dado a possibilidade de se alcançar uma maior abstração do processo e as vantagens do intercâmbio destas necessidades, bem como a criação e reutilização de novos recursos e funcionalidades. Esta visão é apoiada pelo trabalho [Bernsme et al., 2011], que descreve a segurança como um aspecto crítico em ambientes de nuvem. Para atenuar esse problema, o trabalho aponta a necessidade de se estabelecer mecanismos de intercâmbio de informações e o estabelecimento de padrões, propondo que os SLAs também sejam utilizados para descrever os requisitos dos usuários, os contratos de serviços e os requisitos da camada de infraestrutura, por meio de um protocolo padrão usado por vários e diferentes ambientes e seus provedores de nuvem.

d) [Van et al., 2009]

A utilização de técnicas de virtualização em servidores para prover plataformas independentes proporciona uma série de vantagens, tais como: grande flexibilidade advinda da capacidade de consolidar várias máquinas virtuais em um mesmo servidor físico; redimensionamento da capacidade de uma máquina virtual; e migração de máquinas virtuais entre servidores físicos. Porém, [Van et al., 2009] afirma que um desafio fundamental para provedores de computação em nuvem é a automatização do gerenciamento de servidores virtuais, levando em consideração tanto os requisitos de alto nível de QoS das aplicações que estão hospedadas, quanto os custos para gestão dos recursos. A partir dessa constatação, foi proposto um gerenciador de recursos autônomo, com o objetivo de controlar o ambiente virtualizado, separando o provisionamento dos recursos do posicionamento dinâmico das máquinas virtuais. Este gestor visa otimizar uma função de utilidade global que integra tanto o grau de cumprimento do SLA quanto os custos operacionais, sendo utilizado uma abordagem de programação restrita para formular e resolver o problema de otimização.

e) [Bolor et al., 2010]

Datascenters geograficamente distribuídos formam um sistema de computação coletivo em nuvem, onde é feita a hospedagem de múltiplos aplicativos, estando cada um deles sujeito a acordos de nível de serviço. Estes SLAs, um para cada aplicação, exigem um tempo de resposta em conformidade com o valor

especificado, sendo que as requisições que não estiverem em conformidades estarão sujeitas a penalidades. Neste cenário, [Bloor et al., 2010] apresenta uma abordagem heurística que é baseada nas requisições de agendamento de cada servidor em cada um dos datacenters distribuídos geograficamente, tendo o objetivo de minimizar globalmente as penalidades imputada ao sistema de computação em nuvem.

f) [Alhamad et al., 2010a]

O trabalho [Alhamad et al., 2010a] afirma que os clientes de serviços na nuvem não precisam ter um contrato de longo prazo com os prestadores dos serviços, além disso, como a computação em nuvem provê serviços para muitas aplicações de negócio que são críticas, é de grande relevância a existência de mecanismos que sejam confiáveis e flexíveis para gerenciar os contratos on-line. Neste artigo são apresentados critérios que devem ser considerados na fase de concepção de um SLA para a computação em nuvem, além de descritas estratégias de negociação entre consumidor e fornecedor da nuvem com o objetivo de propor um método para a confiabilidade entre as partes envolvidas no processo de negociação. Em [Alhamad et al., 2010b], é apresentado uma evolução do trabalho [Alhamad et al., 2010a], sendo integrado um modelo de confiança para avaliar os serviços da nuvem, a fim de ajudar os usuários da nuvem a selecionar os recursos que sejam mais confiáveis.

g) [Theilmann et al., 2010]

O paradigma da orientação a serviços esta sendo expandido para diferentes domínios, sendo que estes serviços podem ser compostos por outros serviços que, inclusive, podem pertencer a domínios distintos. Conseqüentemente, os SLAs devem ser gerenciados através das hierarquias destes serviços. Em [Theilmann et al., 2010] é apresentado uma arquitetura que foi implementada com este objetivo, sendo também descrito como uma hierarquia de serviços pode ser gerenciada através de SLAs, além dos detalhes de como estabelecer SLAs para as diferentes camadas e como os termos do SLA devem ser traduzidos através da hierarquia dos serviço.

h) [Iqbal et al., 2010]

Em [Iqbal et al., 2010] é afirmado que os SLAs que geralmente são oferecidos pelos provedores da computação em nuvem não dão garantias sobre o tempo de resposta para aplicações web que estão hospedadas na nuvem. Satisfazer uma garantia de média máxima para tempo de resposta em aplicações web é difícil, principalmente considerando que os padrões de tráfego são imprevisíveis. Além disso, a complexa natureza de aplicações web multicamadas aumenta a dificuldade de identificar os problemas críticos e resolvê-los automaticamente. Os autores propõem uma abordagem que oferece garantias para os proprietários de aplicações neste segmento, que tem como objetivo minimizar a utilização de recursos.

i) [Emeakaroha et al., 2010]

A fim de evitar os custos dispendiosos advindos de violações de SLA e ainda tentar reagir as falhas e alterações no ambiente de nuvem, são necessárias estratégias avançadas para os SLAs, tais como conceitos adequados sobre o monitoramento dos recursos. Em [Emeakaroha et al., 2010] é descrito que frequentemente os provedores de nuvem adotam ferramentas de monitoramento que são restritas a localidade e a homogeneidade dos objetos monitorados, não são escaláveis e não suportam o mapeamento de métricas de baixo nível para os recursos do sistema. Os autores apresentam um *framework* denominado LoM2HiS para a gestão dos mapeamentos das métricas de baixo nível dos recursos para os parâmetros SLAs de alto nível. O LoM2HiS pode detectar ameaças de futuras violações do SLA, podendo notificar o componente responsável de modo a evitar as ameaças.

j) [Correia and Brito, 2010]

A gerência de serviços em tecnologia da informação ou *IT Service Management* (ITSM), segundo [Correia and Brito, 2010] é o conjunto de processos que permitem planejar, organizar, dirigir e controlar o provisionamento de serviços de TI. Entre as preocupações de ITSM, considerando a gestão em nível de serviços, estão os requisitos para disponibilidade de serviços, desempenho, precisão, capacidade e segurança. Este artigo apresenta uma visão geral de uma abordagem que considera três problemas específicos: (1) Os SLAs no contexto de ITSM são

informalmente especificados em linguagem natural; (2) As especificações de SLAs não estão fundamentadas em modelos de processos de ITSM; (3) A verificação de conformidade dos SLAs em TI não é realizada no mesmo nível de abstração como no projeto do serviço. Para suavizar esses problemas, é proposto uma abordagem baseada em modelos para especificação de SLA serviços de TI e a verificação de conformidade.

k) [Prasad et al., 2010]

Um método para a alocação de recursos para serviços de processamento de dados em nuvem, levando em conta não apenas o poder de processamento e memória, mas também a velocidade da rede, a confiabilidade e a taxa de transferência (*throughput*) de dados é apresentado no trabalho [Prasad et al., 2010]. Neste trabalho também são apresentados algoritmos para particionamento de dados, com o objetivo de possibilitar a transferência de blocos de dados paralelos, objetivando melhorar a taxa de transferência e os recursos alocados na nuvem. Além disso, são descritos métodos para otimização de preços e determinação do SLA para um determinado trabalho de processamento de dados.

l) [Moon et al., 2010]

Enquanto a computação em nuvem oferece aos clientes vantagens como a capacidade de reduzir custos, transformar as despesas de capital em despesas operacionais e de acelerar a capacidade de inovação, também cria desafios para os provedores de serviços em nuvem, como lidar com uma diversa e dinâmica carga de trabalho, oferecendo um preço competitivo. Sendo realizado todo este esforço para convencer potenciais clientes a utilizar este modelo de prestação de serviços. Além disto, a qualidade dos serviços deve ser minimamente comparável, em todos os aspectos, com os outros paradigmas de serviços. Baseado nestes fatos [Moon et al., 2010] afirma que o sucesso de serviços baseados em nuvem, depende dos dois fatores principais: qualidade dos serviços e gestão de custos.

m) [Fito et al., 2010]

No trabalho de [Fito et al., 2010] é abordado o uso da computação em provedores de hospedagem web, a fim de evitar suas limitações na gestão de

recursos, sendo apresentado um provedor de hospedagem web elástico, que faz uso de técnicas de terceirização, a fim de tirar partido das infraestruturas de computação em nuvem para fornecer escalabilidade e alta disponibilidade para as aplicações web. Além disso, é objetivado maximizar a receita auferida pelo prestador por meio de análises SLA e emprego de um modelo econômico.

n) [Shu and Meina, 2010]

Considerando a gestão de SLA, o seu ciclo de vida deve ser cuidadosamente analisado, pois o ciclo de vida de um SLA inclui questões que estão fora do ambiente de execução do serviço, assim é necessário dividir as funções de gestão baseado no ciclo de vida do SLA. Neste contexto, o trabalho [Shu and Meina, 2010], apresenta uma arquitetura para serviços web e seus requisitos de gerenciamento, sendo estes baseados no ciclo de vida do SLA. Uma plataforma de gestão do SLA foi projetada para a definição do SLA, para o registro do serviço, para o monitoramento e controle do SLA em tempo de execução. O documento ainda apresenta um modelo para o mapeamento de parâmetros de SLA que são anunciados pelos prestadores de serviços para os parâmetros de QoS que são exigidos pelos clientes dos serviços.

o) [Brandic et al., 2010]

Infraestruturas de nuvens auto-gerenciáveis são necessárias para atingir um nível de flexibilidade e para cumprir às exigências dos consumidores de serviços especificadas nos SLAs [Brandic et al., 2010]. Estas infraestruturas devem responder automaticamente a mudanças de componentes, a carga de trabalho e as condições do ambiente, minimizando as interações do usuário com o sistema e impedindo violações dos SLAs acordados. No entanto, a identificação dos responsáveis pela violação de um SLA e a decisão sobre ações reativas que são necessárias para impedir a violação está longe de ser trivial. Sendo assim, os autores, primeiramente apresentam uma abordagem para o mapeamento de métricas de baixo nível dos recursos para os parâmetros de SLA necessários para a identificação de fontes de falha. Em segundo lugar é descrito uma arquitetura de nuvens em camadas para uma propagação *bottom-up* de falhas na camada que podem reagir às ameaças de violação do SLA que forem captadas. Além disso, foi

apresentado um modelo de comunicação para a propagação dessas ameaças de violação para a camada adequada na infraestrutura da nuvem, o qual inclui negociadores, corretores e um serviço de implementação automático.

4.1.1 Comparativo de Atividades

A tabela 4.1 tem o objetivo de catalogar os artigos descritos na seção 4.1, relacionando-os através da identificação das principais atividades que cada um se propõe realizar. Destacando-se desta forma, algumas das possibilidades de utilização de SLAs, além de, de forma objetiva, criando possibilidades para a adequação da proposta, sendo adicionado na última coluna, índice X, quais aspectos são abordados pelo GerNU.

Atividades	Artigo (identificado pelo índice)																
	a	b	c	d	e	F	g	h	I	j	k	l	m	n	o	X	
Ações Corretivas Reativas									X						X	X	
Alocação de Recursos								X									
Automatização		X		X												X	
Confiabilidade						X											
Definição SLA	X					X				X				X		X	
Gerenciamento				X	X		X	X						X		X	
Hierarquia de Serviços							X										
Intercâmbio de Informação			X													X	
Mapeamento (baixo-nível / alto nível)	X								X					X	X	X	
Monitoramento									X							X	
Negociação		X				X										X	
Otimização				X	X			X			X	X	X				
Padronização			X													X	
Reutilização			X														
Desempenho											X						

Tabela 4.1: Atividades dos artigos sobre SLA

4.1.2 Abordagem do GerNU

Um dos objetivos da abordagem proposta por este trabalho é automatizar o máximo possível do processo de negociação, criando um mecanismo simples e intuitivo de forma a reduzir a intervenção humana, acelerando o processo de entrega do serviço para o usuário, sendo que a perspectiva é tornar o uso de SLA mais

efetivo.

Outro aspecto está relacionado com definir claramente as responsabilidades para os clientes e o provedor, permitindo a verificação dos acordos durante a execução do serviço, agregando eficiência e confiança ao SLA. Também é viabilizado um claro mapeamento entre as especificações operacionais e os parâmetros que devem ser monitorados, possibilitando o fornecimento de garantias aos serviços de forma a aumentar a confiabilidade. Além disto, foi utilizado um modelo para, dentre outras possibilidades, viabilizar o intercâmbio de informações entre partes.

4.2 QUALIDADE DE SERVIÇO

a) [Distefano et al., 2011]

No trabalho de [Distefano et al., 2011] é descrito que no provimento de serviços, usualmente se aplica uma política de melhor esforço, sem levar em conta acordos em nível de serviço e a qualidade do serviço, sendo assim apresenta uma proposta focada nos aspectos de QoS, descrevendo como fornecer SLA baseado em garantias para QoS através do Cloud@, que tem o objetivo de fornecer e especificar a arquitetura, os algoritmos e os componentes que implementam as funcionalidades de gestão. A proposta desta abordagem consiste em fornecer uma camada adicional a nuvem, com a responsabilidade de garantir altos níveis de disponibilidade dos recursos, sendo que o sistema é fornecido com uma funcionalidade para negociar o nível desejado de QoS, na forma de disponibilidade de recursos, para ser possível monitorar o nível de qualidade da prestação do serviço, podendo-se então realizar ações para garantir o nível de QoS contratado.

b) [Liu et al., 2011]

Considerando que os requisitos de QoS podem ter muitas dimensões, um projeto de sistema unificado para gestão de diferentes componentes de QoS é necessário para reduzir a complexidade do sistema e até o custo de desenvolvimento de *software*. A partir desta perspectiva [Liu et al., 2011] propõe um *framework* de QoS genérico para sistemas de *workflow* em nuvens. Este *framework*

abrange as principais etapas do ciclo de vida de um *workflow*, sendo constituído por quatro componentes destinados a: especificação dos requisitos, seleção consciente de serviços, monitoramento e tratamento de violação.

c) [Pedersen et al., 2011]

Garantir ou até mesmo prever QoS em redes globais e diversificadas que suportam os contextos complexos da hospedagem de serviços é uma questão desafiante que necessita uma abordagem refinada. Em [Pedersen et al., 2011] é investigado se a latência, em termos de medições de um *ping* simples, pode ser usado como um indicador para outros parâmetros de qualidade, tais como tempo de transferência (*throughput*) e variação (*jitter*).

d) [Kim et al., 2010].

Considerando que aplicações complexas podem ser descritas como processos que invocam diversos serviços em tempo de execução, um requisito não funcional, como a qualidade de serviço, pode ser visto como uma base importante para a seleção de serviços em nuvem. No entanto, um complexo ambiente de rede e a realização de operações ilegais geralmente resultam em dados de QoS não confiáveis, influenciando a precisão e a confiabilidade da nuvem. Desta forma, este trabalho propõe uma abordagem de gerenciamento para serviços de QoS em nuvem [Kim et al., 2010].

e) [Zheng et al., 2010]

Abordagens para ranquear a qualidade de componentes em uma nuvem são cruciais para a tomada de decisão sobre qual componente é ideal, dado um conjunto de componentes candidatos funcionalmente equivalentes. Além disso, um ranking de qualidade ajuda os projetistas de aplicações ou serviços a detectarem os componentes com baixo desempenho em aplicações complexas na nuvem, que geralmente incluem grande número de componentes distribuídos. A partir deste pressuposto, objetivando fornecer uma classificação personalizada de componentes para os projetistas de diferentes aplicações em nuvens, [Zheng et al., 2010] propõe um *framework* para ranqueamento de componentes baseado em QoS.

f) [Xu et al., 2009]

O trabalho de [Xu et al., 2009], no contexto de *workflow*, considera que, como uma nuvem deve prover serviços para diversos usuários ao mesmo tempo e diferentes usuários possuem diferentes requisitos de QoS, logo, uma estratégia de agendamento deve ser desenvolvida considerando vários *workflows* com diferentes requisitos de QoS. Por este motivo foi apresentada uma estratégia para agendamento restrito, baseado em múltiplos parâmetros de QoS para vários *workflows* denominado MQMW. A estratégia pode agendar diversos *workflows* que são iniciados a qualquer momento levando em conta os diversos requisitos de QoS.

g) [Nathuji et al., 2010]

Em muitos serviços, os clientes são cobrados com base na quantidade de recursos usados ou reservados, porém não são feitas garantias em relação ao nível de desempenho das aplicações ou da qualidade dos serviços que os recursos devem proporcionar. Como os provedores da computação em nuvem continuarão utilizando tecnologias de virtualização em seus sistemas, a consolidação das aplicações dos clientes em servidores multicore introduz uma interferência de desempenho entre as cargas de trabalho das máquinas virtuais instaladas, podendo impactar significativamente uma aplicação de QoS. Para enfrentar esse desafio, [Nathuji et al., 2010] defende que a nuvem deve transparentemente prover recursos adicionais, sempre que necessário, para alcançar o desempenho que os clientes teriam alcançado se estivessem sendo executados de forma isolada. Com este propósito, foi desenvolvido o *Q-Clouds*, um *framework* para controle de QoS que ajusta as alocações de recursos para mitigar os efeitos de interferência de desempenho. *Q-Clouds* usa *feedback online* para construir um modelo com várias entradas e várias saídas que captura as interações de interferência de desempenho, para então usá-lo na gestão dos recursos.

h) [Ferretti et al., 2010]

Normalmente, uma aplicação ou serviço pode ser hospedado em uma plataforma de execução construído a partir de recursos reais ou virtuais da nuvem. Neste contexto, os requisitos de QoS podem ser especificados em um SLA que relaciona o aplicativo a sua plataforma de hospedagem. Em [Ferretti et al., 2010] é

proposto uma arquitetura que incorpora um balanceador de carga que distribui a carga computacional através dos recursos da plataforma e monitora a qualidade dos serviços que a plataforma oferece. Se acontecer alguma violação do que foi especificado no SLA, a plataforma é reconfigurada dinamicamente para incorporar recursos adicionais. Em contrapartida, se o SLA é honrado e os recursos da plataforma não estão sendo plenamente utilizados, a reconfiguração da plataforma ocorre para liberar estes recursos.

i) [Li et al., 2009]

O artigo de [Li et al., 2009] apresenta um método para otimização de nuvens usando modelos de desempenho no desenvolvimento, implantação e nas operações das aplicações em execução na nuvem. O artigo descreve a arquitetura da nuvem, os serviços oferecidos pela nuvem para suportar a otimização e a metodologia utilizada pelos desenvolvedores para permitir a otimização da nuvem em tempo de execução. É apresentado um algoritmo de otimização que acomoda diferentes objetivos, diferentes escopos e prazos para ações de otimização, e ainda algoritmos de controle diferentes. A otimização proposta objetiva maximizar lucros na nuvem baseado em QoS e SLA para uma grande variedade de cargas de trabalho.

j) [Kim et al., 2010]

Uma vez que, normalmente, os recursos computacionais são limitados, é muito importante que os provedores de nuvem utilizem eficientemente os seus recursos. Em [Kim et al., 2010] é apresentado um modelo de confiança para a reconfiguração e alocação eficiente de recursos computacionais, que pode satisfazer as várias solicitações dos usuários. Este modelo coleta e analisa a confiabilidade com base no histórico de informações dos servidores do datacenter da nuvem. Em seguida, ele prepara os melhores recursos disponíveis para cada solicitação de serviço, oferecendo os melhores recursos para os usuários.

k) [Goiri et al., 2010]

O trabalho de [Goiri et al., 2010] é focado no desempenho da CPU como métrica de qualidade. Os autores afirmam que as políticas atualmente utilizadas para SLAs falham no quesito confiança, especialmente quando os prestadores

terceirizam recursos para outros provedores. Esse problema acontece porque os provedores de recursos tipicamente apoiam métricas de avaliação muito simples, ou métricas que dificultam uma exploração eficiente dos seus recursos. Neste contexto, é proposto uma métrica em nível de recurso para especificar garantias sobre o desempenho da CPU. Esta métrica objetiva permitir que os provedores aloquem dinamicamente seus recursos entre os serviços executados, dependendo da sua demanda. Este objetivo é conseguido incorporando a utilização da CPU do cliente na definição de métrica, evitando falsas violações de SLA quando a tarefa do cliente não utiliza todos os seus recursos atribuídos.

I) [Vinek et al., 2011]

Os autores de [Vinek et al., 2011] acreditam que a seleção de serviços e fontes de dados para responder a uma determinada solicitação é uma tarefa crucial. Desta forma, atributos não funcionais ou de qualidade de serviço (QoS) precisam ser considerados quando existem vários serviços candidatos com funcionalidades semelhantes. Desta forma, antes de aplicar qualquer estratégia para otimização na seleção do serviço, o sistema precisa ser analisado em termos de métricas de QoS, semelhantemente as estatísticas necessárias para a utilização de um otimizador de consultas em banco de dados. Este artigo apresenta uma abordagem para classificação para os atributos de QoS dos componentes do sistema, a partir do qual funções de agregação de serviços compostos são derivados.

4.2.1 Comparativo de Atividades

A tabela 4.2 tem o objetivo de catalogar os artigos descritos na seção 4.2, relacionando-os através das principais atividades que cada um propõe realizar. Destacando-se desta forma, algumas das possibilidades de utilização de atributos de qualidade, fomentando possibilidades para a adequação da proposta, sendo adicionado na última coluna, índice X, quais aspectos são abordados pelo GerNU.

4.2.2 Abordagem do GerNU

Considerando a importância da utilização de parâmetros de qualidade

adequados para o contexto de nuvens computacionais, um dos objetivos desta pesquisa é propor parâmetros de qualidade relevantes para avaliar os serviços em nuvem. A definição destes parâmetros representa um desafio muito complexo, porque os mesmos devem identificar corretamente a qualidade dos serviços, sendo utilizados como base para um monitoramento regular, que tem o objetivo de elevar o nível de confiança entre o usuário e o provedor.

Como esforço inicial, foram definidos quatro parâmetros baseado no conceito de dependabilidade, sendo trabalho o contexto da classificação de requisitos para ser utilizado em políticas que beneficiam o usuário, tanto no processo de negociação quanto no monitoramento do serviço. Além disto, a partir da percepção da violação de alguma cláusula do SLA, o ambiente de trabalho poderá ser reconfigurado, possibilitando a adição ou remoção dinâmica de recursos.

Atividades	Artigo (identificado pelo índice)													
	a	b	c	d	e	f	g	h	i	j	k	l	X	
Ações Reativas	X													X
Alocação Recursos							X			X				
Balanceamento de Carga								X						
Desempenho							X		X		X			
Gerenciamento	X			X										X
Identificação de Parâmetros			X											X
Múltiplas Dimensões		X				X								
Negociação	X													X
Otimização									X					
Ranqueamento / Seleção				X	X								X	X
Reconfiguração Dinâmica								X		X				X
Workflow		X				X								

Tabela 4.2: Atividades dos artigos sobre QoS

4.3 ABORDAGEM DIRIGIDA A MODELOS

Existem muitos sistemas com o objetivo de fornecer mecanismos para criar a independência de plataforma e modelos de serviços tecnologicamente neutros, para então ser possível adaptá-los para uso e gestão em ambientes específicos. O modelo de arquitetura impulsionada pelo OMG, a MDA, descrito em capítulo anterior é uma dessas possibilidades. A MDA utiliza modelos para isolar a definição do

comportamento do sistema das tecnologias usadas para implementá-lo.

a) [Luck et al., 2002]

Um projeto para configurar adequadamente uma rede virtual privada ou *Virtual Private Network* (VPN) é geralmente difícil e propenso a erros, principalmente porque os objetivos abstratos do projeto são dados por políticas de alto nível que representam vários requisitos, desta forma os *designers* frequentemente são confrontados com requisitos conflitantes. Além disso, é difícil definir um mapeamento adequado entre políticas de alto nível para as configurações de rede de baixo nível, que implementem corretamente os objetivos abstratos. Em [Luck et al., 2002], é apresentada uma alternativa para modelagem de serviços, sendo descrito uma ferramenta de modelagem gráfica que permite a criação de modelos hierárquicos de serviços VPNs, bem como suas políticas de segurança correspondentes. Essa abordagem de gerenciamento é baseada em modelos, onde o próprio sistema, bem como os objetivos de gestão são representados por objetos gráficos e uma combinação de ferramentas e bibliotecas servem de suporte para uma construção interativa e uma análise automatizada.

b) [Garschhammer et al., 2001]

O trabalho [Garschhammer et al., 2001] se propõe a analisar as interações que ocorrem em um ambiente de serviços, considerando a existência de dois atores, um que oferece o serviço e outro que o solicita. Os autores consideram que, apenas examinando essas interações, é possível tirar conclusões sobre a funcionalidade do serviço, sem a necessidade de levar em conta a implementação do serviço, sendo relevante identificar estas interações. Portanto, é importante identificar estas interações e para tanto é utilizado o ciclo de vida dos serviços. Sendo assim, foi definido um modelo de serviço genérico em que um serviço é dividido considerando tanto aspectos do cliente quanto do provedor. Este trabalho propõe uma metodologia *top-down*, onde é utilizado um modelo para identificar componentes de ambos os lados e definir as associações de gestão, bem como as interações entre estes componentes de uma forma inequívoca. As interações ocorrem entre pares que representam unidades organizacionais de ambos os lados, sendo que os papéis são atribuídos a classes de interação. A partir da análise das classes de interação e dos

papéis identificados, será possível especificar interfaces e entidades que participam no provisionamento dos serviços.

c) [Rodosek, 2003]

Em [Rodosek, 2003] é apresentado um modelo de serviço genérico que define pontos de vistas para os serviços baseado em três aspectos: o próprio serviço - refere-se aos elementos de um serviço que são independentes de quaisquer questões relacionadas ao provedor ou ao cliente; o provedor - aborda o aspecto de que as funcionalidades de um serviço podem ser fornecidas de diferentes maneiras por diferentes provedores; e o cliente - considera que um serviço pode ser oferecido a vários clientes, o aspecto relevante para este contexto são os parâmetros de qualidade. Esses modelos podem ser eficazmente utilizados como uma base de informações compartilhadas para várias aplicações, definindo elementos que devem ser incluídos em um SLA.

d) [Miezner et al., 2009]

No trabalho [Miezner et al., 2009], foi descrito como técnicas de modelagem da variabilidade em linhas de produto de *software* podem apoiar os provedores de SaaS no gerenciamento da variabilidade em aplicações SaaS e seus requisitos. Especificamente, propôs-se utilizar modelos de variabilidade explícitas para sistematicamente derivar informações individuais de customização e implantação dos inquilinos de SaaS. Além disso, foi demonstrado como modelos de variabilidade podem ser utilizados, considerando informações sobre os aplicativos SaaS já disponibilizados, de forma que seja possível implantar eficientemente aplicações SaaS para novos inquilinos. Foram aplicados os conceitos de variabilidade externa e interna utilizados em linhas de produto de *software*, para resolver o problema de customização e implantação em aplicativos SaaS.

e) [Zhu and Wang, 2009]

O artigo de [Zhu and Wang, 2009] descreve o projeto e a implementação de um *Service Template Markup Language* (STML) e suas ferramentas de desenvolvimento integradas, com o objetivo de fornecer uma solução abrangente para a personalização de aplicações em plataformas SaaS baseada nas

necessidades individuais dos usuários. Foi proposto um *framework* baseado em uma abordagem MDA para customização de SaaS que possui o objetivo não só de melhorar a eficiência do desenvolvimento de *software*, mas também melhorar a flexibilidade dos sistemas de *software*. Além destas vantagens é apontado a redução da complexidade no desenvolvimento de uma aplicação *on-demand* a uma simples definição dos serviços.

f) [Srinivasmurthy et al., 2009]

Considerando que um dos relevantes requisitos atuais no contexto de serviços é a agilidade em disponibilizá-los, sendo que os provedores que não conseguirem andar nessa tendência, facilmente ficarão fora do mercado. Novos serviços podem ser implementados a qualquer momento, porém existe um valor substancial nas aplicações e sistemas legados, porém disponibilizá-los como serviço é um grande desafio, devido a complexidade pertinente a integração. Desta forma, [Srinivasmurthy et al., 2009] propõe uma abordagem para integração onde o foco está sobre a questão da especificação do serviço, e para isso são utilizados modelos para capturar o escopo, a capacidade e o estado de serviço. Também é fornecido um protótipo denominado Web2Exchange, uma plataforma para modelagem, transformação e integração dos serviços que permite que serviços anteriormente incompatíveis sejam combinados mais facilmente.

g) [Pruyne and Singhal, 2008]

Este trabalho afirma que, normalmente, as descrições de serviço contêm dois componentes: uma interface e uma localização. O primeiro componente é utilizado para definir o que o serviço pode fazer, enquanto que o segundo define como ou onde o serviço pode ser acessado. Estes componentes são importantes, mas não são suficientes para gerar integrações bem sucedidas, sendo necessárias uma série de outras informações de forma que a definição de um serviço seja visto como um modelo, [Pruyne and Singhal, 2008]. Desta forma, um modelo é definido como uma representação formal das operações, estados, entidades associadas e informações dos meta-dados que são expostas pelo serviço aos clientes dentro de um ambiente orientado a serviços ou *Services Oriented Environment* (SOE). Esta proposta tem sua estrutura composta por três partes: Um ambiente de integração de serviços ou

Services Integration Environment (SIE), baseado em modelos e fornece a capacidade para interpretar e operar sobre modelos dentro do sistema; um conjunto de operações modelos, sobre as quais, os serviços são construídos; e os serviços específicos de domínio que fornecem os recursos necessários para cada domínio. Este ambiente utiliza modelos declarativos para representar instâncias em tempo de execução. Estes modelos descrevem o estado real do serviço em tempo de execução, permitindo uma discussão sobre os serviços individuais e compostos, tanto em tempo de projeto quanto em tempo de execução.

h) [Ritu and Manu, 2011]

Os serviços em uma nuvem podem ser executados em diferentes plataformas. Como é evidente, as tecnologias evoluem constantemente e ainda outras tantas surgem rapidamente, desta forma, muitos serviços baseados em tecnologias específicas, tendem a se tornar obsoletos e precisam ser substituídos. Assim sendo, é relevante a adoção de metodologias de desenvolvimento de serviços, onde o impacto dos avanços tecnológicos seja mínimo. Dentro desta perspectiva, o trabalho [Ritu and Manu, 2011], considera que o desenvolvimento de software dependente de tecnologia não é viável em longo prazo, principalmente levando em conta o surgimento acelerado e constante de novas tecnologias. Por isso, os autores descrevem uma abordagem para o desenvolvimento de serviços em nuvens que é independente de tecnologias específicas, utilizando a MDA para garantir que os avanços tecnológicos não causem impactos nas aplicações, ou que, se forem inevitáveis, os mesmos sejam mínimos. Com a utilização desta abordagem, intencionam aumentar a longevidade e capacidade de reutilização dos serviços desenvolvidos em uma nuvem. Este artigo objetiva incorporar a metodologia de desenvolvimento de software orientado a modelos ao desenvolvimento de SaaS em nuvem. Como já visto anteriormente, os modelos em MDA são definidos em três níveis de abstração, CIM, PIM e PSM e o sucesso da MDA depende da automatização das transformações modelo-modelo e modelo-código. Em vez de desenvolver diretamente os serviços de software, estes serviços são modelados como prescreve a MDA, de forma a abstrair as dependências tecnológicas dissociando-se então os efeitos indesejáveis da mudança de

tecnologia.

i) [Cochinwala et al., 2005].

Uma abordagem dirigida a modelos objetivando a introdução rápida de serviços é apresentada por [Cochinwala et al., 2005]. Esta proposta relata que a necessidade manual de escrever e integrar códigos personalizados de sistemas legados pode ser substituído por uma interface de mapeamento entre os componentes dos serviços e os componentes de gerenciamento do sistema. Os serviços são representados como modelos de dependência, a partir dos quais, mapeamentos para interfaces de gerenciamento de componentes individuais dos sistemas de suporte podem ser derivados. O projeto é uma ferramenta gráfica desenvolvida para facilitar a criação dos modelos de serviços, sendo a concepção dos mesmos limitada por um modelo de meta-serviço, que delimita os objetos permitidos e as relações entre eles. O meta-modelo contém objetos de serviço e objetos de recursos, onde, semanticamente, um recurso é utilizado para representar uma entidade física, enquanto que um serviço representa as propriedades de entidades abstratas que implementam alguma lógica de serviço. Tanto serviços e recursos podem conter um número ilimitado de atributos. A ferramenta do serviço suporta não só a definição desses atributos, mas também a especificação de restrições sobre os valores dos atributos. Este meta-modelo facilita a criação de modelos centrados em dados de serviços com associações de dependência definidas entre as entidades do modelo. Ele também oferece suporte integrado para a reutilização dos modelos de serviços ou recursos existentes na definição de novos modelos.

j) [Xiaoyan et al., 2010]

Uma plataforma para aplicativos SaaS baseada na abordagem dirigida a modelos é apresentada por [Xiaoyan et al., 2010]. Esta plataforma fornece uma solução para realizar a personalização e integração de plataformas. Neste artigo é apresentada uma visão geral do framework, uma série de mecanismos para trabalhar com modelos, *templates* de modelos e arquivos com a descrição dos modelos. Em seguida, é descrito como todos estes artefatos trabalham juntos em uma arquitetura multi-clientes, e os aspectos sobre a personalização e integração de

serviços na plataforma SaaS, para então demonstrar como a plataforma trabalha. A arquitetura geral da plataforma baseado na abordagem dirigida a modelos foi concebida a partir de três aspectos: conhecimento multi-cliente, personalização e integração de serviços. Os principais módulos são a *Execution Platform* (EP) e o *Core Model-driven Engines* (CME). O EP tem como objetivo principal a gestão dos serviços de execução, incluindo os serviços de aplicativos comuns e serviços privados baseados na identificação dos usuários. O CME é o núcleo principal de negócios e a camada de controle dos modelos de gestão. A parte básica da arquitetura geral pode ser descrita em duas camadas. A primeira é a camada de modelo que descreve os dados, processos de negócios e interface de usuário. A segunda é a camada de *engines*, baseada em modelos que alimentam os modelos de análise.

k) [Sanz et al., 2008]

Considerando que existe uma falta de metodologias de desenvolvimento precisas que sejam aplicadas a orientação a serviços, conforme [Sanz et al., 2008], a partir da abordagem de desenvolvimento dirigido a modelos, e mais especificamente em MDA, propõe a definição de um modelo de Arquitetura Orientada a Serviços Software (SOA), dentro de um *framework* metodológico dirigido a modelos denominado MIDAS, para o desenvolvimento adaptável e flexível de sistemas. A arquitetura do MIDAS é composta por três eixos: Eixo vertical (Y): reflete os três níveis de abstração da proposta original do MDA; Eixo Horizontal (X): Modelos são separados nas diferentes preocupações envolvidas no desenvolvimento do sistema de informação; Eixo Transversal (Z): Modelos neste eixo são referências aos aspectos que têm influência sobre outros modelos do eixo transversal. A arquitetura é dirigida pelos aspectos do processo de desenvolvimento que permitem especificar quais os modelos, elementos no interior dos modelos ou relações dentro dos modelos devem ser criados durante todo o processo de desenvolvimento de *software*. Com uma visão de arquitetura do sistema a nível PIM, garante-se que não haverá restrições de tecnologia ou implementação ao modelo. Além disso, é facilitado a criação de diferentes modelos no nível PSM, de acordo com a plataforma alvo específica, todos derivados de um modelo PIM único. O

SOSA pode ser utilizado como uma maneira de integrar aplicações empresariais.

1) [Xiao et al., 2009]

Um framework para linha de produção, orientado a serviço e baseado em modelos é apresentado em [Xiao et al., 2009]. São descritos a arquitetura, os métodos de integração de ferramentas bem como a interação entre elas. O método dirigido a modelo do processo de produção também é apresentado. Foi proposto um algoritmo heurístico de escalonamento que visa a produção em escala, sendo descrito como criar e executar uma linha de produção e seus princípios teóricos. Nesse trabalho, o modelo representa o núcleo da camada de integração, padronizando o mecanismo de operação das ferramentas e sendo responsável pela realização de toda lógica dos negócios nas ferramentas. A linha de produção é projetada e configurada utilizando-se a metodologia “arrastar e soltar” e a MDA automatiza a execução da linha de produção de acordo com a regra semântica dos elementos gráficos. Este método melhora a eficiência de execução da linha de produção através da redução da intervenção humana e de proteção contra erros de configuração em transmissões múltiplas.

4.3.1 Comparativo de Atividades

A tabela 4.3 a seguir, tem o objetivo de catalogar os artigos descritos anteriormente, relacionando-os através das principais atividades que cada um propõe realizar. Destacando-se desta forma, algumas das possibilidades de utilização do contexto de modelos e de forma objetiva, sendo fomentando possibilidades para a adequação da proposta, sendo adicionado na última coluna, índice X, quais aspectos são abordados pelo GerNU.

4.3.2 Abordagem do GerNU

A abordagem dirigida a modelos é relevante neste trabalho por modelos podem conter diversos tipos de informação, tais como, dados do usuário, especificação de *hardware* e *software*, dentre outros, podendo ser facilmente compartilhadas, além de agragar flexibilidade ao processo de especificação do

serviço, inclusive coletando-se informações que poderão ser empregadas em diferentes estratégias utilizadas na solução.

Estas informações serão utilizadas para criar um ambiente virtual e para monitorar a qualidade do serviço. O GerNU utiliza modelos com o objetivo de criar estruturas independentes de tecnologia que podem ser manipuladas para gerar outros modelos, através do mapeamento de suas informações para estruturas específicas. Desta forma, todas as atividades do sistema acontecem a partir dos modelos, ocorrendo uma padronização para a operacionalização das funcionalidades do sistema.

Atividades	Artigo (identificado pelo índice)												
	a	b	c	d	e	f	g	h	i	j	k	l	X
Adaptabilidade											X		
Agilidade de Disponibilização						X			X				X
Análise		X											
Automatização de Processos	X												X
Customização				X	X					X			X
Flexibilidade											X		X
Implantação				X									X
Independência de Tecnologia								X					X
Integração						X	X			X	X		
Mapeamento	X								X				X
Padronização			X									X	X

Tabela 4.3: Atividades dos artigos sobre Modelos

4.4 CONCLUSÃO

Pode-se observar que existe um esforço muito grande em se identificar métricas eficientes que de fato consigam refletir a qualidade de um serviço. Em muitos e distintos domínios os pesquisadores e a indústria almejam identificar eficientemente estes atributos. Dentro dos domínios, as especificidades de cada serviço conduzem a um diferente tipo de necessidade que diretamente implica em que características particulares sejam consideradas como métricas.

Esta diversidade de requisitos, domínios e especificidades elevam os temas relacionados a SLA e a QoS a serem continuamente relevantes, pois sempre haverá busca por melhorias e pare se estabelecer garantias que devem atrair os usuários,

que por sua vez, estão cada vez mais exigentes e conscientes do que querem, buscando mecanismos confiáveis para verificar se de fato estão sendo atendidos em conformidade com o que estão pagando.

Neste capítulo foram discutidas uma série de relevantes artigos disponíveis na literatura sobre os temas parâmetros de qualidade de serviços, contratos em nível de serviços e abordagens baseadas em modelos. Foram destacadas as principais atividades a que se relaciona cada trabalho.

No próximo capítulo será apresentado uma abordagem para o especificação e aquisição de serviços em nuvens, onde serão descritos seus objetivos, suas características principais, especificações, arquitetura e funcionalidades.

5. GERNU, UMA ABORDAGEM PARA ESPECIFICAÇÃO E PROVISIONAMENTO DE SERVIÇOS

Este capítulo descreve o **GerNU**, cujo os propósitos iniciais são: definir um processo padrão para a especificação e aquisição dos serviços, instanciar de forma dinâmica e automática os serviços, a partir da especificação do usuário e ainda definir parâmetros que possam garantir a qualidade dos serviços.

Desta forma, será feita uma introdução, descrevendo os problemas iniciais que o GerNU pretende atender, destacando-se as perspectivas consideradas e os cenários de implementação vislumbrados. Na sequência, será descrito uma visão geral do desenvolvimento do GerNU, objetivando descrever como o sistema atende os desafios anteriormente elencados. Por fim, será feito um detalhamento nas principais funcionalidades e estruturas que foram desenvolvidas para atender as atividades inerentes a solução proposta.

5.1 INTRODUÇÃO

Considerando o grande crescimento do paradigma de nuvens computacionais, temos o objetivo de descrever uma abordagem para especificação e aquisição de serviços denominado **GerNU**. Para esta fase do projeto, a qual esta tese está incluída, os objetivos estão relacionados com a intenção de apresentar soluções eficientes para três importantes desafios presentes no ambiente de nuvens:

- 1. Padronização do Processo de Especificação e Aquisição dos Serviços;**
- 2. Disponibilização Automática dos Serviços;**
- 3. Definição de Parâmetros para a Qualidade dos Serviços.**

A solução proposta pretende resolver estes três desafios levando em consideração às três perspectivas descritas a seguir:

a) Perspectiva do Usuário Comum

Considera uma pretensão muito comum entre os usuários de nuvens, o desejo de simplesmente utilizar os recursos de uma nuvem computacional, sem a preocupação com a instalação, configuração e manutenção de *hardware* e *software*. O fato é que não é explicitamente relevante como as coisas funcionam, o importante é que exista um caminho fácil para utilização e o serviço prestado seja confiável.

b) Perspectiva do Usuário Experiente

Esta perspectiva já leva em consideração usuários mais específicos, usuários da área de TI que geralmente já trabalham com o paradigma de nuvem e conhecem seu potencial. O objetivo é o desenvolvimento de mecanismos que automatizem os procedimentos para a construção e disponibilização de ambientes virtuais, de forma que seja possível ao usuário manter seu foco em outros aspectos do negócio.

c) Perspectiva do Provedor a Nuvem

Como não poderia ser desconsiderada, esta perspectiva tem a intenção de atender a necessidade do provedor. O propósito é disponibilizar a automatização de alguns procedimentos do ambiente de nuvem, diminuindo a intervenção humana durante os processos, diminuindo o tempo de resposta para procedimentos rotineiros e ainda se tornando menos suscetível a erros.

Desta forma, visualizamos um cenário que envolve o paradigma de nuvens no qual será utilizado uma abordagem dirigida a modelos, de forma a ser possível estabelecer um padrão para o processo de especificação e aquisição dos serviços, possibilitando sua criação automática e dinâmica, contemplando a verificação da sua qualidade, sendo que todos estes processos consideram as reais necessidades dos usuários. Sendo assim, foram contempladas três possibilidades de implementação que serão posteriormente avaliadas no Capítulo 6 onde são descritos os Testes e a Avaliação da nossa abordagem. Estas três perspectivas são brevemente descritas a seguir:

1ª Perspectiva - O provedor da nuvem desenvolve um mecanismo que

realiza um mapeamento do modelo de ambiente virtual do GerNU para sua própria estrutura de imagens virtuais, de forma que seja possível instanciar o serviço, mesmo que originalmente criado em outro provedor. Este é o panorama menos provável devido a grande dependência externa.

2ª Perspectiva - Adicionar a abordagem proposta um mecanismo de mapeamento que realize um processo de conversão das informações do modelo de ambiente virtual do GerNU para um ou mais tipos específicos de imagens utilizados pelos provedores de nuvem. Esta possibilidade é mais factível, apresentando um pequeno nível de dependência, onde se faz necessário apenas o estabelecimento de um mecanismo de comunicação entre o GerNU e o provedor para troca de informações necessárias (tais como dados do usuário e aspectos segurança) para a criação um ambiente virtualizado, onde o provedor terá o trabalho de simplesmente instanciar o serviço

3ª Perspectiva - A partir do modelo de ambiente virtual, o sistema proposto deve automaticamente e dinamicamente criar e disponibilizar os serviços para os usuários. Esta possibilidade não introduz nenhuma dependência de terceiros para prover um serviço. O GerNU deverá executar todos os procedimentos necessário para o provisionamento dos serviços em uma nuvem e ainda poderá ser utilizado para disponibilizar ambientes virtuais autônomos para ambientes privados.

Na abordagem utilizada pelo GerNU será utilizado um modelo para o ambiente virtual da nuvem, seu objetivo é servir como base para os processos de criação, gerenciamento, monitoramento e atualização dos serviços.

Esta abordagem apresenta o diferencial de mesclar as negociações das garantias e requisitos do SLA durante o processo de aquisição do serviço, servindo então o SLA resultante como fonte para extração de modelo do ambiente virtual, que servirá como base para os demais procedimentos no sistema, viabilizando informações para os processos que irão verificar e assegurar as garantias. Futuramente, estas informações poderão também ser utilizadas para auxiliar a alocação dos recursos.

O objetivo é que na prática, durante o processo de aquisição de um serviço, o cliente possa especificar seus requisitos, negociando-os com o provedor, de forma a estabelecer um SLA que atenda suas necessidades e obviamente possa ser suportado pelo provedor, considerando a extrema necessidade que a qualidade dos serviços seja sempre preservada. Assim sendo, o GerNU representa um projeto que utiliza uma abordagem dirigida a modelos para disponibilizar Infraestruturas como Serviço, Plataformas como Serviço e Softwares como Serviço, gerenciando a criação automática das máquinas virtuais que darão suporte aos serviços especificadas pelo usuário, sendo ainda proposto a utilização de SLA para controlar os recursos da nuvem e a qualidade dos serviços baseado em atributos contextualizados ao ambiente de nuvens.

5.2 VISÃO GERAL DA PROPOSTA

A proposta desta subseção é retratada pela figura 5.1 que claramente descreve a metodologia geral adotada para o desenvolvimento do GerNU, sendo que o primeiro passo foi fazer a definição de um processo flexível para a especificação e aquisição de serviços na nuvem. Este processo tem como característica marcante a possibilidade que o usuário dispõe para definir os requisitos do seu serviço. O segundo momento envolveu o projeto do cenário que servirá o usuário, permitindo que o usuário faça sua especificação, atendendo as especificidades do processo de aquisição. O terceiro e último passo envolve a especificação dos agentes do sistema que são responsáveis por controlar todas as funcionalidades do GerNU.

5.2.1 Processo para Especificação e Provisionamento de Serviços no GerNU

A partir do processo de observação da metodologia para aquisição de serviços disponibilizada nos provedores de nuvens mais populares no mercado, percebe-se que tipicamente este processo obedece a sequência exibida na figura 5.2, sendo cada uma destas etapas descritas a seguir.

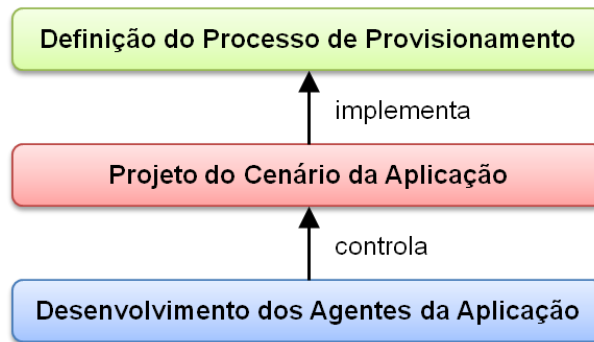


Figura 5.1: Visão geral do desenvolvimento do GerNU

1. Registro no Provedor - a primeira tarefa que usualmente o cliente deve realizar é a criação de um cadastro no provedor, para então poder acessar as opções disponibilizadas para aquisição do serviço.

2. Seleção da Imagem Virtual - o usuário deverá selecionar, dentre as imagens virtuais disponíveis, a que seja mais adequada aos propósitos do seu serviço. Nestas imagens estão contidas o sistema operacional com aplicações instaladas, prontas para utilização.

3. Configuração da Instância - este processo representa a configuração de *hardware* dos recursos que serão disponibilizados para o serviço, podendo também ser feita a definição de características de segurança do ambiente.

4. Provisionamento do Serviço - após toda a configuração do ambiente virtual, esta tarefa corresponde ao efetivo processo para criação de uma máquina virtual, de forma que o serviço seja disponibilizado para o usuário.

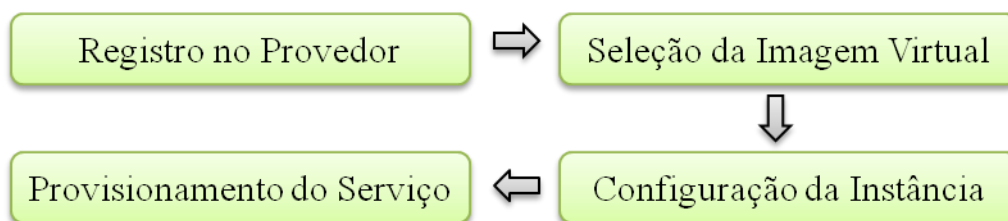


Figura 5.2: Processo usual para provisionamento de serviços em nuvem

Apesar de ser um fato que são disponibilizados diversos tipos de imagens, representando diferentes combinações de hardware e software, mesmo assim, de certo modo este processo usual pode ser considerado inflexível, uma vez que um

usuário não poderá especificar exatamente os requisitos que deseja contratar, sendo imposto aos mesmos apenas as opções previamente disponibilizadas. Desta forma, objetivando que a criação de serviços em nuvens possa atender exatamente as necessidades dos usuários, de forma que os aspectos relacionados com a definição de hardware, sistema operacional, aplicações de desenvolvimento, aplicações de produção (como editor de texto, planilhas, dentre outros) e a definição dos parâmetros de qualidade, sejam realizados de maneira intuitiva e flexível, sendo todas estas informações estruturadas em um arquivo como uma imagem virtual, que tem o objetivo de documentar a estrutura de uma máquina virtual, o GerNU definiu o processo de aquisição de serviços representado pela figura 5.3

O processo proposto é baseado em modelos e leva em consideração a qualidade dos serviços, objetivando preservar as garantias do SLA. Este processo é operacionalizado a partir de um modelo do ambiente virtual, sendo que a partir deste, poderá ser possível importar e exportar serviços entre provedores de nuvem, através do mapeamento do modelo utilizado pelo GerNU para a estrutura específica do provedor. As atividades especificadas pelo processo foram definidas para atender as necessidades dos usuários, relacionados a *hardware*, *software* e parâmetros de qualidade (cada provedor deverá especificar exatamente quais parâmetros podem ser suportados por sua infraestrutura), ao contrário da habitual imposição de imagens virtuais *pré-fabricadas* feitas pelos provedores de nuvem.

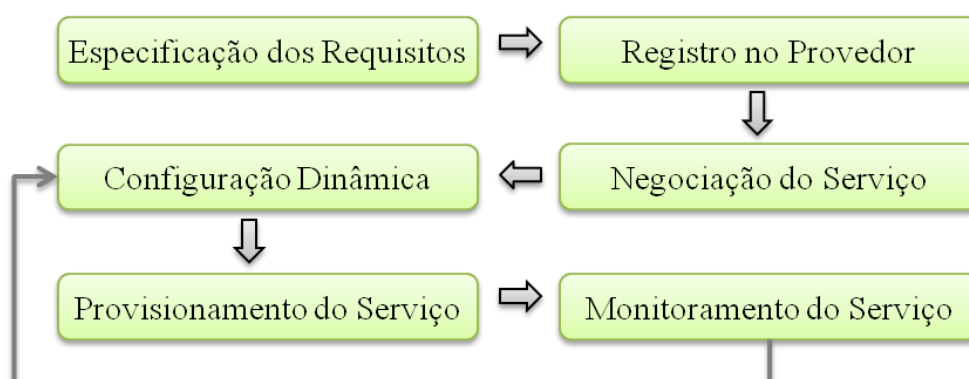


Figura 5.3 Processo proposto para provisionamento de serviços em nuvem

1. Especificação de Requisitos - O propósito desta atividade é permitir que um usuário possa especificar com exatidão para seu serviço seus requisitos

relacionados a *hardware* (CPU, memória, armazenamento, outros) e *software* (sistemas operacionais e aplicações), necessários ao seu ambiente virtual. Além disto, o usuário deve especificar os atributos para cada parâmetro de qualidade que deseja contratar. Este é um flexível processo de criação de serviços personalizados que leva em consideração os recursos disponíveis com o objetivo de garantir o SLA acordado.

2. Registro no Provedor - O cliente deve criar uma conta no provedor para contratar um service. São necessárias informações pessoais como email, número do cartão de crédito e outros. Através das informações desta conta será possível acessar os serviços e serem realizados contatos, cobranças e reembolsos.

3. Negociação do Serviço - Esta atividade objetiva disponibilizar um mecanismo para negociação dos requisitos de hardware, parâmetros de qualidade e valores dos serviços, sendo possível negociar maiores ou menores quantidades destes itens. Além do que esta atividade deve ser a mais automatizada possível.

4. Configuração Dinâmica - O ambiente virtual deve representar exatamente os requisitos especificados pelo usuário, desta forma, esta atividade é responsável por criar automaticamente e dinamicamente a estrutura virtual que servirá de suporte para o serviço personalizado do usuário. Esta é uma tarefa com um elevado nível de complexidade, principalmente considerando o ambiente distribuído e heterogêneo da computação em nuvem.

5. Provisionamento do Serviço - Esta atividade está relacionada com a efetiva entrega do serviço para o usuário, devendo acontecer após toda a especificação do serviço, sendo necessário um gerenciamento do acesso aos serviços, de forma a preservar os dados do usuário.

6. Monitoramento do Serviço – Considerando-se que, a relevância do serviço para o usuário não encerra após o mesmo ser disponibilizado para utilização, assim, deve ser iniciado o monitoramento dos parâmetros de qualidade e do ambiente virtual como um todo, objetivando garantir o SLA acordado com um serviço eficiente e confiável.

Resumindo o propósito deste processo, a partir das atividades propostas pelo

mesmo, considerando que as informações relevantes para cada uma das atividades estarão estruturadas em um modelo do ambiente virtual, será possível instanciar o serviço automaticamente e garantir sua qualidade através do alto nível de flexibilidade proporcionado pela especificação do serviço.

5.2.2 Cenário Proposto

Considerando o contexto descrito na introdução deste capítulo e objetivando atender o processo de aquisição de serviços especificado na subseção 5.2.1, foi desenvolvido pelo GerNU um cenário com suas funcionalidades representadas pela figura 5.4.

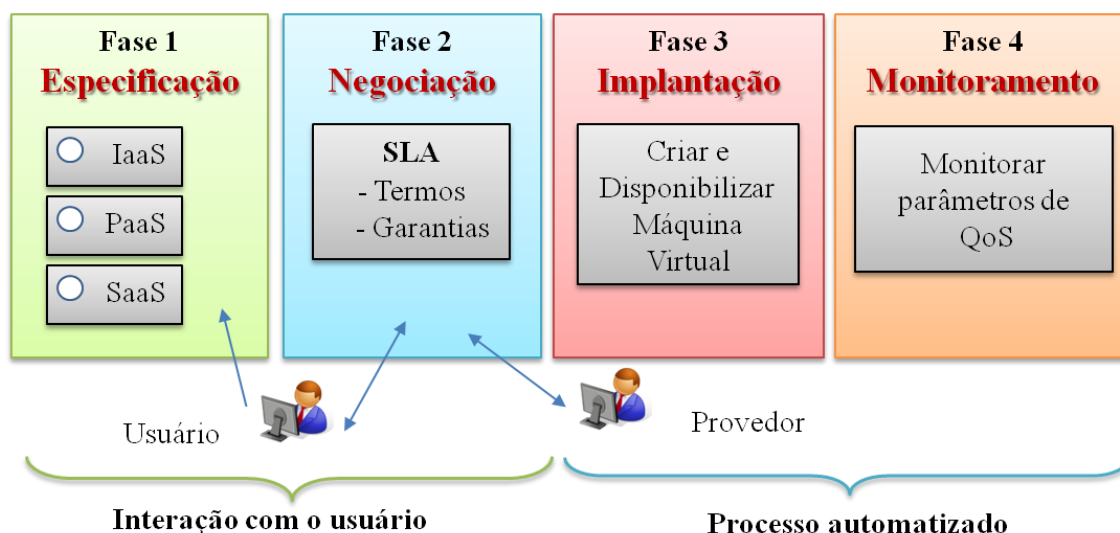


Figura 5.4: Cenário de Aplicação do GerNU

Este cenário tem o objetivo de disponibilizar um ambiente no qual o usuário, de forma simplificada, especifique suas necessidades, negocie os termos de garantias dos serviços, bem como seus valores e então dinâmica e automaticamente tenha o serviço disponível e monitorado na nuvem. Este cenário será detalhado a seguir. A finalidade desejada é que este cenário esteja disponível em um portal web, dado a atual facilidade de acesso a internet, intencionando-se com isto uma maior visibilidade do produto e como consequência, uma maior utilização.

Como pode ser observado na figura 5.4, que representa a especificação do cenário de trabalho / funcionalidade do GerNU, o processo de utilização do mesmo

consiste em quatro fases, nas quais o cliente interage com a primeira e a segunda, especificando e negociando suas intenções, enquanto que a terceira e quarta fases estão sob a responsabilidade de processos automatizados, estando os mesmos relacionados com a implantação e monitoramento do serviço.

O objetivo geral é disponibilizar para os usuários um ambiente virtual de trabalho pronto para uso, para tanto, o usuário deverá especificar seu ambiente virtual, conforme sua conveniência, intuitivamente selecionando as opções disponíveis. Após a especificação, devem ser negociados os termos do contrato e na sequência terá disponibilizado seu serviço (IaaS, PaaS ou SaaS), que por sua vez deverá ser continuamente monitorado para assegurar o SLA. O propósito de cada uma destas fases, bem como sua utilização e principais interações são descritas a seguir:

5.2.2.1 Fase 1 - Especificação

A primeira definição que deverá ser feita no ambiente do GerNU está relacionada com o tipo de serviço desejado, sendo inicialmente possível optar-se entre IaaS, PaaS e SaaS. A escolha por um serviço do tipo IaaS, provavelmente será feita por usuários mais experientes que já possuem conhecimentos suficientes para manipulação de infraestruturas em nuvens. Para este caso, será necessário especificar os requisitos de *hardware* desejados e fazer a definição dos atributos para os parâmetros de QoS do serviço, devendo ser especificadas as informações que podem genericamente ser observadas a partir da figura 5.5.

Especificação IaaS			
Hardware		QoS	
Tipo	Qtd	Parâmetro	Atributo
CPU	<input type="text"/>	1	<input type="radio"/> A <input type="radio"/> B <input type="radio"/> C
Memória	<input type="text"/>	2	<input type="radio"/> A <input type="radio"/> B <input type="radio"/> C
Disco	<input type="text"/>	3	<input type="radio"/> A <input type="radio"/> B <input type="radio"/> C
Largura de Banda	<input type="text"/>	N	<input type="radio"/> A <input type="radio"/> B <input type="radio"/> C

Figura 5.5: Especificação de um IaaS

No caso da seleção do usuário ser um serviço do tipo PaaS, além do que é necessário para uma IaaS, devem ser especificados qual o sistema operacional melhor atende suas necessidades e ainda quais aplicações de desenvolvimento são necessárias e devem estar instaladas na sua plataforma. O GerNU disponibilizará uma lista de sistemas operacionais e aplicações de desenvolvimento que podem ser instaladas na plataforma. Obviamente, as aplicações são dependentes do sistema operacional, ou seja, quando for selecionado, por exemplo, o sistema operacional Ubuntu Server 11.10 - 64 bits serão disponibilizadas apenas as aplicações próprias para este sistema operacional. A figura 5.6 retrata genericamente esta opção, sendo que a lógica utilizada pelo GerNU para a disponibilização das aplicações, baseado no tipo do sistema operacional, também está descrito a seguir através do Algoritmo 1. A partir da linha 3, pode-se observar que será realizado uma consulta do banco de dados do sistema com o objetivo de recuperar todas as aplicações que estão disponíveis para um dado sistema operacional. Se a aplicação estiver ativa, isto é, sem nenhuma restrição para instalação, ela será disponibilizada para a seleção do usuário.

O diagrama, intitulado "Especificação PaaS", apresenta duas listas de opções. A primeira lista, localizada em um campo verde, contém opções de sistemas operacionais: "Sistema Operacional 1", "Sistema Operacional 2", "Sistema Operacional 3", "..." e "Sistema Operacional N". A opção "Sistema Operacional 2" está selecionada, indicada por um círculo preenchido. A segunda lista, localizada em um campo amarelo, contém opções de aplicações: "Aplicação 1", "Aplicação 2", "Aplicação 3", "..." e "Aplicação N". Todas as opções de aplicações possuem um campo de seleção vazia (caixa de seleção).

Figura 5.6: Especificação de um PaaS

Quando a opção do usuário for um serviço do tipo SaaS, duas opções serão possíveis, na primeira pode ser selecionada uma *appliance* (plataforma virtualizada pronta para uso), com o software disponível no repositório do sistema. Esta opção requer um esforço adicional no sentido de desenvolver este modelo especial que inclusive inclui o sistema operacional e as aplicações de suporte necessárias, sem ser disponibilizado acesso aos mesmos, sendo disponibilizado exclusivamente a aplicação principal. Desta forma, será necessário especificar os requisitos de

hardware e parâmetros para QoS.

Algoritmo 1 - Seleção de Aplicações

```
1:  SelecionarAplicacoes (){
2:    Entrada: SistemaOperacional;
3:    Array app = buscar_app (SistemaOperacional);
4:    for i = 0 to i < app.length
5:      if app[i].status = true then
6:        exibir_app (app[i]);
7:      end if
8:    end for
9:  }
```

A segunda opção considera que o cliente precisa de uma aplicação específica que ainda não dispõe de uma *appliance*, sendo necessária uma negociação personalizada, com o objetivo de verificar a viabilidade para implantação da aplicação, definindo-se os valores para o serviço, podendo esta perspectiva ser observada genericamente na figura 5.7.

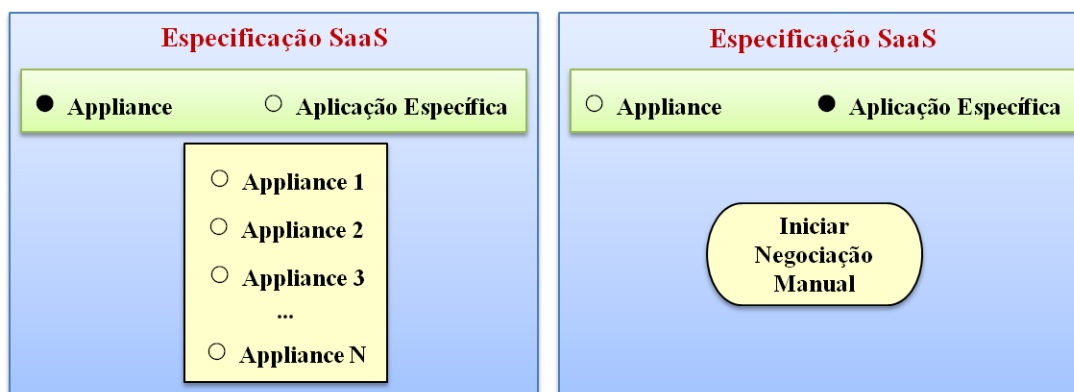


Figura 5.7: Especificação de um SaaS

Inicialmente, tanto para PaaS quanto para SaaS, serão disponibilizados apenas sistemas operacionais, aplicações de desenvolvimento e aplicações de produção (*appliances*) mais comuns no mercado, contudo opções não contempladas originalmente poderão ser incluídas posteriormente, de acordo com a demanda, sendo possível ao usuário requerer tanto aplicações quanto sistemas operacionais novos.

O sistema disponibiliza três configurações de hardware, Mínima, Flexível e Robusta, também sendo possível que o usuário faça uma definição personalizada destes requisitos, especificando exatamente a quantidade para cada item de *hardware* sendo todo o processo de especificação realizado através de simples seleções na interface gráfica do GerNU, tornando o procedimento simples e rápido.

Associados a especificação dos serviços, serão automaticamente disponibilizados os valores dos mesmos, de forma a facilitar a decisão do usuário em contratar o serviço. Estes requisitos e valores podem ser negociados posteriormente, bem como os parâmetros para os atributos de qualidade, para tanto se faz necessário passar para a fase seguinte.

5.2.2.2 Fase 2 - Negociação

Após a especificação do serviço, são iniciadas as atividades relacionadas com a primeira etapa desta fase, envolvendo a realização do cadastro do usuário no sistema, de forma a viabilizar contatos, cobranças e ressarcimentos. A segunda etapa ocorre após a finalização da primeira, sendo fornecido um SLA padrão gerado pelo sistema, especificando termos e garantias para utilização dos serviços. Além disto, serão disponibilizados atributos para cada um dos parâmetros de qualidade, tendo sido estes previamente definidos pelo administrador da nuvem. Todas estas informações são previamente cadastradas no banco de dados do sistema, servindo então como ponto de partida para o processo de negociação.

Os requisitos de *hardware* também são denominados de requisitos restritivos, pois dependendo das quantidades especificadas no momento da requisição, considerando o tempo gasto pelo usuário durante o processo de especificação, que efetivamente não pode ser previsto, o provedor pode não ter mais os recursos disponíveis, sendo então estes, os primeiros elementos a serem avaliados, considerando-se a infraestrutura da nuvem. De forma concisa, se um usuário requer, por exemplo, 10 megabytes de armazenamento, e o provedor não possui esta quantidade disponível, este fato cria uma restrição definitiva para a prestação do serviço.

Se os limites providenciados pelo sistema forem suficientes para atender as

necessidades do usuário, um acordo é estabelecido de imediato e o processo continua automaticamente. Caso contrário, os parâmetros e os seus custos podem ser negociados através de uma interface própria para negociação, onde novas propostas podem ser requisitadas pelo usuário, sendo possível gerar propostas até um limite definido pelo administrador. Esta etapa está sujeita a intervenção humana, onde um administrador poderá ter que avaliar cada proposta, caso as oscilações estejam fora de parâmetros previamente configurados, o que inviabiliza uma assimilação automática.

A qualquer momento, o cliente pode retornar a fase de Especificação com o objetivo de alterar as definições dos seus requisitos para atender suas expectativas, para então reiniciar o processo de Negociação. Esta fase finaliza Negativamente quando não é possível estabelecer um acordo, sendo então encerrado o processo. A finalização Positiva implica que um SLA foi acordado, desta forma, as especificações presentes no SLA, que são constituídas de um texto fixo adicionado das informações obtidas durante a especificação, juntamente com os dados do usuário serão utilizados para gerar um modelo do ambiente virtual para este cliente, sendo permitido ao cliente gerar quantos ambientes de trabalho desejar.

Os atributos de qualidade adotados por este trabalho, bem como suas especificações serão descritos posteriormente, bem como o processo de negociação. O Algoritmo 2 descreve o processo de negociação adotado pelo GerNU.

Algoritmo 2 - Processo de Negociação

```
1:  negociacao(){
2:    descontoAtual = desconto já concedido a proposta;
3:    descontoMaximo = valor definido pelo administrador da nuvem;
4:    stepDesconto = % desconto para cada proposta (definido pelo admin);
5:    if descontoMaximo <= descontoAtual + stepDesconto
6:      descontoAtual = descontoAtual + stepDesconto;
7:      valorProposta = criar_nova_proposta();
8:      exibir_nova_proposta();
9:    Else
10:     disponibilizar_negociacao_manual();
11   end IF
12: }
```

A linha 5 é utilizada para fazer a verificação sobre a possibilidade de ser realizado automaticamente uma nova proposta para o usuário. Caso o limite de desconto já tenha sido alcançado, será iniciado o processo de negociação manual através da linha 10. A partir da linha 7, temos a invocação da função que cria uma nova proposta de valores, sendo que a mesma é baseada na relevância que o usuário previamente atribuiu a cada parâmetro do seu serviço, ficando esta abordagem para ser descrita detalhadamente posteriormente.

5.2.2.3 Fase 3 - Implantação

A partir de uma conclusão positiva da fase de Negociação, um processo automatizado será iniciado para criar uma máquina virtual em conformidade com requisitos de hardware acordados no SLA e ainda considerando o sistema operacional e as aplicações especificadas, de forma a apresentar um desempenho adequado, objetivando preservar os parâmetros de qualidade do serviço.

Um modelo do ambiente virtual será utilizado para criar o serviço do cliente. Este modelo, além de permitir a verificação dos parâmetros de qualidade será utilizado para realizar atualizações no ambiente quando for necessário. A partir da criação deste modelo, definimos três perspectivas de implementação, sendo que estas já foram descritas na introdução deste capítulo, que podem ser consideradas como factíveis e que tem por objetivo disponibilizar uma imagem ou estrutura de virtualização em um formato conhecido, de forma que com sua utilização seja possível disponibilizar um serviço.

Esta é uma fase automática, que tem sua primeira etapa iniciada com um agente do sistema extraindo informações do SLA para criar o Modelo do Ambiente Virtual, sendo este utilizado como referência para as demais atividades dos demais agentes do sistema. Com o modelo de ambiente criado, inicia-se a segunda etapa desta fase que tem por objetivo efetivamente criar a máquina virtual que dará suporte ao serviço, para então disponibilizar o ambiente virtual para utilização pelo usuário. Esta segunda etapa pode ser observada na figura 5.8, onde pode se perceber quatro atividades principais: a primeira, **Criação da Máquina Virtual**,

corresponde a solicitação junto ao virtualizador de uma máquina virtual, ou seja, neste momento é criada uma estrutura básica gerando a identificação da máquina, para que posteriormente sua estrutura seja atualizada pelos requisitos do usuário.

A segunda etapa, **Configuração da Instalação**, é a atividade que dinamicamente constrói toda a estrutura necessária para a instalação dos requisitos de *hardware* e *software* do usuário e suas configurações. A **Execução da Instalação** corresponde a terceira etapa, e representa o processo efetivo de criação do serviço conforme foi especificado pelo usuário. Por fim, a quarta etapa, **Disponibilização do Serviço**, corresponde às atividades pertinentes à liberação do serviço para uso do cliente e o modelo de ambiente tem seu status atualizado para ativo.

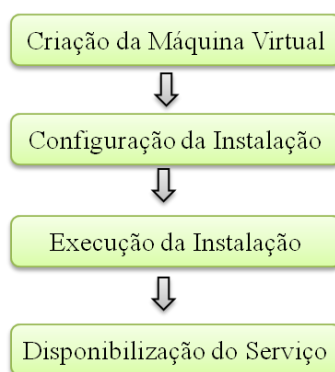


Figura 5.8: Etapas da Implantação

Os principais algoritmos utilizados nesta fase serão descritos na subseção que descreve a estrutura de software utilizada pelo GerNU, sendo importante ressaltar que todo o processo de implantação foi desenvolvido para acontecer dinamicamente, visto que no sistema não existem imagens virtuais pré-fabricadas.

5.2.2.4 Fase 4 - Monitoramento

A fase de Monitoramento, assim como a de Implantação foi desenvolvida para funcionar de maneira automatizada, tendo por objetivo garantir que ambos, cliente e provedor, sejam atendidos em conformidade com as especificações do SLA. A partir do momento que o ambiente de trabalho virtual do cliente esteja disponível, será iniciada a fase de Monitoramento, sendo que esta deve ser realizada por mecanismos disponibilizados pelo provedor da nuvem, não fazendo parte do

escopo deste trabalho as atividades de monitoramento.

Se ocorrer uma degradação na qualidade do funcionamento do ambiente virtual de trabalho, o modelo de ambiente poderá ser reconfigurado, objetivando atender as expectativas do cliente que foram acordadas no SLA. O monitoramento proposto pelo GerNU deverá ocorrer das duas maneiras descritas a seguir:

a) **Monitoramento dos atributos de qualidade:** processo onde os atributos de qualidade serão periodicamente supervisionados e armazenados, observando-se, a cada interação, se os mesmos estão em conformidade com a especificação do SLA. Com o armazenamento das medições será possível criar diversos tipos de relatórios, intencionando identificar problemas ou propensões a problemas de forma a ser possível identificar padrões que alertariam sobre a possibilidade de violação do SLA.

b) **Questionários de satisfação:** processo no qual serão enviados regularmente formulários ao usuário com a intenção de identificar o nível de satisfação do mesmo com o serviço. Além disto, podem servir como conexão para identificar problemas potenciais e possíveis serviços adicionais que poderão ser oferecidos. Acima de tudo, intenciona-se aumentar a confiança dos usuários a partir da percepção dos esforços adotados em garantir cada uma das garantias estabelecidas no SLA.

A figura 5.9 apresenta a perspectiva que descreve o cenário da aplicação proposto pelo GerNU atendendo o processo de aquisição de serviços. Pode ser observado que, a primeira atividade do processo de aquisição foi atendida pela fase de Especificação. As atividades de registro e negociação são atendidas pela fase de Negociação. Para atender as atividades de configuração dinâmica e provisionamento do serviço foi desenvolvida a fase de Implantação. Por fim, as atividades de acompanhamento são implementadas na fase de Monitoramento.

5.2.2.5 Ciclo de Vida

Como pode ser visto na figura 5.10, as quatro fases presentes no cenário proposto representam exatamente os quatro tipos de atividades presentes no ciclo de vida deste sistema, sendo também exibidos os principais artefatos produzidos por cada atividade. O artefato Detalhes do Ambiente é utilizado para armazenar as informações iniciais relativas aos requisitos de *hardware* e *software* que estarão

disponíveis na plataforma do cliente. Já o artefato Modelo do Ambiente, incorpora o objeto Detalhes do Ambiente, inclusive com possíveis alterações provenientes do processo de Negociação, além de toda especificação dos atributos de qualidade, sendo utilizado para a geração automatizada do ambiente virtual que deverá ser utilizado pelo usuário.

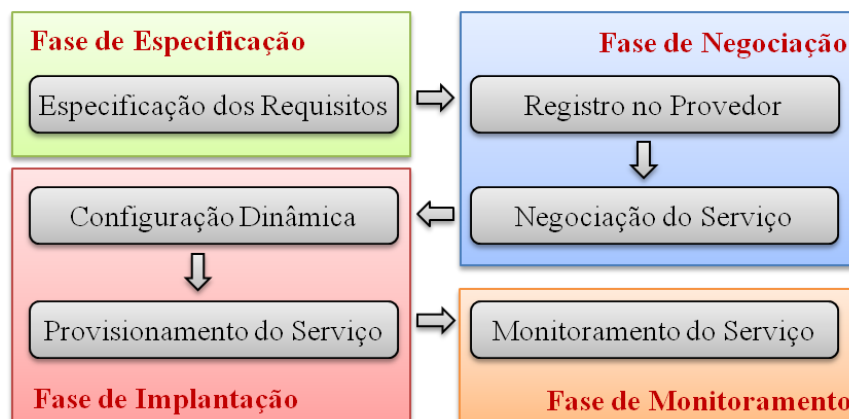


Figura 5.9: O GerNU atendendo o processo para aquisição de serviços

A atividade de implantação tem como fruto uma máquina virtual criada sobre a estrutura da nuvem e finaliza iniciando o processo de Monitoramento, que tem o objetivo de verificar se existem descumprimentos do SLA em relação ao serviço ou insatisfação por parte do cliente, caso uma destas duas situações sejam verdadeiras o modelo de ambiente deverá ser atualizado e o ambiente dinamicamente reconfigurado.

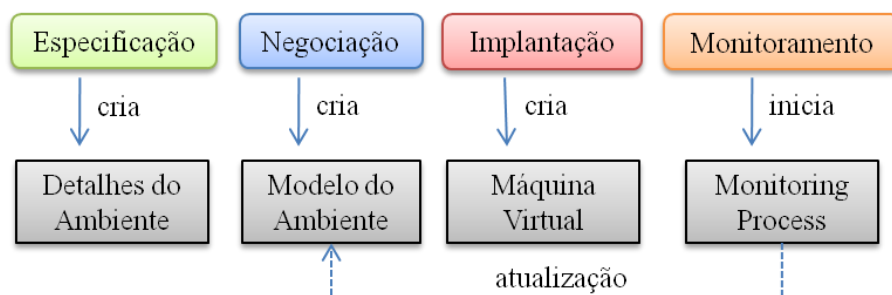


Figura 5.10: Ciclo de Vida do GerNU

5.2.3 Agentes do Sistema

No GerNU, são utilizados quatro agentes para controlar suas funcionalidades,

a saber, Negociador, Extrator, Criador e Monitor, sendo estes, responsáveis pelas principais funcionalidades do ambiente. Na figura 5.11 podem ser observadas as principais interações que ocorrem entre os agentes e com o banco de dados do sistema durante a aquisição, provisionamento e monitoramento do serviço. A seguir serão descritas as atividades de cada agente.

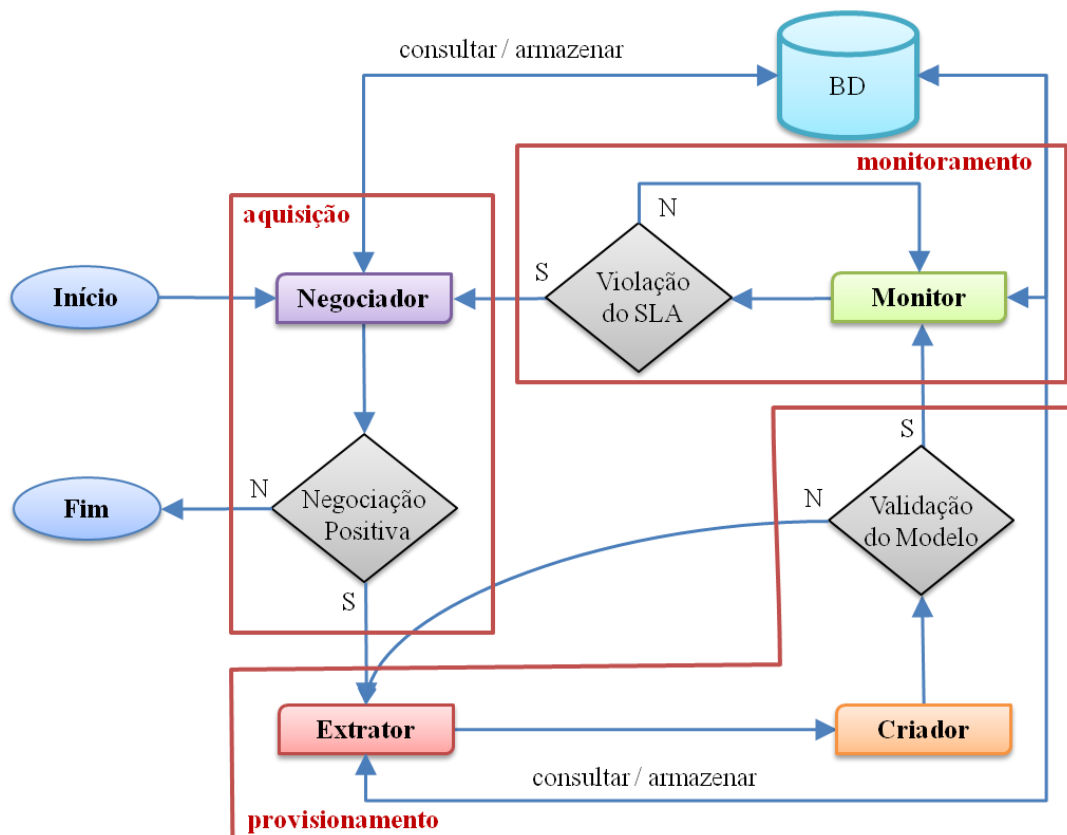


Figura 5.11: Agentes do GerNU

5.2.3.1 Agente Negociador

Este agente atua durante as fases de Especificação e Negociação do ciclo de vida do GerNU. Inicialmente, suas atividades estão relacionadas com a disponibilização de informações para os usuários. Primeiramente, estas informações são sobre os tipos de serviços (IaaS, PaaS e SaaS) que o sistema pode criar e entregar. Em seguida são disponibilizadas informações sobre os sistemas operacionais e aplicações disponíveis no repositório do sistema. Durante este processo também será necessário especificar a configuração do *hardware*, sendo que o GerNU permite uma especificação personalizada, onde o usuário define

exatamente a quantidade de processadores, memória e capacidade de armazenamento que deseja. Outra possibilidade é a utilização de uma das configurações de *hardware* padrão fornecida pelo sistema. Desta forma, é permitido ao usuário escolher a opção mais adequada, considerando, por exemplo, o desempenho desejado para seu serviço. Por fim, são disponibilizadas as informações necessárias para a especificação dos parâmetros de qualidade para os serviços. Todas estas informações correspondem a especificação do serviço e servem como dados de entrada iniciais para o SLA, configurando os valores iniciais do contrato.

O Agente Negociador também controla todo o processo de negociação do SLA sendo responsável por disponibilizar o cadastro do usuário e pela interface de negociação. Nesta interface o usuário terá ciência da composição dos custos e poderá solicitar uma nova proposta de valor. Este agente é responsável por avaliar a possibilidade de criação de uma nova proposta, especificando os novos valores para os serviços que serão apresentados. A última atividade deste agente é a criação de um SLA para a especificação que foi negociada, estabelecendo o contrato que regerá as atividades entre as partes. As atividades do Negociador podem ser encerradas por dois motivos: o primeiro, denominado Encerramento Positivo, quando as partes conseguem fazer um acordo, estabelecendo-se então um contrato, tendo então como atividade final iniciar o Agente Extrator. A segunda possibilidade, denominada Encerramento Negativo, acontece quando uma das partes desiste da negociação e nenhum acordo é efetivado, nesta ocasião as atividades do agente se encerram descartando-se as informações da negociação. A figura 5.12 exibe as principais interações do agente Negociador, que inicia disponibilizando para os usuários sua estrutura de serviços, incluindo os sistemas operacionais, as aplicações de desenvolvimento, as aplicações de produção e os atributos de qualidade, todos estes a partir de informações no repositório do sistema. Todas estes dados representam as possibilidades que foram disponibilizadas pelo administrador da nuvem e que devem ser gerenciadas pelo GerNU.

5.2.3.2 Agente Extrator

Este agente inicia suas atividades a partir do momento em que um SLA é

estabelecido. A sua responsabilidade é extrair do SLA informações que são fundamentais para o Modelo do Ambiente Virtual, portanto, a partir do SLA será criado uma estrutura que será utilizada para criar a máquina virtual que executará o serviço do cliente.

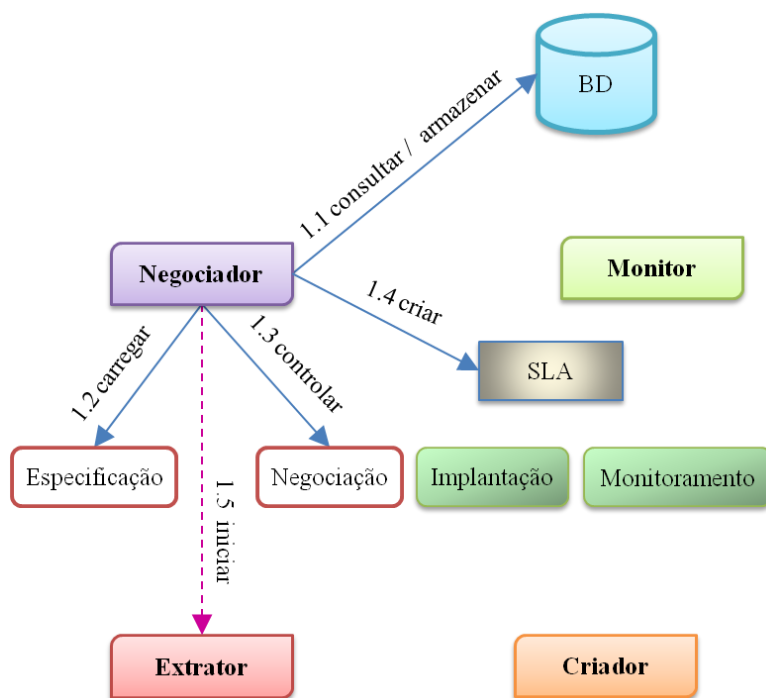


Figura 5.12: Agente Negociador

O Modelo do Ambiente contém diversas informações, dentre elas, detalhes sobre o usuário, informações a respeito da estrutura de hardware e de software, informações relacionadas com as garantias do serviço e ainda informações sobre os parâmetros de qualidade do serviço. Uma segunda maneira que o Agente Extrator pode ser iniciado é quando existe a necessidade de uma atualização no ambiente virtual que suporta o serviço. Por exemplo, quando por alguma razão se faz necessário aumentar ou diminuir os recursos da máquina virtual, este agente reconstrói o Modelo do Ambiente Virtual para que as atualizações sejam efetivamente executadas e reflitam no serviço disponibilizado para o usuário. A figura 5.13 exibe as principais interações do agente Extrator. Sendo que a partir do SLA acordado entre o cliente e o provedor é criado o Modelo do Ambiente. O Agente Extrator realiza suas atividades durante a fase de Implantação.

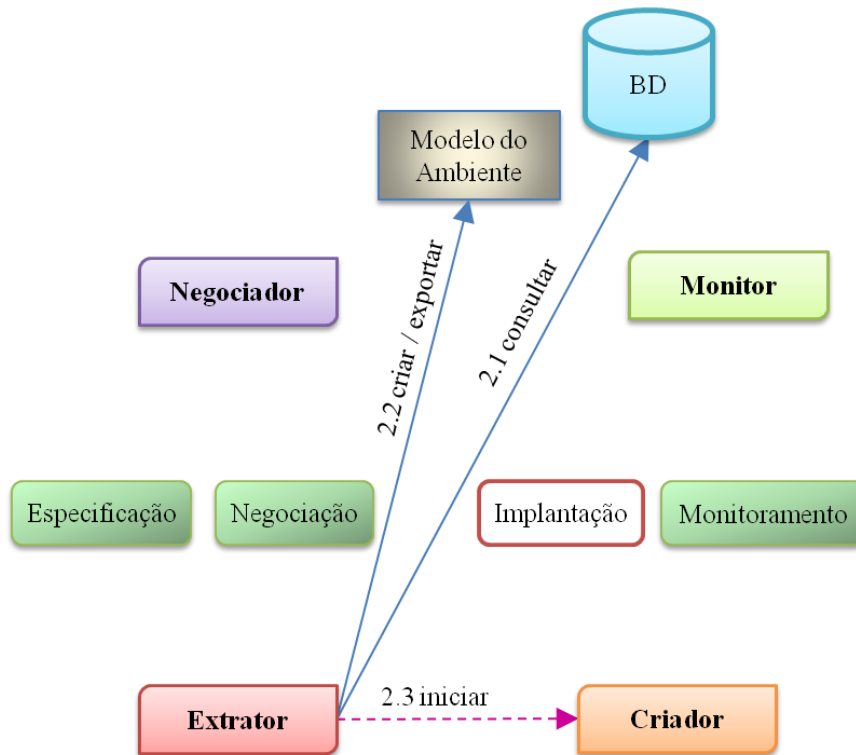


Figura 5.13: Agente Extrator

5.2.3.3 Agente Criador

A primeira tarefa deste agente é executar uma validação no Modelo do Ambiente criado pelo Agente Extrator. Após a validação, seu trabalho será direcionado por uma das três possibilidades identificadas na fase de Implantação, sendo esta especificação feita pelo administrador. Considerando a primeira possibilidade, as atividades do Agente Criador encerram com a validação do modelo, sendo a responsabilidade transferida para o provedor. No caso da segunda perspectiva, a partir do mecanismo de mapeamento o provedor receberá uma imagem virtual e será responsável por disponibilizar o serviço. Para o terceiro caso descrito na fase de Implantação, o trabalho do Agente Criador se estende a validação do sistema, devendo o mesmo criar uma *appliance* para o *software* e um *template* para o *hardware* do ambiente que será virtualizado.

Para este trabalho, uma *appliance* representa uma imagem virtual contendo o sistema operacional e as aplicações que foram requisitadas pelo usuário. O *template* de *hardware* especifica os parâmetros de *hardware*, como CPU, memória e

armazenamento, que estarão disponíveis para o ambiente.

O agente Criador constrói a *appliance* para o serviço de forma dinâmica e em tempo de execução, baseado nas informações contidas no Modelo de Ambiente, sendo também o *template* de *hardware* gerado a partir dos dados presentes no modelo. A construção da *appliance* e do *template* precisam ser direcionadas para a utilização de um virtualizador específico, sendo este utilizado pelo Agente Criador para efetivamente criar e disponibilizar o serviço para o usuário.

O virtualizador utilizado para os testes deste trabalho é o VirtualBox da Oracle, porém ressalta-se que qualquer outro poderia ser utilizado sem maiores restrições, visto que os comandos específicos do virtualizador não estão diretamente acoplados ao sistema. A figura 5.14 exibe as principais interações do agente Criador.

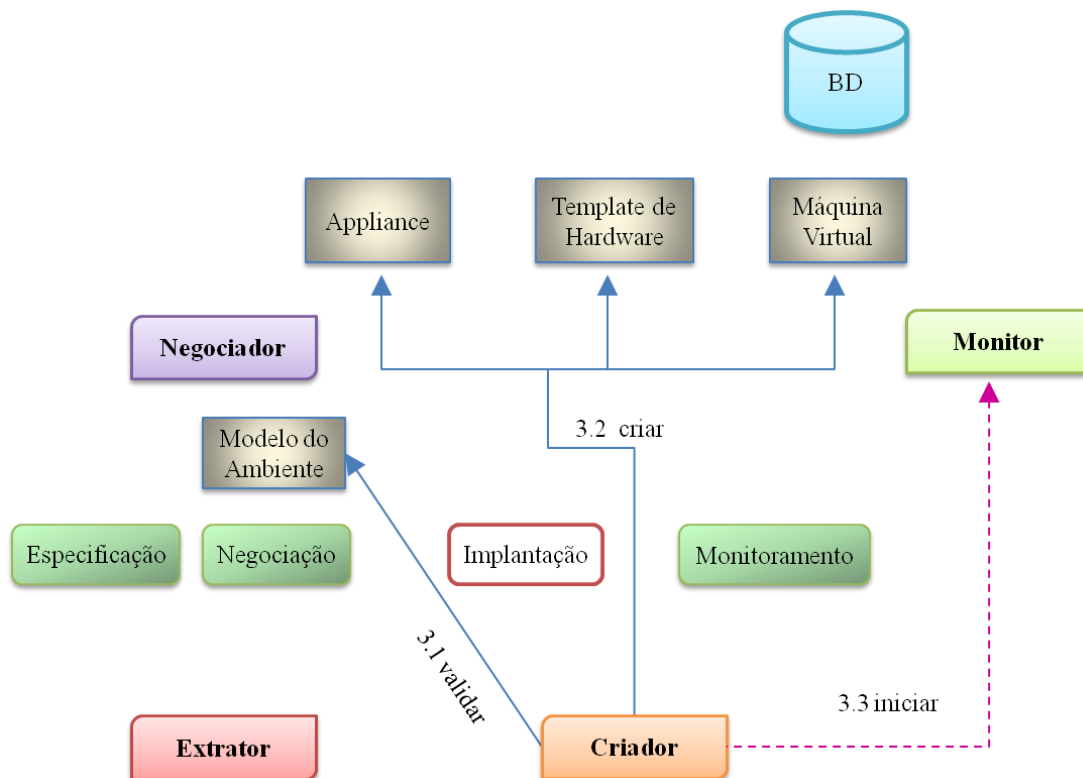


Figura 5.14: Agente Criador

Destaca-se nesta imagem a validação do modelo do ambiente virtual, realizada a partir da verificação da existência de informações mínimas para a criação de um serviço, sendo fundamental para garantir a consistência do ambiente virtual. Posteriormente são criados os artefatos de suporte para a criação da

máquina virtual. Durante a fase de Implantação é que são realizadas as atividades do Agente Criador.

5.2.3.4 Agente Monitor

As atividades do Agente Monitor iniciam quando o serviço é disponibilizado para utilização pelo usuário. A partir deste momento são monitoradas e armazenadas as informações sobre o serviço, incluindo os parâmetros de qualidade do ambiente, sendo todas elas avaliadas regularmente. Se os parâmetros de qualidade não estiverem sendo atendidos em conformidade com o estabelecido no SLA ou acontecerem sobrecargas no sistema, o Agente Monitor pode invocar o Agente Negociador para oferecer uma alternativa como a atualização do ambiente para o usuário de forma a preservar o acordo estabelecido. Neste contexto, o objetivo é os valores adotados por esta política venham claramente beneficiar o cliente, pretendendo reverter qualquer cenário de insatisfação, por exemplo, oferecendo mais recursos cobrando menos que o preço normalmente praticado.

Intencionando diminuir a comum distância e talvez indiferença entre provedor e cliente, as tarefas deste agente também envolvem um contexto interativo, sob a forma de questionários de satisfação. Sendo assim, o sistema interage com o usuário a procura de feedback sobre o seu contentamento em relação ao serviço e também com a intenção de resolver problemas rapidamente e eficientemente, além de identificar novas funcionalidades que podem interessar ao usuário. A figura 5.15 exibe as principais interações do Agente Monitor, ressaltando que ele constantemente atualiza o banco de dados, criando um histórico das avaliações do ambiente virtual, de forma a possibilitar a geração de diversificados relatórios. O Agente Monitor tem suas atividades restritas a fase de Monitoramento.

5.3 VISÃO DETALHADA DA PROPOSTA

Nesta subseção pretende-se descrever mais detalhadamente políticas e funcionalidades que são utilizadas pelo GerNU, tais como as infraestruturas de hardware e software, os parâmetros de qualidade propostos e as políticas de negociação e gestão do SLA.

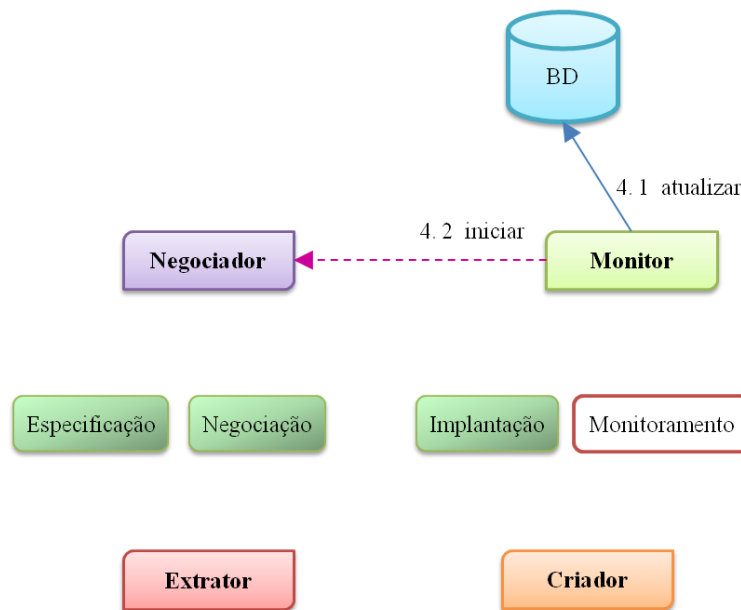


Figura 5.15: Agente Monitor

5.3.1 Abordagem Dirigida a Modelo

Após a fase de Negociação, considerando que a mesma foi positiva, ou seja, terminou com o estabelecimento de um SLA, um modelo de ambiente virtual será gerado. A partir de então, toda a infraestrutura de *hardware* e *software* da máquina virtual que disponibilizará o serviço acordado, será baseada nas informações contidas neste modelo, sendo que todas elas foram extraídas do SLA. Desta forma, as informações coletadas ao longo das fases de Especificação, como dados do usuário, sistema operacional e aplicações, e da fase de Negociação, como os atributos para os parâmetros de qualidade, valores dos serviços e das penalidades, serão incorporados ao modelo do ambiente, que a partir de então servirá como base para todas as atividades desempenhadas no sistema, incluindo atualizações, sendo estas realizadas primeiramente no modelo para então, através de agentes automatizados, realizarem automaticamente a configuração das alterações no ambiente virtual. Cada serviço criado no sistema terá seu próprio modelo de ambiente que contém todas as informações necessárias para o seu gerenciamento e para as atividades de atualização, servindo também como base para exportações futuras para outros provedores de serviço.

O modelo do ambiente virtual, como pode ser visto na figura 5.16, conterá três principais tipos de elementos, porém dependendo da necessidade e visando agregar maior completude ao contexto, outras informações podem ser adicionadas ao modelo. Todas as informações do modelo serão descritas no formato Extensible Markup Language (XML), sendo apresentado uma breve descrição a seguir.

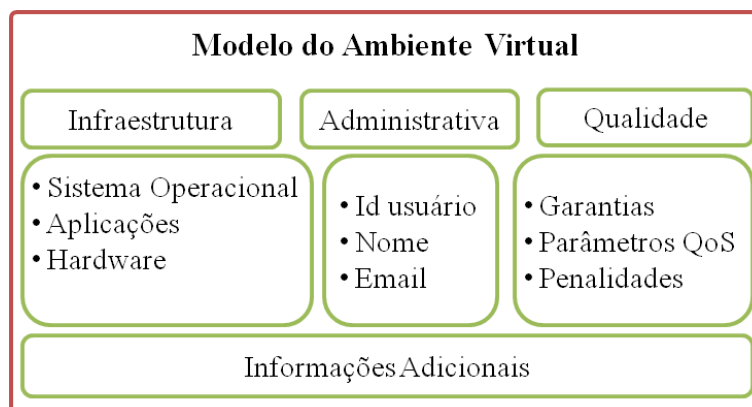


Figura 5.16: Modelo do Ambiente Virtual

1. Informações de Infraestrutura: Descrevem as informações sobre a especificação de *hardware* e *software* que estarão disponíveis no serviço. Estas informações são coletadas durante a fase de Especificação, onde o usuário seleciona suas opções para os seus requisitos. O GerNU irá permitir que um usuário verifique as possibilidades de combinações disponíveis, avaliando os custos e requisitos iniciais propostos pelo sistema, sem a necessidade de um cadastro prévio.

2. Informações Administrativas: Representam informações coletadas durante a primeira etapa da fase de Negociação e estão relacionadas com os dados pessoais do usuário. Após especificar seus requisitos o usuário que desejar efetivar a contratação do serviço e / ou negociar os valores do serviço, deverá criar uma conta para então prosseguir com o processo de aquisição centrado ambiente de trabalho.

3. Informações de Qualidade: Estas informações descrevem as garantias do SLA, apresentando os atributos para cada um dos parâmetros de qualidade dos serviços, conforme a especificação exigida pelo usuário. Estas informações são coletadas durante a segunda etapa da fase de Negociação e também representam os atributos que serão alvo da fase de Monitoramento, sendo extremamente

importantes para garantir que os termos do SLA estão sendo respeitados.

A seguir apresentamos algumas vantagens provenientes da utilização de uma abordagem baseada em modelos como esta do GerNU, que utiliza um modelo de ambiente como artefato principal, tendo as atividades do sistema acontecendo em torno deste modelo:

- a) O modelo do ambiente contém todas as informações necessárias, previstas no escopo do sistema, para a criação de ambientes virtuais, podendo ser atualizadas, de modo a incluir novas ou remover existentes a qualquer tempo.
- b) A partir do modelo do ambiente, agentes podem ser utilizados para automaticamente gerar máquinas virtuais, contemplando as necessidades dos usuários em relação a hardware (CPU, memória, armazenamento, etc.) e software (sistemas operacionais e aplicações), provendo serviços prontos para utilização.
- c) As alterações que vierem a ser necessárias serão realizadas em um único artefato, o modelo do ambiente, sendo então automaticamente refletidas para todo o ambiente.
- d) Um modelo pode ser exportado para outros formatos de imagens virtuais utilizados por diferentes provedores da computação em nuvem, desta forma, o serviço do usuário poderá ser criado em um ambiente de nuvem e portado para outro sem grandes problemas.

Um exemplo do arquivo XML que representa o modelo do ambiente poderá ser visto posteriormente na seção que descreve os casos de uso dos testes realizados para fazer a avaliação da proposta do GerNU.

5.3.3 Infraestrutura de Hardware

Nesta subseção temos o objetivo de descrever o ambiente computacional de produção e desenvolvimento que foi utilizado como suporte para o projeto GerNU, sendo que esta fase de desenvolvimento aconteceu no Laboratório de Computação Científica Distribuída (ComCiDis) pertencente ao Laboratório Nacional de Computação Científica (LNCC). Atualmente o ComCiDis e o Laboratório de

Modelagem Hemodinâmica (Hemolab) têm sob sua gerência dois conjuntos de clusters instalados e em operação.

O primeiro conjunto é composto por:

- Um cluster ALTIX XE 1300, com uma controladora e 94 nós, interfaces de rede Ethernet, 48 TB de armazenamento e 1.152 MB de memória, com um total de 1.140 núcleos.
- Um cluster ALTIX XE 1300 GPU Cluster, com 24 nós de trabalho, interfaces de rede Ethernet, 12 TB de armazenamento e 576 MB de memória, GPU Tesla M2050 -T20, com um total de 10.752 núcleos de GPU, e 144 núcleos de CPU.

O segundo conjunto é composto por:

- Um cluster BULL HPC CLUSTER SYSTEM, com uma controladora e 84 nós, interfaces de rede Ethernet Infiniband, 21 TB de armazenamento nos nós, sendo 250 GB em cada um, e um Storage Array de 12 TB, 3.060 MB memória, com um total de 1.020 núcleos, 84 nós e 1 controladora.
- Um cluster BULL HPC CLUSTER SYSTEM - GPU Cluster, composto por 8 nós de trabalho, interface de rede Ethernet Infiniband, 2 TB de armazenamento, 192 MB memória, GPU Tesla M2050-T20 com um total de 3.584 núcleos de GPU e 48 núcleos de CPU.

Os dois conjuntos de equipamentos têm como principal diferença a camada de comunicação e em função disto pode ser determinado seu tipo de utilização. O conjunto Altix trabalha sobre uma rede Ethernet e é utilizado para a criação de ambientes virtuais de trabalho, plataformas e aplicações, enquanto o conjunto BULL possui uma camada de comunicação InfiniBand, sendo utilizado para o processamento de alto desempenho pelo HemoLab, em simulações, sem o uso da camada de virtualização.

De forma específica, durante as fases de desenvolvimento e testes do GerNU, foram alocados a partir do conjunto ALTIX XE 1300, 100 núcleos de processamento, 200 MB de memória e 5 TB de armazenamento. A utilização destes

recursos tem como objetivo atender a toda demanda relativa aos servidores necessários ao ambiente, o banco de dados e para o gerenciamento de arquivos, além de prover o suporte para a nuvem privada de testes.

5.3.4 Infraestrutura de Software

Esta subseção tem como objetivo descrever as estruturas de softwares que servem como suporte para o funcionamento do GerNU, e além disto, apresenta detalhadamente o funcionamento do procedimento para criação dinâmica das máquinas virtuais, visto que no ambiente não existem imagens virtuais previamente montadas, sendo as mesmas criadas em tempo de execução de forma a atender os requisitos específicos do usuário relacionados a *hardware* e *software*.

Com a perspectiva de utilizar apenas *software* livre, tanto o sistema operacional quanto as aplicações disponibilizadas no ambiente são gratuitas. O ambiente de execução é composto por dois servidores, ambos utilizando o sistema operacional Ubuntu Server 11.10 de 64 bits.

A primeira máquina está configurada com o servidor web Apache Tomcat, o banco de dados MySQL e a aplicação do GerNU, sendo que o servidor web disponibiliza o ambiente da aplicação para utilização pública. O banco MySQL é utilizado tanto para armazenar informações de gerenciamento do GerNU quanto para armazenar as informações das atividades do usuário, como cadastro pessoal e especificação dos seus requisitos.

A linguagem de programação utilizada para o desenvolvimento foi Java, tendo sendo implementadas diversas Java Server Pages (JSP) e classes Java para controlar todas as funcionalidades através dos agentes do sistema. Esta máquina foi configurada com quatro *cores*, oito *gigabytes* de memória RAM e duzentos *gigabytes* para armazenamento.

O segundo servidor é utilizado para dar suporte ao ambiente de pré-execução das máquinas virtuais, denominado *Preboot eXecution Environment* (PXE). A configuração deste servidor permite ao ambiente do GerNU inicializar as máquinas virtuais utilizando a interface de rede, sem a dependência da disponibilidade local de

dispositivos de armazenamento como discos rígidos ou ainda um sistema operacional. Para executar suas tarefas esta máquina foi configurada com dois *cores*, quatro *gigabytes* de memória RAM e trezentos *gigabytes* para armazenamento.

Os outros recursos computacionais que inicialmente compõem a nuvem de testes que o GerNU está associado, serão utilizados para dar suporte ao processo de virtualização, sendo estes disponibilizados para suportar os serviços dos usuários. Estas máquinas também estão configuradas com o sistema operacional Ubuntu Server 11.10 e possuem instalado o virtualizador da Oracle, o Virtualbox, de forma a permitir que seus recursos sejam compartilhados.

A infraestrutura de software do GerNU e suas interações básicas podem ser observadas na figura 5.17. No Servidor Web se concentra a lógica de negócio do sistema, sendo também executadas as funcionalidades de gerenciamento do sistema e da qualidade dos serviços. O servidor Web interage com o Servidor PXE, para que este, a partir do modelo do ambiente, configure os esquemas para instalação de cada máquina virtual em conformidade com os requisitos do usuário. A máquina virtual realiza o seu primeiro boot pela rede, com o objetivo de realizar a instalação configurada pelo Servidor PXE, personalizando o serviço disponibilizado no ambiente da nuvem.

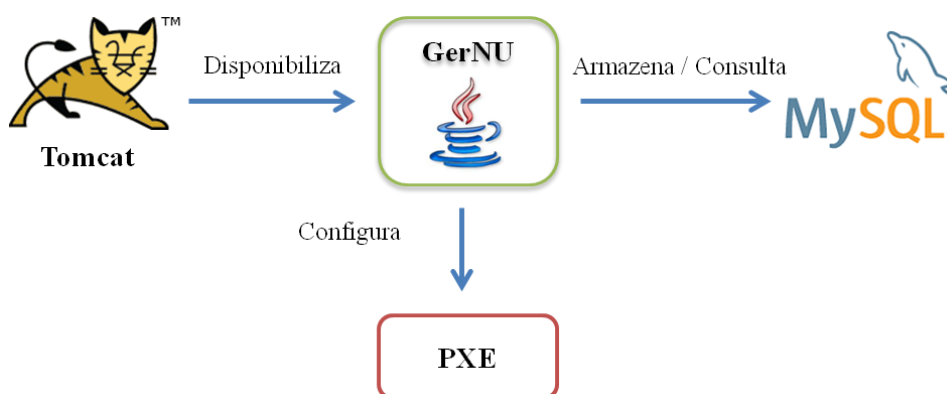


Figura 5.17: Infraestrutura de Software

5.3.4.1 Preboot eXecution Environment (PXE)

O PXE apresenta um padrão para realização de boot remoto, permitindo que

uma máquina física ou virtual realize sua inicialização e a instalação da distribuição de um sistema operacional através da rede, sendo que todos os arquivos necessários ao boot são carregados a partir de um serviço de FTP previamente configurado. Graças ao PXE é possível obter instâncias *diskless* que não dependem de unidades de armazenamento locais que contenham as imagens da instalação ou os discos de inicialização, sendo estes arquivos gerenciados pelo PXE. Este fato representa a vantagem de que os mesmos arquivos poderem ser utilizados por quaisquer umas das máquinas conectadas à rede para realização da sua inicialização.

Além de garantir que todos os computadores clientes suportem a inicialização pela interface de rede, é importante ter disponível uma conexão com a internet rápida e eficiente, porque normalmente os computadores clientes irão buscar todos os pacotes necessários para a instalação a partir de repositórios na internet. Outra possibilidade seria armazenar todos os pacotes no servidor PXE, desta forma, não se faz necessário uma conexão à internet, sendo suficiente apenas a conexão LAN com o servidor PXE, porém será preciso muito mais espaço de armazenamento no servidor PXE, além do constante gerenciamento destes arquivos, principalmente considerando que se pretende disponibilizar distintas distribuições.

Usualmente, utiliza-se um servidor PXE para fazer simultaneamente, a instalação de uma mesma distribuição de sistema operacional em diversas máquinas através da rede. Por exemplo, em uma situação onde existem 1000 máquinas que precisam ter um determinado sistema operacional instalado, em detrimento do processo manual, que consome grande quantidade de tempo, configurando-se corretamente um servidor PXE, basta conectar todas as máquinas na rede e ligá-las para automaticamente ser iniciado a execução o boot de instalação.

A figura 5.18 exibe um modelo de servidor PXE, onde a partir de sua estrutura, diversas máquinas, utilizando a rede, podem ter instalados diversos sistemas operacionais de forma simplificada, sendo exemplificado a sequência de passos com a descrição de cada um feita através da legenda.

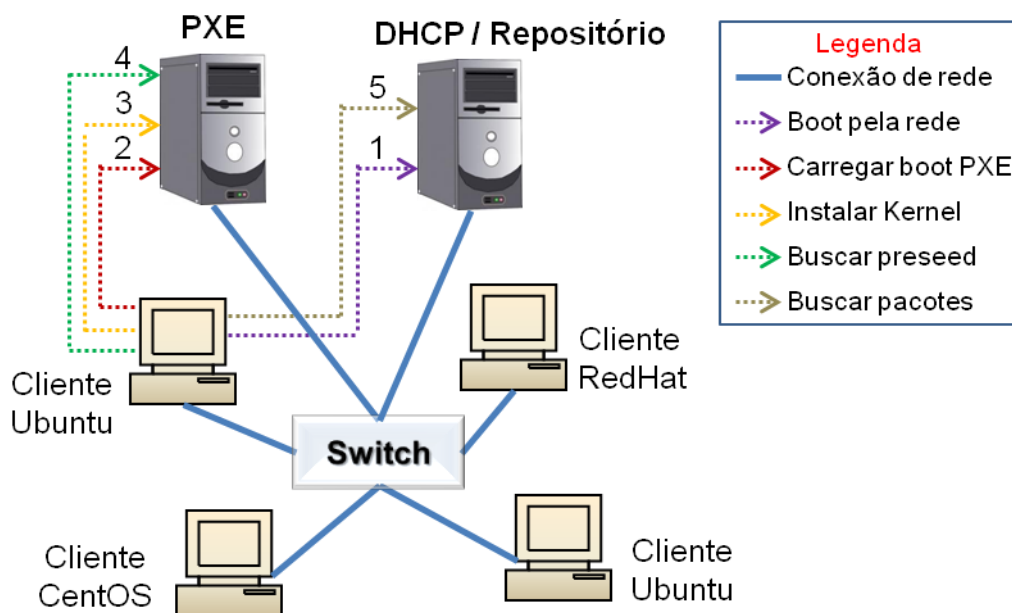


Figura 5.18: Um modelo PXE

Para o correto funcionamento do servidor PXE é necessário estar disponível um servidor para configuração dinâmica de terminais, um *Dynamic Host Configuration Protocol* (DHCP) e um servidor para transferência de arquivos, um *File Transfer Protocol* (FTP). No contexto deste trabalho, utilizamos o *dhcp3-server* como servidor DHCP e o *Trivial File Transfer Protocol* (TFTP) como servidor FTP. Este processo é iniciado quando uma máquina configurada para iniciar pela rede é ligada, inicialmente será solicitado um endereço IP ao DHCP, sendo esta solicitação atendida através da utilização do processo normal de descoberta DHCP. Depois que o cliente obtiver um endereço IP válido, será necessário localizar e estabelecer uma conexão com o servidor PXE. Após conexão ao PXE, o FTP será o protocolo de rede usado para fazer o download de todos os arquivos necessário para a inicialização de rede, incluindo a imagem de inicialização.

Dado o objetivo do GerNU, a abordagem utilizada com o servidor PXE foi modificada para atender o perfil dinâmico das atividades de uma nuvem, sendo que o mesmo é utilizado como um mecanismo de instalação personalizada, que a partir das especificações do usuário, define dinamicamente qual o sistema operacional e quais aplicações serão utilizadas para construir a imagem que será utilizada pelo virtualizador que deverá disponibilizar os serviços.

A configuração dos arquivos para o PXE, no ambiente do GerNU, se dá da seguinte maneira: primeiro, é previamente estabelecido um diretório de instalação para cada um dos sistemas operacionais disponíveis no repositório do sistema, sendo que o mesmo contém os arquivos para o boot e para a instalação. Este diretório é especificado no TFTP, que sempre verá esta pasta como a raiz para instalação de qualquer sistema operacional. Sendo assim, a partir deste diretório raiz, a estrutura do GerNU foi projetada para que de forma simplificada, seja suficiente adicionar ou remover uma pasta com os arquivos de instalação e atualizar os registros do banco de dados, para que a estrutura suporte a instalação de um novo sistema operacional. A identificação de qual subpasta será utilizada acontece a partir da seleção realizada pelo usuário no GerNU para então ser possível criar uma pasta personalizada de instalação, configurada com as aplicações especificadas pelo usuário. Após a realização da instalação, toda estrutura de arquivos que foi manipulada para configurar o ambiente de instalação será removida definitivamente do servidor. Um exemplo de como pode ficar a estrutura de arquivos pode ser verificada na figura 5.19.

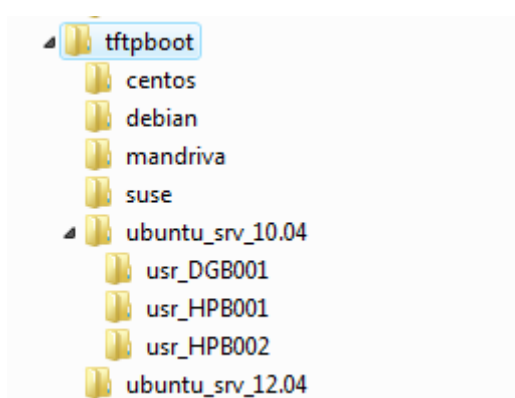


Figura 5.19: Estrutura dos arquivos

O processo para criação de uma máquina virtual pode ser observado na figura 5.20, sendo que inicialmente uma máquina virtual é solicitada ao virtualizador do ambiente, porém, após esta criação, ela representa apenas o conjunto de hardware, embora já esteja configurado com os requisitos solicitados pelo usuário, onde deverão ser instalados os softwares que o usuário especificou. Inicialmente, esta máquina virtual é obrigatoriamente configurada para realizar seu boot pela rede. Desta forma, todas as vezes que for necessário criar um novo serviço, alguns

arquivos de configuração devem ser manipulados levando em consideração as informações no Modelo do Ambiente Virtual para que os requisitos do usuário sejam atendidos. O ambiente do GerNU dispõe de alguns scripts que devem ser executados para cada novo serviço. São estes scripts que efetivamente possibilitam a inicialização do processo de instalação dos serviços, pois tem o objetivo de configurar junto ao servidor PXE toda a infraestrutura necessária para a criação do ambiente virtual através do seu endereço MAC (*Media Access Control*) e do seu endereço IP na rede. Além deste arquivo, também devem ser configurados as pastas que especificam qual sistema operacional será instalado no ambiente virtual. Estes arquivos refletem de forma única o serviço que será disponibilizada para o usuário.

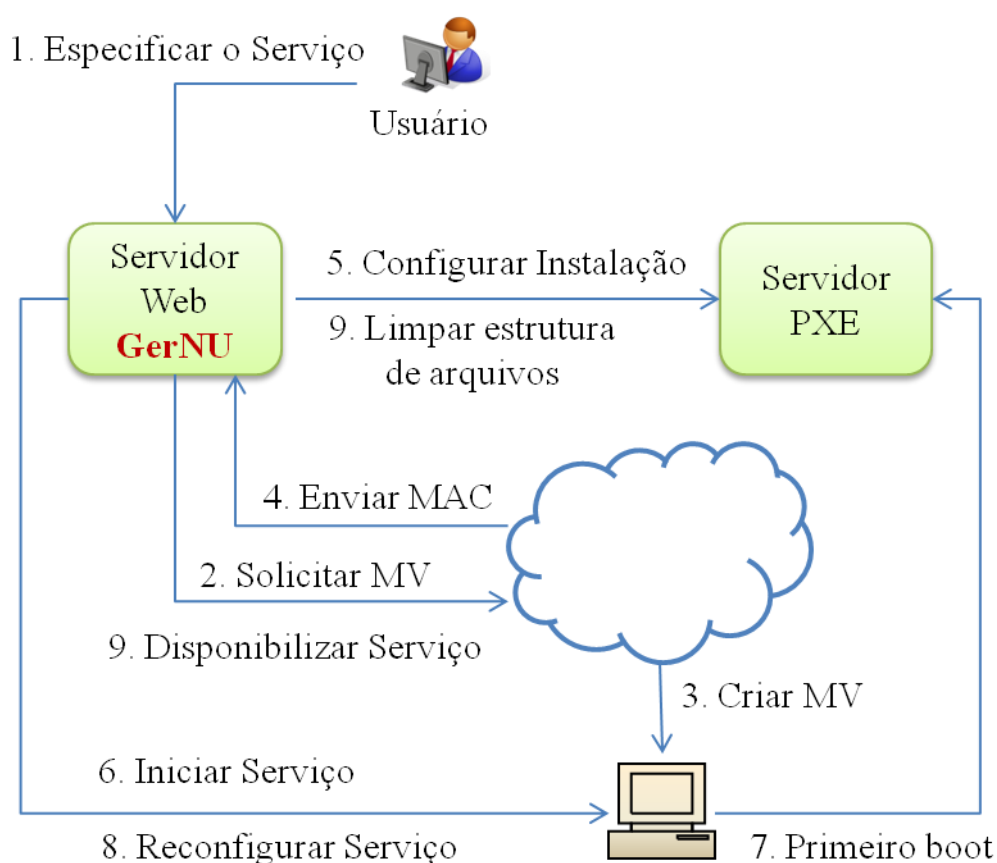


Figura 5.20: Processo para criação de uma máquina virtual

Após a realização de todo este processo, será realizado o primeiro boot da máquina virtual, devendo aguardar resposta do DHCP da rede, para que em conjunto com o servidor PXE e conforme os arquivos previamente configurados, seja iniciado o processo de instalação.

Quando a instalação estiver encerrada, a máquina virtual deve ser reconfigurada para que sua inicialização aconteça normalmente a partir de sua unidade de armazenamento local, ficando independente do servidor PXE. Por fim, todos os arquivos de configuração são descartados.

O algoritmo 3 retrata o processo utilizado pelo GerNU para criação dinâmica da máquina virtual que dará suporte aos serviços. A primeira atividade envolve o armazenamento dos dados da especificação, visto na linha 3. A linha 4 retrata o procedimento para criação da máquina virtual, sendo que este envolve comandos específicos para um virtualizador. Uma solicitação do MAC da máquina criada é feita ao virtualizador do host na linha 5. A linha 6 busca um IP disponível na faixa de IPs reservada para a aplicação. Objetivando criar todos os comandos relativos a instalação de cada uma das aplicações requisitadas pelo usuário foi escrita a linha 7, sendo que esta função concatena os comandos para instalação (previamente testados e armazenados no banco de dados do sistema) de cada uma das aplicações requeridas na especificação. Após a criação da máquina virtual, por fim, na linha 8 é chamado o algoritmo que executará a preparação de toda a infraestrutura do PXE, de forma que seja possível configurar todos os requisitos do usuário.

Algoritmo 3 - Processo de Criação dos Serviços

```
1:  criacaoServicos(){
2:      entrada: esp = dados da especificacao;
3:      salvarDadosEspecificacao (esp);
4:      criarMV (esp.nome_mv, esp.so, esp.cores, esp.mem, esp.hd);
5:      esp.mac = getMAC (esp.nome_mv);
6:      esp.IP = getIPDisponivel ();
7:      esp.InstalarApp = criarComandosInstalacaoApp(esp.apps);
8:      configPXE(esp);
9:  }
```

Já o algoritmo 4 a seguir, retrata o processo de configuração do servidor PXE para implantação dos requisitos de software. A linha 3 retrata a função que configura o arquivo utilizado pelo servidor DHCP, sendo necessário identificar o MAC e o IP do máquina virtual. A linha 4 tem o objetivo de configurar a estrutura de pastas que o

servidor de FTP utiliza. A linha 5 configura com atributos específicos do sistema operacional o arquivo de inicialização. Um script de instalação é enviado para a máquina virtual com o objetivo de atualizar os seus repositórios e efetivamente fazer a instalação das aplicações, sendo excluído após sua utilização, sendo esta atividade realizada a partir da linha 6. Após todas estas tarefas de configuração do ambiente, o serviço será iniciado pela primeira vez pela linha 7, objetivando que todos os requisitos sejam efetivamente implantados no ambiente virtual. Finalmente, a linha 8 executa o procedimento de reconfiguração do boot da máquina virtual, tornando-a independente do servidor PXE.

Algoritmo 4 - Configuração PXE

```
1:  configPXE(){
2:      entrada: esp = dados da especificacao;
3:      configDHCP (esp.nome_mv, esp.so, esp.mac, esp.ip);
4:      criarPastaBoot (esp.nome_mv, esp.so);
5:      configSeed (esp.nome_mv, esp.so);
6:      configInstalarApp (esp.InstalarApp);
7:      iniciarServico (esp.nome_mv);
8:      reconfigurarServiço (esp.nome_mv);
9:  }
```

A aplicação desta solução está limitada ao escopo de nuvens privadas, embora que, a partir das informações disponíveis no modelo do ambiente, um *broker* poderá ser empregado para viabilizar a sua utilização em um ambiente de nuvens híbridas, disponibilizando um cenário de uso com escala global. Outra utilização destas informações pode estar relacionado aos aspectos como escalabilidade, elasticidade, dentre outros, podendo cada política utilizar para beneficiar sua estratégia.

5.3.5 Atributos de Qualidade

Neste trabalho iremos utilizar o conceito de dependabilidade, bem como a definição de seus atributos, para que, de forma contextualizada em relação ao ambiente de nuvens, alguns destes atributos sejam utilizados como parâmetros de

qualidade para os serviços gerenciados pelo GerNU. Além destes, intencionando oferecer sólidas garantias aos usuários, outros atributos também poderão ser utilizados, sendo possível adicionar novos atributos a qualquer tempo a estrutura do GerNU. Desta forma, os parâmetros que já foram especificados serão descritos a seguir.

5.3.5.1 Introdução a Dependabilidade

A perspectiva de serviços baseados na internet, além dos sites de redes sociais e aplicações multimídia [Marwah et al., 2010], tem contribuído grandemente com o crescimento em número, tamanho e potência dos datacenters [Armbrust et al., 2009] [Mihailescu et al., 2011]. A facilidade de uso e de acesso a estes serviços tem criado uma dependência dos mesmos, gerando a necessidade de que os mesmos sejam altamente confiáveis [Patterson, 2002]. Para atender estes requisitos são necessários investimentos, por exemplo, o fornecimento de redundância dos serviços [Leitner et al., 2010] [Marwah et al., 2009]. Além destes custos, questões relacionadas a sustentabilidade se apresentam como desafio para o gerenciamento de datacenters [Callou et al., 2010]. Logo, a análise da dependabilidade pode fornecer mecanismos para melhorar a qualidade dos serviços, além de possibilitar o planejamento de melhores infraestruturas para datacenters, minimizando os custos e o impacto ambiental.

Neste trabalho, a dependabilidade será considerada como uma propriedade de um sistema computacional, representando uma funcionalidade, tal como usabilidade, desempenho ou custo. Essa propriedade está relacionada com a capacidade dos sistemas computacionais em prestar um serviço comprovadamente confiável, sendo que, atualmente, tem sido foco de pesquisa em diversas áreas da Ciência da Computação, devido à sua importância para o usuário e para o ambiente de execução. Além de que, dado o grande crescimento dos serviços oferecidos através da internet, a dependabilidade tem se tornado um atributo de grande interesse para o contexto do desenvolvimento de *hardware* e *software*, na implantação e operacionalização dos serviços [Maciel et al., 2010].

Em [Laprie et al., 1992], encontram-se as definições iniciais dos conceitos de

dependabilidade, sendo desde então aplicados e largamente utilizados, sendo este livro, *Dependability: Basic Concepts and Terminology*, reconhecido como precursor e grande impulsionador deste tema. [Avizienis et al., 2001] afirma que sistemas dependáveis são aqueles que são analisados a partir das métricas de dependabilidade. Uma especificação funcional de um sistema descreve o objetivo de um serviço, o motivo dele ter sido criado. Um serviço é considerado correto quando implementa sua especificação funcional. A falha do sistema representa um evento que ocorre quando a entrega do serviço não ocorre corretamente. A dependabilidade envolve várias métricas, e segundo [Laprie et al., 1992], seu conceito é composto por três partes: ameaças, atributos e meios para alcançá-lá, conforme pode ser observado na figura 5.18, que é uma adaptação da proposta original.

A dependabilidade também pode ser definida, segundo a *International Federation for Information Processing* (IFIP) ou Federação Internacional para Processamento de Informação, em [Basili et al., 2004], como a confiabilidade de um sistema computacional, no qual a confiança está baseada nos serviços que oferece.

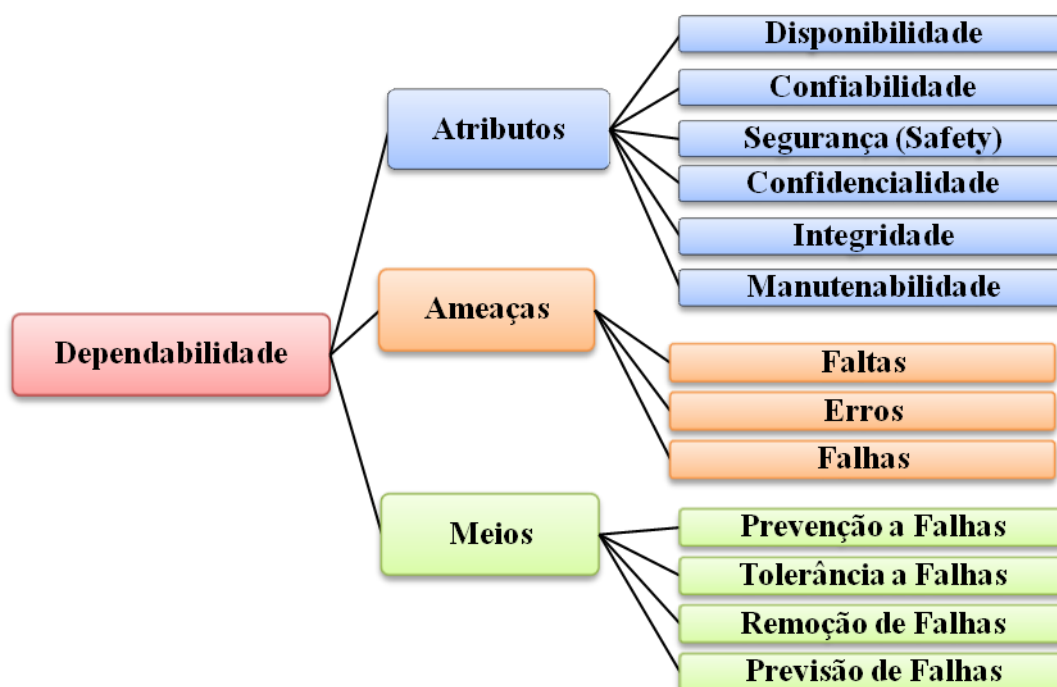


Figura 5.18: Arvore da Dependabilidade, adaptado de [Laprie et al., 1992]

As características intrínsecas de uma infraestrutura de nuvem refletem

prontamente na qualidade dos serviços disponibilizados. Desta forma, considerando a relevância que o conceito de dependabilidade tem alcançado na literatura, observa-se que os tipos de componentes que estão sendo utilizados, sua quantidade e qualidade, além da maneira como são utilizados, afetam diretamente a dependabilidade do ambiente. Assim sendo, considera-se pertinente a adequação dos atributos da dependabilidade para serem utilizados como meio para avaliar a qualidade dos serviços entregues em um ambiente de nuvem, sendo então aplicados como no GerNU.

5.3.5.2 Atributos da Dependabilidade

Dado a importância do tema dependabilidade, existem algumas ferramentas que se propõe a fazer a avaliação da dependabilidade em sistemas computacionais, porém neste trabalho, serão utilizados os atributos da dependabilidade individualmente. Desta maneira, pretende-se que os serviços sejam monitorados utilizando-se parâmetros definidos a partir destes atributos, sem contudo haver a necessidade de um cálculo efetivo para identificar valores para a dependabilidade propriamente dita. Desta forma, a seguir serão descritos os atributos da dependabilidade que serão utilizados neste trabalho, bem como identificado a forma de computar seus valores.

a) Confiabilidade

Este atributo está relacionado com uma contínua e correta entrega de um serviço. A confiabilidade de um sistema tem se tornando um fator de grande relevância para que um produto tenha boa aceitação entre os usuários. A confiabilidade de um item ou sistema corresponde a probabilidade de que os mesmos desempenhem satisfatoriamente o seu objetivo por um dado período de tempo até que ocorra uma primeira falha [Leemis, 1995] [Maciel et al., 2010] [Xie et al., 2004]. Ou seja, é a probabilidade de que o sistema não falhará por um tempo específico [Kuo and Zuo, 2003] [Rausand e Hoyland, 2004]. Por exemplo, uma confiabilidade de 90% para 1000 horas de atividade, significa que existem 90% de chances que o sistema não falhe em 1000 horas de operação.

Para uma correta identificação da confiabilidade de um sistema, [Ebeling,

1997] afirma que três aspectos devem ser considerados:

1. Uma definição não ambígua das possíveis falhas do sistema;
2. Identificação de uma unidade de tempo;
3. Observar o sistema em condições normais, sujeito a condições físicas reais, para evitar a geração de confiabilidade viciada.

Para descrever a confiabilidade de um dado sistema, é necessário conhecer a sua configuração, o estado em que ele é definido como operacional e suas regras de operação [Kuo and Zuo, 2003].

A definição matemática do atributo de confiabilidade, também denominada de função de confiabilidade $R(t)$, é a probabilidade de que o sistema irá funcionar corretamente sem falhar no intervalo de tempo de 0 a t [Maciel et al., 2010]. Esta função pode ser expressa conforme a equação 5.1, proposta por [Ebeling, 1997]. Nesta função, a confiabilidade é medida no intervalo $[0,t]$ sendo que a probabilidade $P(T \geq t)$ refere-se à chance de um sistema, decorrido o intervalo de tempo $[0,t]$, não apresentar nenhuma falha, ou seja, para um intervalo de tempo nulo, o sistema não apresentará falha, porém considerando um intervalo de tempo muito grande, também será nula a chance de um sistema não falhar.

$$\text{Confiabilidade} = R(t) = P(T \geq t), t > 0 \quad (5.1)$$

Outra função importante no estudo de confiabilidade é a probabilidade de falha $F(t)$, que nada mais é do que o complemento de $R(t)$, ou seja, descreve a probabilidade de um sistema falhar, decorrido um intervalo de tempo $[0,t]$.

$$\text{ProbabilidadeFalha} = F(t) = 1 - R(t) = P(T \leq t) \quad (5.2)$$

Um parâmetro bastante importante do estudo de confiabilidade é o tempo médio entre falhas, mais conhecido na literatura como *Mean Time Between Failures* (MTBF). Esta métrica é adequada para especificar quanto tempo um serviço funciona adequadamente antes de ocorrer uma falha, ou seja, expressa o tempo médio entre interrupções do serviço. Por exemplo, se for estipulado que o MTBF para o serviço X é de 1200 horas, significa que o serviço não deve falhar mais que uma vez a cada 1200 horas.

$$MTBF = \frac{\text{Tempo disponível}}{\text{quantidade de falhas}} \quad (5.3)$$

Outro parâmetro relevante é o tempo médio para reparo, comumente denominado *Mean Time To Repair* (MTTR), que se refere ao tempo médio gasto para que um serviço volte a normalidade de funcionamento. Não é suficiente apenas realizar um reparo rapidamente, principalmente se interrupções do serviço acontecem frequentemente, desta forma, associando-se este atributo ao MTBF pode-se auferir a eficiência dos reparos.

$$MTTR = \frac{\text{Tempo indisponível}}{\text{quantidade de falhas}} \quad (5.4)$$

b) Manutenibilidade

Quando um serviço deixa de funcionar corretamente, normalmente são realizadas atividades de reparo para corrigir a falha. Estas atividades têm como objetivo a recuperação do estado operacional do serviço, podendo ser realizados apenas ajustes ou até mesmo a substituição de componentes de *hardware* e *software*. Considerando esta perspectiva, manutenibilidade é definida em [Xie et al., 2004] como a probabilidade que um serviço ou sistema com falhas pode ser restaurado a seu estado operacional após um dado período de tempo. Assim, manutenibilidade pode ser entendido como a probabilidade de se isolar e reparar uma falha em um serviço dentro de um determinado momento.

Para definir matematicamente a manutenibilidade, [Ebeling, 1997] propõe a definição de uma variável aleatória contínua, T , especificada como o tempo necessário para reparar um serviço que tenha apresentado falhas anteriormente. Esta proposta pode ser verificada na função 5.5. Considerando um intervalo de tempo $[0,1]$ em uma unidade de tempo arbitrária, $M(t)$ é a probabilidade de um reparo genérico ser executado, representando a recuperação dos serviços após uma falha.

$$\text{Manutenibilidade}(t) = P(T \geq t) \quad (5.5)$$

c) Disponibilidade

Este outro importante atributo da dependabilidade usa diretamente os conceitos de confiabilidade e manutenibilidade e pode ser instantânea ou

estacionária. Segundo [Sahner et al., 1996], disponibilidade é a probabilidade de um serviço, sistema ou componente realizar uma função requerida em certo instante de tempo ou em um período de tempo especificado. A disponibilidade instantânea representa a probabilidade de que o sistema não falhe em um determinado tempo [Xie et al., 2004], e pode ser definida como o tempo em que um serviço funciona adequadamente para os usuários [Kuo and Zuo, 2003], ou a prontidão de um sistema para entregá-la corretamente um serviço em um dado momento [Stapelberg, 2008] [Avizienis et al., 2001].

A disponibilidade instantânea pode ser matematicamente representada pela função 5.6.

$$\text{DispInstantanea}(t) = P(\text{Sistema está operacional no tempo } t) \quad (5.6)$$

A disponibilidade em estado estacionário [Stapelberg, 2008] pode ser entendida como a porcentagem de tempo em que um componente ou sistema pode desempenhar suas funcionalidades [Wiboonrat, 2008]. Em outras palavras, é a probabilidade de um sistema estar funcionando em qualquer momento.

A disponibilidade em estado estacionário é dada pela função 5.7:

$$\text{Disponibilidade} = \frac{\text{MTBF}}{(\text{MTBF} + \text{MTTR})} \quad (5.7)$$

Exemplificando, podemos considerar um serviço monitorado por um intervalo de tempo de 10 horas. Caso o serviço em questão funcione corretamente por 5 horas (*uptime*) e nas outras 5 horas esteja em manutenção, a disponibilidade do sistema pode ser identificada como de 50%.

d) Segurança (Safety)

Segundo [Sun et. al, 2010], este atributo, pode ser definido como a característica de um sistema ou serviço em não falhar de uma maneira que cause danos catastróficos durante um determinado período de tempo. Este atributo pode ser matematicamente definido através da função 5.8, onde S é um conjunto de danos catastróficos no intervalo $[0, t]$ e $p(i)$ é a probabilidade de um estado estável no estado i .

$$\text{Segurança}(t) = \sum_{i \in S} p(i) \quad (5.8)$$

e) Integridade

Este atributo é definido como uma forma de proteção contra modificações ou destruições indevidas das informações, incluindo garantias de não repúdio e autenticidade da informação em um serviço [Sun et. al, 2010]. A integridade pode ser matematicamente definida pela função 5.9, onde S corresponde a um conjunto de modificações ou destruições impróprias no intervalo de tempo $[0, t]$, sendo que $p(i)$ representa a probabilidade do estado i ser um estado integro .

$$\text{Integridade}(t) = \sum_{i \notin S} p(i) \quad (5.9)$$

5.3.5.2 Atributos de QoS do GerNU

A proposta desta subseção tem o objetivo de especificar parâmetros para qualidade de serviços que sejam relevantes ao cenário de nuvens. O propósito inicial é identificar métricas que possam ser efetivas na qualificação de um serviço. Como dito anteriormente, o contexto de dependabilidade inspirou a definição de alguns dos atributos de qualidade propostos pelo GerNU. Porém, dado as características de um ambiente de nuvens, se faz necessário que os conceitos sejam contextualizados, desta forma, a seguir serão descritos os parâmetros inicialmente abordados pelo GerNU, sendo este um esforço com o objetivo de permitir um monitoramento do ambiente e uma qualificação do serviço mais eficiente. Outros parâmetros considerados relevantes podem ser avaliados e inseridos na estrutura da aplicação a qualquer tempo. Cada provedor de nuvem deve identificar corretamente quais parâmetros e seus respectivos atributos sua estrutura e mecanismos pode garantir, sendo de sua responsabilidade a provisão dos mecanismos para viabilizar cada um dos tipos ofertados.

Um aspecto importante durante a especificação de um serviço se refere ao custo, pois de forma geral, em relação a qualquer atributo de QoS, quanto maior for a exigência, maiores serão as taxas cobradas pelo serviço, influenciando diretamente no custo final.

a) Disponibilidade

Este parâmetro envolve os conceitos de confiabilidade e manutenibilidade,

sendo relacionado ao tempo em que o serviço estará disponível, pronto para utilização do cliente. As definições para este atributo são baseadas nas referências descritas na tabela 5.1. Um aspecto importante se refere ao custo, quanto maior a garantia de disponibilidade maior serão as taxas aplicadas. De forma geral, em relação a qualquer atributo de QoS, quanto maior for a exigência, maiores taxas serão cobradas pelo serviço, influenciando diretamente no custo final.

Referência	Atributo	Descrição
0	Não Adotado	Este é o tipo mais econômico. Não existem garantias quanto a disponibilidade. Se um serviço for interrompido, obviamente ele deverá ser restaurado, porém não haverá sanções e o tempo é dependente do provedor.
1	Sem Política de Reembolso	Serão definidos dois parâmetros, um para identificar o período de avaliação e o outro representando um valor máximo em horas que o serviço poderá ficar indisponível. Pode-se, por exemplo, ser especificado que a cada 1000 horas o serviço poderá ficar indisponível no máximo 2 horas. Alternativamente pode ser utilizado um mecanismo que considere um percentual ao contrário de um período fixo. Neste tipo é garantido o tempo máximo de indisponibilidade e no caso de ocorrer, nenhum valor será ressarcido.
2	Com Política de Reembolso	Difere do tipo Sem Política de Reembolso, exclusivamente na aplicação de penalidades para o caso de ocorrer alguma indisponibilidade.
3	Downtime Zero	Nesta modalidade, o provedor garante que o serviço estará sempre disponível. Obviamente políticas de redundância devem ser aplicadas para que se cumpra este tipo e no caso do serviço ser interrompido implicará no pagamento de multas mais elevadas.

Tabela 5.1: Disponibilidade

b) Política de Backup

O objetivo deste parâmetro abrange a segurança (*safety*) e integridade a que se refere a dependabilidade. Desta forma, este atributo apresenta garantias contra intrusão, catástrofes, modificações indevidas. De forma geral, o objetivo está relacionado com garantir que as informações do usuário serão preservadas, tanto relacionadas a sua infraestrutura quanto a seus dados de negócio. A tabela 5.2 descreve as possibilidades iniciais que o GerNU utiliza. O provedor tem a responsabilidade de identificar quais tipos de segurança podem ser oferecidos,

obviamente considerando quais deles o provedor pode oferecer.

c) Monitoramento do Ambiente Virtual

Este atributo relaciona-se com a identificação dos níveis da carga de trabalho das máquinas virtuais que dão suporte aos serviços. Este monitoramento tem o objetivo de mapear o nível de utilização dos recursos alocados para o cliente.

O ponto de vista observado é, mesmo que não existam violações do SLA, podem estar ocorrendo sobrecargas ou desperdício de recursos, fatos estes que são ruins tanto para o cliente quanto para o provedor, além de poderem onerar desnecessariamente o serviço ou até mesmo não atender as requisições com o desempenho desejado.

Desta forma, por exemplo, partindo do pressuposto que um cliente deseja a versatilidade de sempre manter a qualidade do seu serviço com o menor custo, e que a utilização do mesmo sofre picos e pode continuamente aumentar ou diminuir, este atributo sempre informará sobre este acréscimo ou decréscimo da utilização do serviço.

Referência	Atributo	Descrição
0	Não Adotado	Este é o tipo mais econômico. Não é aplicado nenhuma ação para proteção das informações.
1	Sem Acesso	O provedor garante o backup, porém não disponibiliza mecanismos para o usuário acompanhar este processo.
2	Com Acesso	O provedor realiza o backup e disponibiliza para o cliente mecanismos que possibilitem o mesmo auditar estas informações.
3	Domínio do usuário	Neste tipo, o provedor disponibiliza os mecanismos, como por exemplo, uma segunda máquina virtual, porém o backup é orquestrado e de responsabilidade do usuário.

Tabela 5.2: Políticas de Backup

A partir desta informação será possível realizar uma reconfiguração no serviço, para utilização de mais ou menos recursos de forma que a não se degrade a utilização do serviço e o valor do serviço seja compatível com cada configuração utilizada em cada período. A tabela 5.3 descreve as possibilidades iniciais para monitoramento do ambiente virtual.

Referência	Atributo	Descrição
0	Não Adotado	Este é o tipo mais econômico. Não existem atividades para monitorar e alertar o cliente sobre o nível de utilização do serviço.
1	Monitorado	A utilização dos recursos contratados é monitorada e periodicamente é avaliado se novos recursos são necessários ou mesmo se é possível atender a contento com menos recursos. Estas informações servem de subsídio para o cliente configurar de forma eficiente seu serviço.

Tabela 5.3: Monitoramento do Ambiente Virtual

d) Variabilidade do Desempenho

Através do uso da virtualização e do compartilhamento de recursos, o paradigma das nuvens pode ser visto como um único conjunto de recursos físicos para uma grande quantidade de usuários com necessidades distintas. Segundo [Iosup et al., 2010], inicialmente as nuvens foram construídas para suportar cargas de trabalho menores que as tipicamente encontradas em tarefas da computação científica, assim, esta prática pode apresentar perdas significativas no desempenho.

Uma suposição geralmente aceita, sobre a variabilidade de desempenho, considerada que o desempenho de uma máquina virtual será o mesmo, independentemente da máquina física em que ela estiver alocada. Esta é uma suposição razoável para um ambiente homogêneo, onde as máquinas físicas são idênticas e as máquinas virtuais estão executando o mesmo sistema operacional e aplicativos. Entretanto, em um ambiente de computação em nuvem, espera-se compartilhar um conjunto composto por recursos heterogêneos, onde as máquinas físicas podem variar em termos das capacidades de seus recursos e da afinidade dos dados [Rego, 2012]. Estendendo esta percepção, o fato de diversas máquinas virtuais alocadas em uma máquina física, a carga de trabalho individual de cada máquina virtual contribui diretamente para a variabilidade de desempenho da máquina física. De forma objetiva, a variabilidade de desempenho está presente em ambiente de nuvens e pode trazer sérios prejuízos aos clientes devido sua alteração, por exemplo, do tempo de resposta de uma aplicação, que pode culminar com a geração de resultados equivocados. A tabela 5.4 formata os parâmetros pelos quais a variabilidade deverá ser oferecida como requisito de QoS no GerNU.

Referência	Atributo	Descrição
0	Não adotado	Tipo mais econômico. Não é apresentado nenhuma garantia para o caso de ocorrer variação no desempenho do ambiente.
1	76 - 100 %	Taxa máxima de variabilidade apresentada pelo ambiente virtualizado estará dentro da faixa especificada no parâmetro.
2	51 - 75 %	Taxa máxima de variabilidade apresentada pelo ambiente virtualizado estará dentro da faixa especificada no parâmetro.
3	26 - 50 %	Taxa máxima de variabilidade apresentada pelo ambiente virtualizado estará dentro da faixa especificada no parâmetro.

Tabela 5.4: Variabilidade

5.3.6 Processo de Negociação

A metodologia de negociação adotada no GerNU tem o objetivo de ser simples e intuitiva, sendo o mais automatizada possível, de forma a diminuir ao mínimo a intervenção humana, acelerando o processo para conclusão da negociação e diminuindo a possibilidade de erros. Dado este propósito, uma série de atributos de negociação são previamente configurados pelo administrador da nuvem no GerNU, estabelecendo-se limites que determinam os parâmetros para que o sistema crie propostas alternativas, de forma a estabelecer um processo de negociação automatizado. Para as situações que excedam os limites especificados, um administrador humano será acionado para intervir no processo e a tomar a decisão. A fase de negociação acontece no GerNU em dois níveis. Primeiramente acontece uma negociação implícita, onde os requisitos do serviço são especificados, para então ser iniciada uma negociação explícita que se relaciona com o valor do serviço. A figura 5.19 representa uma proposta detalhada, feita a partir da especificação do usuário e demonstrando os valores que compõem o custo do serviço, que deverá ser apreciada pelo usuário, para que se dê início a uma negociação explícita.

5.3.6.1 Negociação Implícita

Este processo de negociação acontece durante a fase de especificação. O

usuário negocia, no sentido de escolher entre várias opções ou explicitamente com a possibilidade de especificar suas necessidades em nível de *software*, *hardware* e parâmetros de qualidade. A negociação dos requisitos de software acontece com a flexibilidade que o GerNU dá ao usuário em relação a possibilidade de especificar, tanto o sistema operacional quanto quais aplicações deseja que estejam disponíveis em seu serviço, de forma a criar um ambiente de trabalho personalizado.

Os requisitos de *hardware* para os serviços, como quantidade de cores, memória, armazenamento e outros, podem ser negociados pelo usuário no GerNU através da especificação de exatamente qual a quantidade de cada um destes itens ele deseja contratar, de forma a atender as necessidades computacionais específicas relacionadas ao perfil do seu serviço. Além desta possibilidade de personalização, são oferecidas três configurações padronizadas, que podem ser observadas na tabela 5.5.

Software		
Tipo	Descrição	
Sistema Operacional	Ubuntu Server 11.10	
Aplicações	MySQL 5.0	
	NetBeans 7.2	
	Apache 2.0	
Hardware		
Tipo	Qtd	Valor (R\$)
CPU (UND)	2	0,0035
Memória (GB)	4	0,0020
Armazenamento (GB)	600	0,0015
Qualidade do Serviço		
Parâmetro	Opção	Valor (R\$)
Disponibilidade	70%	0,030
Política de Backup	Sem Acesso	0,010
Monitoramento	Não Adotado	0,000
Variabilidade	51-75%	0,020

Valor do Serviço (R\$)

0,067 / hora

Figura 5.19: Proposta Inicial

A **configuração mínima** tem o objetivo de atender usuários que desejam apenas executar serviços com tarefas simples. Quando o serviço precisa atender tarefas mais complexas, que exigem um pouco mais de desempenho pode ser

utilizado a **configuração flexível**. Por fim, a **Configuração robusta** pretende atender os serviços com maior necessidade de desempenho ou com grande manipulação de dados.

Configuração	Cores (Unidade)	Memória (GB)	Armazenamento (GB)
Mínima	1	1	200
Flexível	2	2	400
Robusto	4	4	600

Tabela 5.5: Configurações de hardware – Padrão do GerNU

A criação destas configurações foi inspirada a partir dos modelos utilizados pela Amazon, ditas adequadas para a maioria dos aplicativos, podendo ser observados na tabela 5.6. Obviamente, as especificações de hardware utilizados no GerNU foram adequados para a realidade dos recursos computacionais disponíveis no ambiente de testes.

Configuração	Cores (Unidade)	Memória (GB)	Armazenamento (GB)
Instância Pequena	1 EC2	1,7 GB	160
Instância Média	2 EC2	3.75	410
Instância Grande	4 EC2	7,5	850
Instância Extragrande	8 EC2	15	1690

Tabela 5.6: Configurações de hardware - Padrão da Amazon

Os atributos de qualidade adotados pelo GerNU também são negociados através da seleção de quais parâmetros o usuário considera adequado para o serviço que está contratando, devendo ser priorizado o estabelecimento de um contexto adequado a natureza da proposta do serviço e como o cliente precisa que o serviço se comporte em relação a desempenho e disponibilidade, por exemplo.

5.3.6.2 Negociação Explícita

A fase de negociação explícita é iniciada quando o cliente acaba de fazer toda a especificação do seu serviço, sendo que durante a especificação, dinamicamente, um valor para o serviço é calculado e exibido, variando com as opções selecionadas pelo usuário. O valor do serviço é calculado a partir dos valores especificados pelo administrador da nuvem para cada atributo do serviço, incluindo os parâmetros de qualidade. Esta fase explícita inicia com uma proposta do sistema através de uma

interface gráfica que tem o propósito de permitir a negociação dos valores, formalizando-se uma clara relação entre a especificação do usuário com os custos do serviço, sendo exibido de forma detalhada a composição do custo. Desta forma, o usuário pode fazer uma avaliação objetiva do custo, levando em consideração os detalhes expostos e o objetivo que deseja alcançar com o serviço, podendo desta forma, alterar sua especificação original considerando sua necessidade e os valores disponibilizados.

A política de propostas automáticas adotadas pelo GerNU é baseada no *Analytic Hierarchy Process* (AHP), que tem como fundamento a decomposição e síntese das relações entre os critérios de um objetivo geral, de forma a ser possível priorizar indicadores, objetivando aproximar-se da melhor resposta para um problema de decisão [Saaty, 1986]. Ao analisar-se os elementos hierárquicos de um problema, considera-se como o peso dos fatores individuais que compõem o problema, influenciam o seu objetivo geral. Desta forma, normalmente o AHP é utilizado em problemas cuja relação da influência dos fatores não é uniforme, possibilitando a identificação de prioridades. Estas prioridades representam os pesos relativos que destacam as diferenças entre os critérios. Com a determinação das prioridades para os fatores primários, o julgamento sobre a melhor opção reduz-se a uma sequência de comparação por pares, capturando-se medidas subjetivas e objetivas que esclarecem o domínio de um critério sobre outro ou de uma opção em detrimento das demais.

Uma série de trabalhos justificam a adoção do AHP, visto sua aplicabilidade já ter sido comprovada, tanto academicamente quanto no escopo de negócios. Em [Grandzol, 2005] é apresentado um projeto para seleção de corpo docente; a identificação de prioridades para pesquisa na agricultura é descritas em [Becker and Braunschweig, 2004]; a medição da eficiência do gerenciamento de atividades de pesquisa e desenvolvimento em universidades foi realizada em [Feng et al., 2004]; Em [Murakami and Almeida, 2003] é apresentado uma aplicação para decisão estratégica de tecnologia da informação; uma avaliação de riscos em projetos relacionados com Enterprise Resource Planning (ERP) é feita em [Huang et al., 2004]; a identificação de prioridades em um sistema de gestão de segurança é

destacada em [Chan et al., 2004]. Todos estes trabalhos aplicaram o AHP para viabilizar as soluções adotadas.

A proposta da metodologia do AHP é dividir o problema geral em avaliações mais simples, mantendo a participação desses problemas menores na decisão global. Em outras palavras, considera que para resolver um problema complexo, será mais conveniente dividi-lo em subproblemas, que serão solucionados individualmente e depois somados, representando a decisão do problema original.

Uma hierarquia (detalhamento do problema) bem construída representa um modelo da realidade, gerando uma clara visão geral do sistema. Além disto, modelos hierárquicos são estáveis (visto que, pequenas modificações implicam também efeitos pequenos) e flexíveis (porque adições a uma hierarquia bem estruturada não perturbam o desempenho do sistema). Entretanto, uma hierarquia apenas não representa uma ferramenta poderosa no processo de tomada de decisões ou de planejamento, sendo necessário avaliar a relevância dos elementos de um nível em relação aos elementos do nível superior, assim como considerar as forças relativas entre os níveis e os objetivos gerais.

A prática da tomada de decisões relaciona-se com a avaliação de alternativas, satisfazendo-se um conjunto de objetivos. O problema está em escolher a alternativa que melhor satisfaz o conjunto total de objetivos. No caso do AHP, o objetivo é a identificação de pesos numéricos para cada alternativa, sendo esta representação numérica o resultado do julgamento ou comparação da relação entre dois elementos que possuem o mesmo objetivo. O grupo de todos os julgamentos deve ser representado através de uma matriz quadrada, onde os elementos são comparados entre si.

O trabalho de [Saaty, 1986] recomenda uma escala que é representada pelos valores inteiros entre 1 e 9, onde o valor 1 representa total indiferença na relevância de um critério em relação ao outro, e o valor 9 significa extrema importância de um critério considerando o outro. Os demais valores representam os estágios intermediários de relevância, sendo que as comparações entre os próprios critérios devem computadas com o valor 1, pois um critério deve sempre ser igualmente importante a ele mesmo. Para representar o elemento mais importante da

comparação será utilizado um valor inteiro da escala, e o menos importante, será o inverso dessa unidade, esta relação de reciprocidade é necessária para manter-se a consistência entre duas atividades ou critérios. Desta forma, os elementos da diagonal principal da matriz serão sempre 1, sendo necessário computar apenas os elementos acima ou abaixo da diagonal principal, visto que os mesmos são inversamente proporcionais. A escala por pode ser observada na tabela 5.7.

O resultado da comparação entre os critérios deve representar as respostas para duas perguntas que seguem:

1. Qual dos dois critérios é mais importante considerando o problema principal?
2. Utilizando a escala de 1-9, da tabela 5.7, considerando o problema principal, qual a relevância entre os critérios?

Relevância	Definição
1	Critérios igualmente relevantes.
3	Um critério é levemente mais relevante.
5	Um critério é moderadamente mais relevante.
7	Um critério é fortemente mais relevante.
9	Um critério é absolutamente mais relevante.
2, 4, 6, 8	Valores intermediários entre os valores adjacentes

Tabela 5.7: Escala de [Saaty, 1986] para o AHP.

A seguir é apresentado na tabela 5.8 um exemplo de matriz proposta de acordo com o método AHP. Neste exemplo A, B, C e D, representam os critérios para uma tomada de decisão. Por exemplo, caso a relevância do critério A em relação ao critério B seja quantificada em 5, este valor está armazenado na matriz no índice [0,1], obviamente a relação entre os sub-critérios B e A, representada na matriz pelo índice [1,0] será 1/5. Através da matriz, pode se observar que, para o objetivo deste problema, A é cinco vezes mais relevante do que B e seis vezes mais relevante do que C.

	A	B	C	D
A	1	5	6	7
B	1/5	1	4	6
C	1/6	1/4	1	4
D	1/7	1/6	1/4	1

Tabela 5.8 : Matriz exemplo de uma tabela AHP

Outro aspecto muito importante nesta metodologia está relacionado com a consistência da matriz, sendo que esta característica deve ser garantida. Isto quer dizer que, a partir de uma quantidade básica de dados, deve ser possível deduzir-se logicamente os demais. Por exemplo, considerando que A é 5 vezes mais relevante do que B, e A é 6 vezes mais relevante que C, podemos inferir que:

$$\left\{ \begin{array}{l} \text{Dado } A = 5 * B, A = 6 * C \\ \text{temos que } B/C = 6/5 \end{array} \right.$$

Desta forma, 6/5 é o valor consistente para a posição [B, C]. Portanto, se o resultado da comparação na posição [B, C] for diferente de 6/5, então a matriz é considerada inconsistente, como ocorre na matriz da tabela 5.8. Caso a matriz não seja consistente, as comparações precisam ser refeitas para que a metodologia alcance resultados coerentes.

5.3.6.3 Estratégia de Negociação

No contexto deste trabalho, para a política de negociação automatizada do GerNU, foi utilizado o método AHP como mecanismo para possibilitar a geração de propostas de preços alternativas para os usuários durante o processo de negociação. Ou seja, caso o cliente não aprove o valor do serviço que foi inicialmente calculado, sendo este baseado nos valores especificados pelo administrador da nuvem, o GerNU utiliza esta metodologia para oferecer um novo valor para o serviço, baseando-se nas preferências do usuário e utilizando parâmetros de negociação também especificados pelo administrador, objetivando com isto, finalizar positivamente a negociação.

Considerando os propósitos do GerNU e adequando-o para utilização do AHP, foi definido como problema principal a qualidade do serviço, no sentido de identificar a partir da natureza computacional do mesmo, quais os aspectos mais relevantes para se alcançar este objetivo, ou seja, identificar dentre os critérios / requisitos do serviço, quais são mais importantes para atender a expectativa do cliente.

A partir desta definição, foi considerado que os subproblemas adequados

para a estrutura hierárquica que o GerNU utilizará para o desenvolvimento do AHP são:

1. O conjunto de hardware
2. Os n atributos de qualidade contratados para o serviço.

O que acontece na prática é que durante a especificação do serviço, também são identificados os graus de relevância de cada critério em relação aos outros, sendo então construída uma matriz para o AHP. Atualmente, considerando os parâmetros para QoS já propostos pelo GerNU, esta matriz terá no máximo dimensão 5X5. Após a criação da matriz, é realizada uma avaliação da consistência na matriz visto que eventualmente o usuário pode ter dificuldade para qualificar os requisitos do serviço. Caso a matriz esteja inconsistente o processo será repetido, a partir de uma interface mais detalhada, destacando-se qual o objetivo deste processo, o que precisa ser identificado e como o mesmo deve acontecer partir da identificação do grau de relevância para cada critério, considerando o objetivo que o cliente deseja alcançar com o serviço. Caso a matriz continue inconsistente o mecanismo de propostas automatizadas é desabilitado, permanecendo apenas a negociação manual, que necessita da intervenção do administrador.

Considerando uma matriz consistente, é possível identificar quais aspectos do serviço são mais relevantes para o usuário, podendo-se então oferecer descontos especificamente relacionados com as necessidades primordiais do usuário. O percentual máximo de descontos e limites para cada etapa de negociação são previamente configurados na base de dados do sistema.

A seguir temos um exemplo de como funciona o AHP no contexto do GerNU, sendo que a tabela 5.9 descreve a especificação do serviço e a tabela 5.10 representa a matriz de decisão gerada pela comparação dos critérios.

Item	Critério	Status
A	Hardware	Contratado
B	QoS - Disponibilidade	Contratado
C	QoS - Política de Backup	Contratado
D	QoS - Monitoramento	Contratado
E	QoS - Variabilidade	Contratado

Tabela 5.9: Critério AHP do GerNU

	A	B	C	D	E
A	1	1/9	1/5	1/3	1/7
B	9	1	9	9	9
C	5	1/9	1	5	1/7
D	3	1/9	1/5	1	1/7
E	7	1/9	7	7	1
Σ	25	1,44	17,40	22,33	10,43

Tabela 5.10: Matriz de decisão

Para a verificação tradicional da consistência proposta por Saaty, o primeiro passo consiste em normalizar a matriz de decisão. Esta tarefa é realizada dividindo-se o valor de cada elemento pelo somatório de sua coluna. O autovetor, que determina a classificação dos fatores é calculado somando-se a linha de cada subproblema. A normalização, o autovetor e a classificação podem ser observados na tabela 5.11. Os valores das tabelas 5.11, 5.12 foram arredondados para duas casas decimais com o objetivo de facilitar a visualização.

	A	B	C	D	E		Autovetor	Classificação
A	0,04	0,08	0,01	0,01	0,01		0,03	5
B	0,36	0,69	0,52	0,40	0,86		0,57	1
C	0,20	0,08	0,06	0,22	0,01		0,11	3
D	0,12	0,08	0,01	0,04	0,01		0,05	4
E	0,28	0,08	0,40	0,31	0,10		0,23	2
Σ	1	1	1	1	1		1	

Tabela 5.11: Matriz Normalizada

A partir da tabela 5.10, percebe-se que para o propósito deste serviço, a disponibilidade foi considerada como o critério mais relevante, seguido pela variabilidade, política de *backup*, monitoramento e por fim o *hardware*. Desta forma, havendo a necessidade e considerando-se os parâmetros de negociação, o GerNU pode criar uma proposta de valor que beneficia a disponibilidade e variabilidade.

Calculando o somatório da multiplicação do autovetor pelo somatório da matriz de decisão descobre-se o autovalor, conforme a tabela 5.12, que será utilizado para calcular o coeficiente de inconsistência através da fórmula 5.1.

Autovetor	0,03	0,57	0,11	0,05	0,23		
∑ matriz decisão	25	1,44	17,40	22,33	10,43		Autovalor
π	0,79	0,82	1,99	1,19	2,44		7,22

Tabela 5.12: Cálculo autovalor

$$\text{CoeficienteInconsistencia} = \frac{(\text{autovalor} - \text{quantidade subproblemas})}{\text{quantidade subproblemas} - 1} \quad (5.1)$$

Saaty determinou o índice aleatório de consistência, baseado na quantidade de critérios avaliados. Neste trabalho, com cinco critérios, deve ser utilizado o valor 1.12 para multiplicar pelo coeficiente de consistência para obter-se a razão de consistência, caso esta seja menor que 10% a matriz pode ser considerada consistente. A razão de consistência deste exemplo é igual a 0,50, logo a matriz não é consistente, sendo assim, o processo de definição de relevância deve ser feito até que seja gerado uma matriz consistente ou encerrado o processo de negociação

O GerNU trabalha com uma negociação dinâmica, permitindo a especificação dos requisitos do serviço e ainda pode oferecer descontos baseados no propósito do serviço, de forma a propiciar que o objetivo do cliente seja alcançado. A figura 5.20 ilustra o processo de negociação, considerando tanto o lado do sistema/provedor quanto do cliente.

Todos os requisitos que compõem o serviço podem ter seus atributos alterados durante a negociação. O cliente pode aceitar a proposta inicial de valor, solicitar uma nova proposta ou encerrar a negociação. Quando o cliente aceita uma proposta os recursos computacionais são reservados e um SLA para a prestação de serviço será gerado automaticamente. Quando for solicitado uma nova proposta, será realizado uma avaliação, a partir dos parâmetros de negociação, para identificar se ainda existe margem para negociação. Havendo margem uma nova proposta será criada e enviada para o usuário. Caso contrário a negociação automatizada se encerra, ainda podendo ser acionado uma negociação personalizada diretamente com o administrador da nuvem.

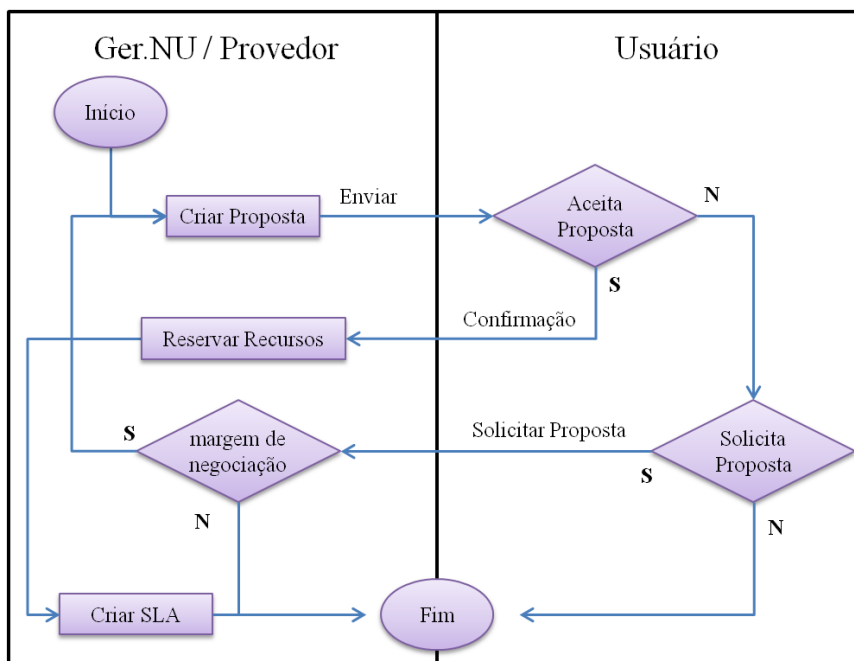


Figura 5.20: Processo de negociação do GerNU

5.3.6.4 Finalização da Negociação

O processo de negociação apresenta duas possibilidades de finalização: uma **Negociação Positiva**, quando se chega a um acordo e então um SLA com as especificações será gerado, ou uma **Negociação Negativa**, onde uma das partes desiste e o acordo não é realizado. Quando uma negociação é positiva, a primeira atividade que ocorre é a avaliação dos recursos restritivos. Esta avaliação consiste em verificar se a nuvem, a partir daquele momento, pode atender os requisitos de *hardware* que estão sendo solicitados.

Esta avaliação representa uma verificação simples, porém essencial ao bom andamento do processo de criação de um serviço. Esta funcionalidade pode ser observada através do algoritmo 5 descrito a seguir. Este algoritmo que recebe o id da nuvem que deve ter seus requisitos consultados, bem como os valores de cada requisito, retornando um valor booleano, sendo que no caso de verdadeiro, os recursos computacionais já ficam reservados, podendo o processo continuar. Caso contrário, não havendo recursos suficientes, é solicitado ao usuário refazer sua especificação de hardware.

Algoritmo 5 - Avaliação dos recursos

```
1: recursos_restritivos (){
2:   Entrada: id_nuvem, cpu, memoria, armazenamento;
3:   boolean recursosOK = falso;
4:   for host 1 to host n
5:     if host(i).cpu >= cpu AND host(i).memoria >= memória AND
           host(i).armazenamento >= armazenamento
6:       recursosOK = true;
7:     end IF
8:   end for
9:   return recursosOK;
10: }
```

A negociação é um processo de tomada de decisão que envolve duas ou mais partes almejando encontrar um acordo que satisfaça os requisitos dos participantes na presença de conflitos e interesses. Cada participante possui interesses independentes, porém todos eles são interdependentes, visto que não conseguem alcançar seus objetivos unilateralmente [Kersten and Lo, 2001]. Neste trabalho, optou-se por uma estratégia que beneficia o usuário, porém, de modo geral, o usuário sempre quer o melhor pelo menor preço, enquanto que o provedor deseja fornecer o mínimo e cobrar o máximo, desta forma, em muitas ocasiões pode não ser possível chegar-se a um acordo.

5.3.7 Gestão do SLA

O gerenciamento do SLA, também denominado de *Service Level Management* (SLM) é o processo que acompanha as atividades relacionadas desde os procedimentos para a criação de um SLA até o acompanhamento dos parâmetros definidos por cada contrato. Ainda podem ser considerados aspectos sobre interações que devem ser ativadas dependendo dos resultados do monitoramento.

Além de parâmetros técnicos, aspectos do serviço da prestadora devem ser acompanhados para garantir a continuidade do contrato e evitar conflitos entre as partes, tais como: nível de satisfação geral do contratante em relação a contratada e qualidade do gerenciamento do serviço. Porém, a conduta do cliente também está

passível de acompanhamento, visando principalmente garantir os aspectos de segurança do ambiente.

Um SLA é definido por um conjunto de objetivos de nível de serviço (SLO), logo, temos que: $SLA = (SLO_1; SLO_2; SLO_3; \dots; SLO_n)$. Cada SLO define um atributo de qualidade dentro do SLA. Desta forma, afirmar que um SLA foi satisfeito implica em que cada um dos níveis dos SLOs envolvidos esteja dentro dos limites estabelecidos no SLA.

A abordagem utilizada pelo GerNU para o gerenciamento dos SLAs baseia-se principalmente em realizar uma avaliação contínua dos parâmetros, considerando não apenas se os limites foram violados, mas também se seus valores se encontram próximos aos limiares acordados, identificando tendências para violações do SLA, sendo possível notificar o administrador que deverá avaliar as ocorrências e tomar as medidas cabíveis para evitar uma violação. Desta forma, será possível interagir com o usuário, objetivando identificar possíveis causadores de sobrecargas nos níveis dos SLOs, inclusive agindo como um mecanismo para melhorar a usabilidade do serviço e a utilização dos recursos da nuvem, aumentando o nível de satisfação do usuário com o serviço. Para os usuários que optaram por um gerenciamento automático dos recursos do seu serviço, obviamente com um custo mais elevado, através da atualização do modelo do ambiente virtual, os requisitos do serviço serão automaticamente reconfigurados. Já para os clientes que optaram por uma opção mais econômica, periodicamente serão avisados das taxas de utilização dos seus recursos, podendo ser propostos incrementos ou decrementos dos mesmos.

Se os parâmetros de qualidade estiverem dentro do que foi contratado, nenhuma ação, além de registrar as medições, deverá ser tomada. Caso contrário, possíveis ações corretivas poderão ser iniciadas, bem como o administrador será acionado para verificar cada situação, identificando seu grau de gravidade e deliberando quais atividades devem ser acionadas.

Outra característica de interatividade desta abordagem é controle do nível de satisfação do cliente, sendo este realizado periodicamente através de formulários cujo objetivo é identificar se o cliente está, e o quanto está, satisfeito com o serviço ou não. Outro objetivo é identificar se existe a necessidade de melhorias a serem

realizadas nas funcionalidades oferecidas e até mesmo se existe interesse na implementação de novas funcionalidades. A partir destes questionários, o provedor poderá iniciar ou intensificar os esforços para que o cliente seja atendido da melhor maneira, assim poderá oferecer mais recursos ou serviços baseado no feedback do próprio usuário.

A figura 5.21, retrata a política de gerenciamento adotada pelo GerNU.

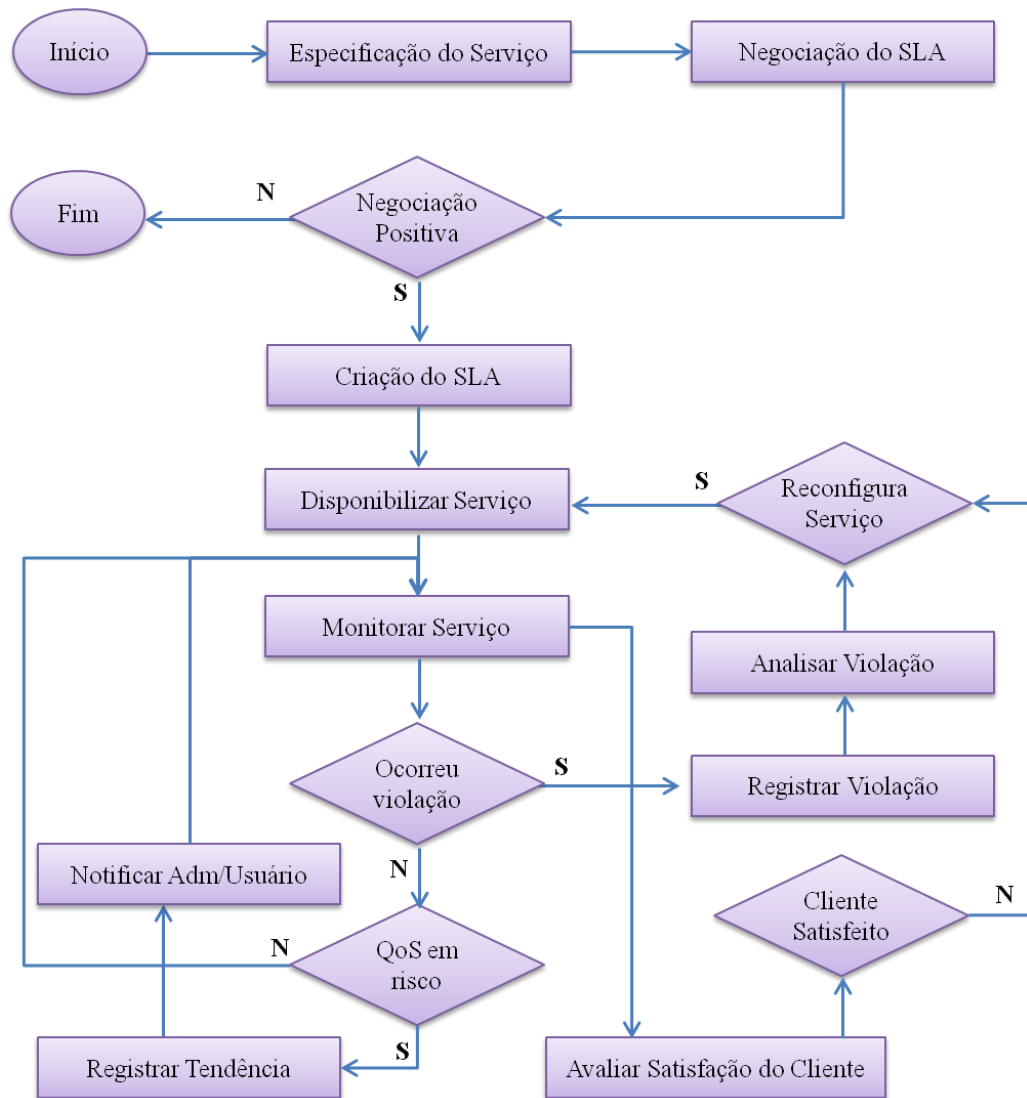


Figura 5.21: Gerenciamento do SLA no GerNU

5.5 CONCLUSÃO

Neste capítulo foi apresentado o GerNU, uma abordagem para o

gerenciamento de nuvens computacionais cujos objetivos iniciais estão relacionados com a padronização do processo de aquisição dos serviços, possibilitando que os mesmos sejam criados e disponibilizados dinamicamente e que ainda seja realizado o controle da qualidade de cada serviço. Foram descritos o processo de aquisição proposto bem como o cenário da aplicação que atende a implementação do mesmo com sua estrutura de agentes que controlam as funcionalidades do GerNU. Além destes foram retratadas a abordagem dirigida a modelos utilizada pela aplicação, as infraestruturas de *hardware* e *software* descrevendo o processo de criação dinâmica dos serviços, os atributos de qualidade propostos que foram baseados no conceito de dependabilidade, o processo de negociação baseado na relevância que os requisitos tem para os usuários e o processo de gestão de SLA.

O próximo capítulo descreverá as os testes realizados a partir da abordagem proposta pelo GerNU, inclusive retratando a avaliação realizada pelos usuários que realizam o teste.

6. AVALIAÇÃO DA PROPOSTA

Inicialmente serão descritas as três perspectivas definidas para implementação da solução proposta, em seguida serão exibidas telas do GerNU com o objetivo de demonstrar sua utilização, por fim será apresentada a avaliação da proposta.

6.1 O GERNU ENVIA O MODELO DO AMBIENTE PARA UM PROVEDOR DISPONIBILIZAR O SERVIÇO

Nesta perspectiva o provedor da nuvem deverá implementar uma parte da solução, desenvolvendo um mecanismo que realiza o mapeamento das informações existentes no modelo de ambiente virtual proposto, para estrutura de imagem virtual que por ele adotada. Observa-se esta perspectiva através da figura 6.1.

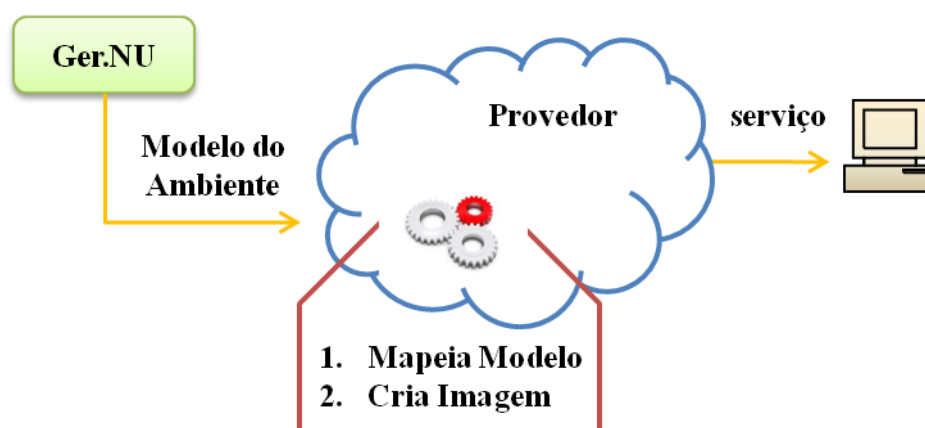


Figura 6.1: Provedor cria imagem e disponibiliza o serviço

O modelo proposto apresenta informações estruturais necessárias para que qualquer fornecedor de nuvem possa criar um serviço a partir da associação das informações do modelo com sua própria infraestrutura. Esta perspectiva é a menos realista, sendo muito improvável que um grande provedor de nuvem queira um mecanismo de mapeamento que inclusive possibilitará que serviços inicialmente criados em sua infraestrutura sejam portados para outros ambientes. Desta forma, devido esta dependência de terceiros, as atividades do GerNU estão restritos a criação de um ambiente onde o usuário pode especificar um serviço, embora este

cenário minimamente possa representar uma espécie de tutorial, descrevendo uma série de parâmetros que um usuário poderá exigir do seu provedor, representando então um *checklist* de requisitos a serem exigidos de um provedor de nuvem.

6.2 O GERNU ENVIA UMA IMAGEM VIRTUAL PARA O PROVEDOR DISPONIBILIZAR O SERVIÇO

O objetivo desta perspectiva é agregar ao ambiente proposto pelo GerNU um mecanismo que faça o mapeamento das informações do modelo de ambiente virtual, para um dos vários tipos de imagem virtual disponíveis no mercado, e que são usualmente utilizados pelos provedores de nuvem. O tipo alvo da imagem virtual dependerá do provedor. Esta realidade representa uma proposta mais realística, considerando o contexto comercial, visto que principalmente facilita a absorção desta proposta por qualquer provedor, que terá apenas o trabalho de instanciar um ambiente virtual pronto, diminuindo consideravelmente o nível de dependência entre as partes, sendo necessário apenas o estabelecimento de um protocolo de comunicação entre o GerNU e o provedor.

A figura 6.2 retrata esta perspectiva, demonstrando as interações necessárias entre o GerNu e o provedor para que o serviço seja efetivamente disponibilizado pelo provedor.

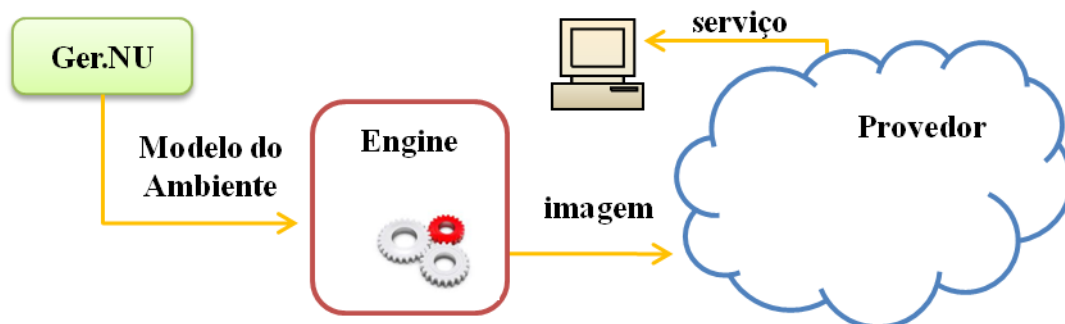


Figura 6.2: GerNU cria imagem e envia para o provedor disponibilizar o serviço

Considerando este cenário, foi utilizado o *middleware* para nuvens focado em usabilidade, habilitado para configurar e monitorar recursos, armazenar arquivos, criar ambientes virtualizados e dedicados, de plataformas a clusters virtuais, possibilitando que sejam acessados remotamente, de forma simples e intuitiva,

através de uma interface web, denominado Neblina [Fernandes et al., 2011], que é um projeto desenvolvido dentro do âmbito das atividades do grupo de pesquisa de Computação Científica Distribuída (ComCIDis) do Laboratório Nacional de Computação Científica (LNCC), onde o autor desenvolveu as atividades relativas a implementação que envolve esta proposta.

A utilização do Neblina foi incentivada pelo fato de que, dado o acompanhamento do seu desenvolvimento no grupo de pesquisas ComCidis do LNCC, a obtenção de informações detalhadas sobre características do seu ambiente de nuvem puderam ser realizadas de forma simplificada, com maior autonomia, acelerando o desenvolvimento. Na perspectiva deste *middleware*, o GerNU contribui para a consolidação da infraestrutura que está sendo criada.

Atuando como um *middleware*, o Neblina tem o propósito de gerenciar e disponibilizar uma grande quantidade de recursos virtualizados entre seus usuários, como serviço, particularizando características, otimizando a capacidade computacional e compartilhando arquivos e informações, possibilitando a interação entre os diversos usuários. Por meio de ambientes virtualizados dedicados e customizáveis, esta ferramenta fornece espaços de trabalhos remotamente acessíveis, com conexão segura, através de um navegador.

O Neblina possui uma interface web que possibilita que os administradores da nuvem também possam controlar todo o ambiente computacional, obtendo informações de estados, informações de recursos, gerenciando, gerando e disponibilizando as plataformas de forma simples e intuitiva.

O Neblina pode usar como hosts de virtualização para compor os recursos da nuvem, qualquer máquina que tenha os requisitos de sua configuração atendidos. Ou seja, até mesmo desktops podem atuar como nós de virtualização desta plataforma. Dessa forma, é possível aproveitar os recursos ociosos destes equipamentos, e não só os de servidores dedicados a isso.

Objetivando implementar um suporte para esta perspectiva foi desenvolvido no GerNU uma funcionalidade responsável por realizar o mapeamento das informações presentes no modelo de ambiente virtual utilizado pelo GerNU para a estrutura de imagem virtual utilizada pelo Neblina.

Atualmente o Neblina utiliza como mecanismo de virtualização o VirtualBox da Oracle, sendo possível utilizar qualquer um dos formatos suportados por este virtualizador. Desta forma, durante o mapeamento, o GerNu cria uma imagem virtual do tipo Open Virtualization Format (OVF). Este formato foi definido por vários líderes no campo da virtualização, com o objetivo de ser uma plataforma independente, eficiente, extensível e aberta, sendo um formato para distribuição de máquinas virtuais. O propósito foi tentar criar um padrão de formato de imagens virtuais, estabelecendo-se uma distribuição segura e flexível.

Primeiramente foi necessário interagir com a equipe de desenvolvimento do Neblina com o objetivo de negociar um protocolo de interação que permitiria a troca das informações necessárias, tais como dados do usuário, localização imagem, dentre outras, de maneira a viabilizar o provimento da solução desejada. Neste sentido, foi estabelecido a utilização de um Webservice para comunicação entre as aplicações, sendo necessário ao pleno e correto funcionamento da proposta que o usuário tenha cadastros dos dois ambientes, no GerNU para especificar o serviço e no Neblina para acessar o serviço.

Após a especificação do serviço no ambiente do GerNU é realizado um mapeamento destes dados para ser criado dinamicamente uma imagem virtual. Sendo assim, o Neblina recebe esta imagem virtual e uma série de informações do modelo do ambiente virtual que foram previamente determinadas, sendo então, executado toda a configuração necessária dentro do seu ambiente, possibilitando então que o serviço seja disponibilizado para o usuário.

Foram realizadas várias simulações, onde um usuário utiliza o ambiente para fazer a especificação do seu serviço e negociar o SLA. Após estas atividades o modelo do ambiente virtual é extraído, sendo possível observar um exemplo deste modelo na figura 6.3. Todas as simulações realizadas no ambiente de testes nos laboratórios do LNCC, até o presente momento, foram satisfatórias, sendo o serviço criado e disponibilizado para o usuário corretamente, validando a abordagem proposta. A figura 6.4, retrata a tela do Neblina exibindo os serviços disponíveis para um determinado usuário, sendo a figura 6.5 utilizada para exibir o serviço iniciado e pronto para utilização através de um acesso remoto.

```

<?xml version= "1.0"?>
<EnvironmentModel>
  <User id="">
    <name= "HPB" address= "Rua Abcd, 17"
      birth_dt= "12.12.1912" numbercard= "123412341234" >
  </User>
  <HardwareRequirements>
    <cpu= "4" memory= "4096" storage= "500"
      ip= "146.134.234.104" >
  </HardwareRequirements>
  <SoftwareRequirements>
    <os>
      <type= "Ubuntu Server 11.10" bits = "64" >
    </os>
    <development>
      <app= "MySQL 5.5" >
      <app= "NetBeans 7.1.1" >
      <app= "TomCat 7.0" >
    </development>
    <production>
    </production>
  </SoftwareRequirements>
  <QoSRequirements>
    <availability= "1" backup= "3"
      monitor= "1" variability= "2" >
  </QoSRequirements>
</EnvironmentModel>

```

Figura 6.3: Modelo do ambiente virtual

Alcançar resultados positivos a partir da utilização de um *middleware* dedicado representa uma possibilidade real de utilização com outros *middlewares*. Como objetivo futuro, desejamos realizar outros testes de integração com diferentes *middlewares*, objetivando os procedimentos, dificuldades e resultados alcançados.

6.3 O GERNU PROVISIONA O SERVIÇO

O objetivo principal desta perspectiva é que, a partir do modelo de ambiente virtual, o GerNU crie e disponibilize os serviços para os usuários no próprio ambiente

de nuvem em que está inserido.

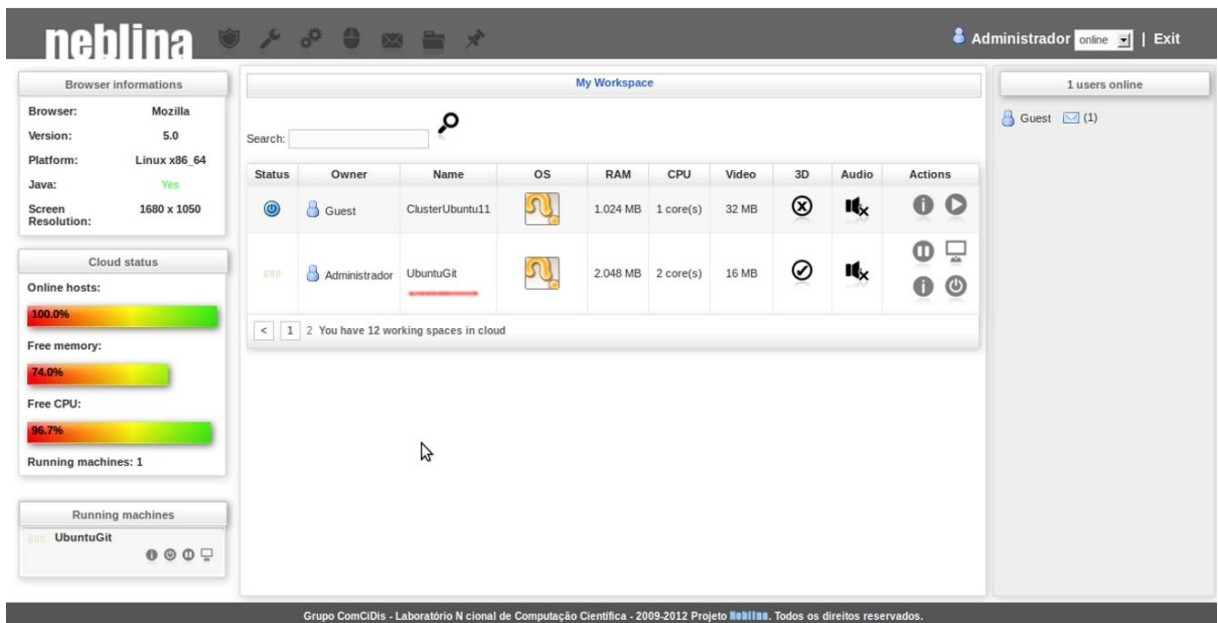


Figura 6.4: Neblina exibindo os serviços disponíveis

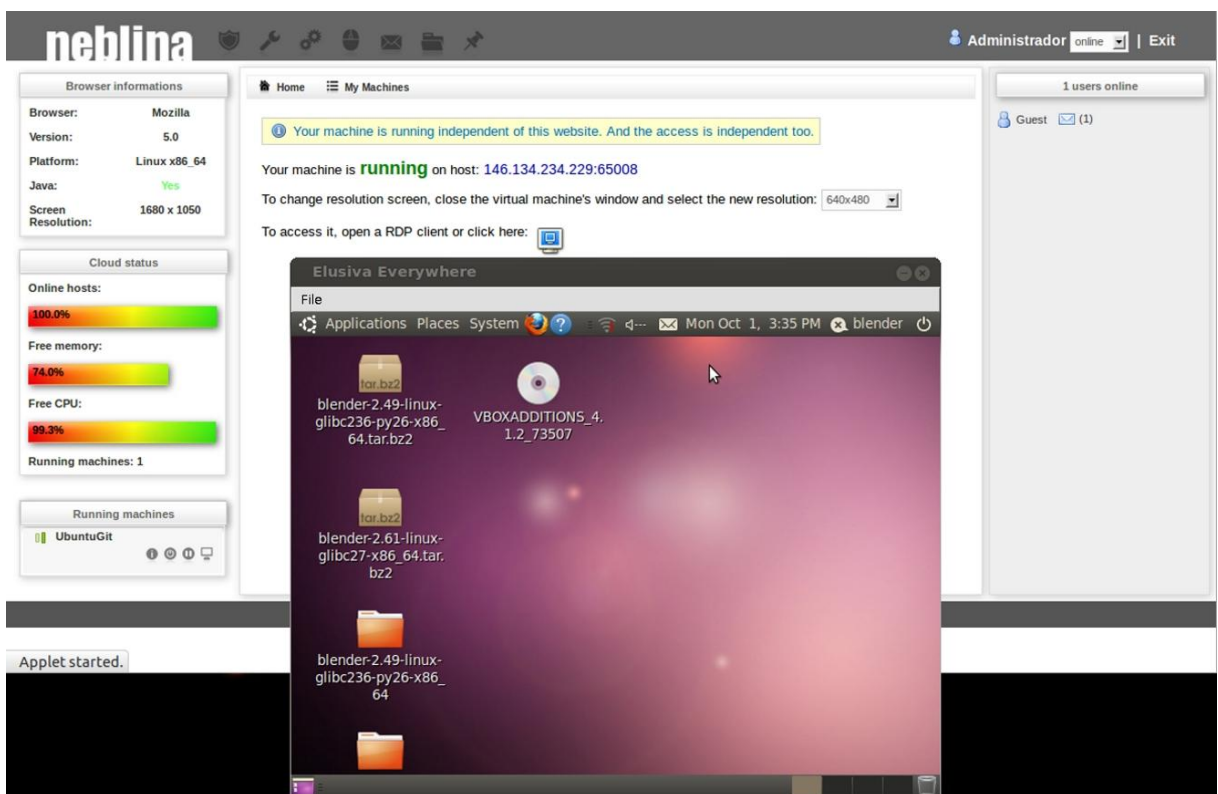


Figura 6.5: Neblina disponibilizando remotamente o serviço para o usuário

Esta perspectiva não cria nenhuma dependência com terceiros para

disponibilizar o serviço, sendo toda estrutura controlada pelo próprio sistema, embora crie a necessidade de implementação de uma série de mecanismos relacionados a segurança, infraestrutura, escalonador de recursos, monitoramento da qualidade dos serviços, dentre outros que ainda não foram contemplados nesta pesquisa. Todas as atividades relacionadas a especificação e disponibilização do serviço estão sob a responsabilidade do GerNU adicionando independência e flexibilidade a esta solução. Pode-se observar esta perspectiva na figura 6.6.

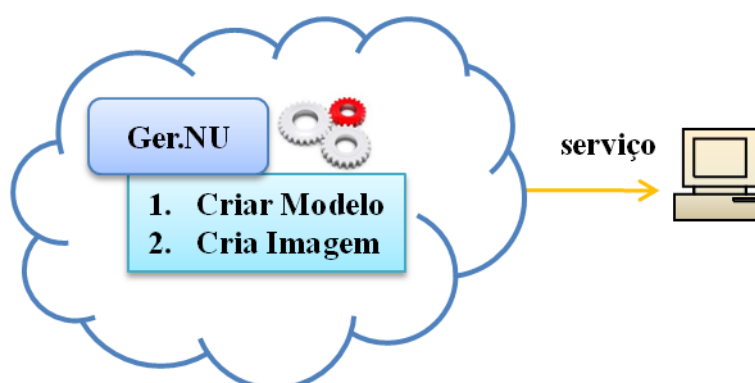


Figura 6.6: O GerNU cria a imagem e disponibiliza o serviço

Inicialmente, a definição de qual host e quais recursos computacionais serão utilizados foi feita de forma simplificada, objetivando-se estruturar o processo de criação e disponibilização dos serviços, validando o objetivo principal do GerNU. Neste sentido, também foi utilizado o VirtualBox para o processo de criação das máquina virtuais, tendo sido todo o processo de configuração descrito em capítulo anterior. Para o contexto de disponibilização do serviço ao usuário foi utilizado o Elusiva RDP, uma solução que permite conectar remotamente computadores utilizando o *Remote Desktop Protocol*. Esta é uma solução *free* desenvolvida em Java. Após o serviço ter sido devidamente criado e configurado, o serviço pode ser inicializado e através deste mecanismo, que utiliza uma *applet* Java, é provido o acesso remoto seguro ao serviço para o usuário, utilizando-se exclusivamente o browser do mesmo. A partir de então o cliente pode realizar as tarefas que desejar com o seu serviço.

A figura 6.7 retrata a etapa final de criação e configuração do serviço no GerNU, sendo que o usuário recebe no seu browser atualizações sobre o andamento do procedimento. Também pode ser observado o serviço

disponibilizado, estando pronto para utilização.

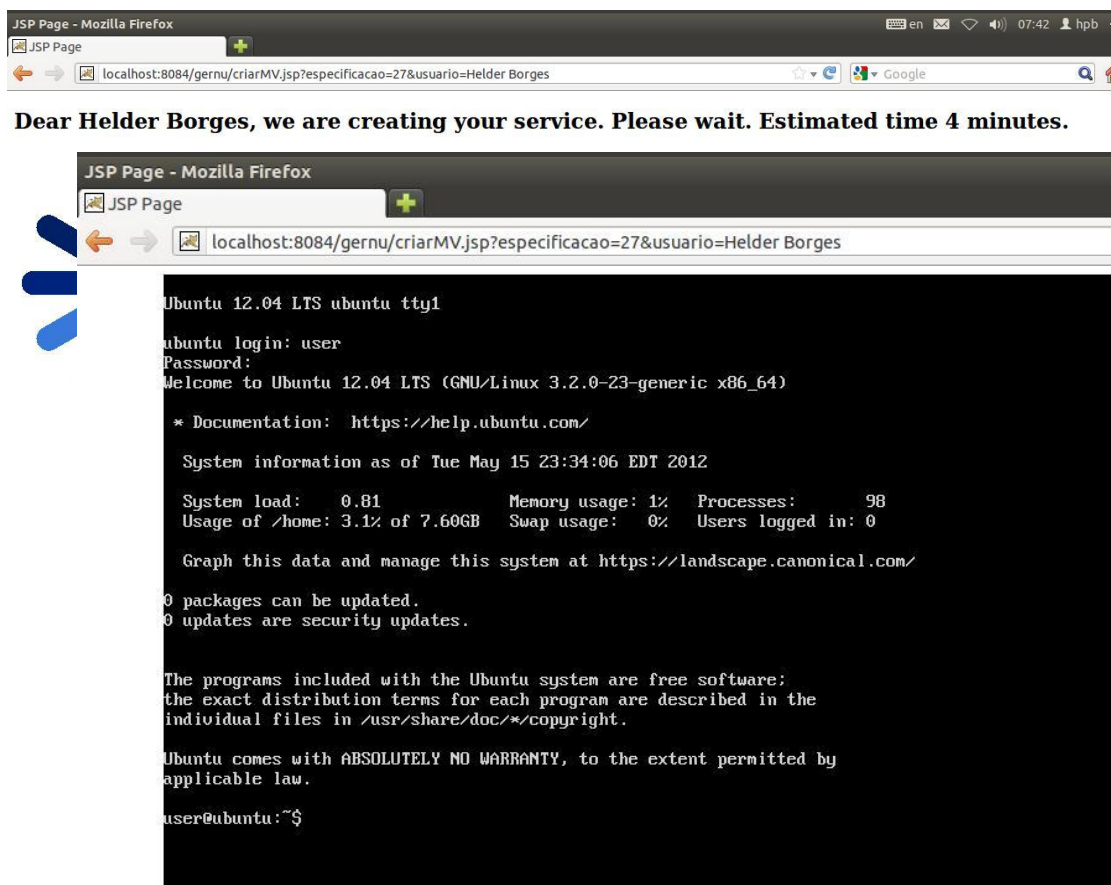


Figura 6.7: Criação e provisão do serviço a partir do GerNU

Outra possibilidade para esta perspectiva está relacionada com sua utilização por usuário que desejam apenas utilizar ambientes virtualizados, sem a necessidade de utilização de recursos de uma nuvem, ou seja, o usuário dispõe de recursos computacionais em seu ambiente e deseja uma imagem virtual para executar no mesmo. Neste contexto, o GerNU poderá realizar todo o processo de especificação do serviço, disponibilizando o download da imagem virtual criada ao final. Desta forma, o usuário poderá utilizá-la em seu próprio ambiente.

6.4 UTILIZAÇÃO DO GERNU

A necessidade da especificação do serviço é compartilhada entre as três perspectivas de implementação da proposta deste trabalho, assim, as etapas disponibilizadas pelo GerNU para permitir que o usuário realize esta tarefa podem

ser observadas nas seis figuras a seguir, sendo demonstrado na figura 6.8 a especificação dos requisitos de *software*.



Figura 6.8: Especificação de software

A figura 6.9 retrata o procedimento para especificação dos requisitos de *hardware*, tendo sido disponibilizado as configurações padrões e sendo possível personalizar a especificação. Os parâmetros de qualidade, bem como cada um dos seus atributos que foram garantidos pelo administrador da nuvem, estando disponíveis para serem contratados para os serviços podem ser visualizados na figura 6.10. A etapa da especificação dos serviços que envolve a definição de prioridades relacionadas aos requisitos do serviço que serão utilizadas durante o processo de negociação, pode ser observado na figura 6.11.

A figura 6.12 tem o objetivo de retratar o procedimento de cadastro e login no sistema, enquanto que a figura 6.13 descreve a composição do custo do serviço especificado, permitindo que seja solicitado uma nova proposta de valores para o serviço, sendo então apresentado uma nova proposta baseada na relevância de cada parâmetro para a execução do serviço, devendo ser observado que este cálculo considera informações previamente cadastradas pelo administrador da nuvem.

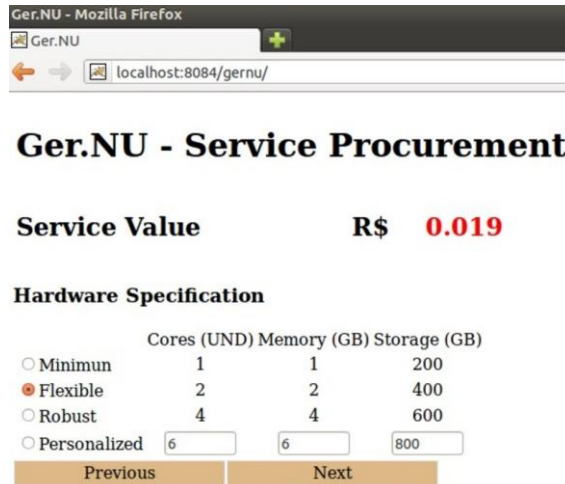


Figura 6.9: Especificação de hardware

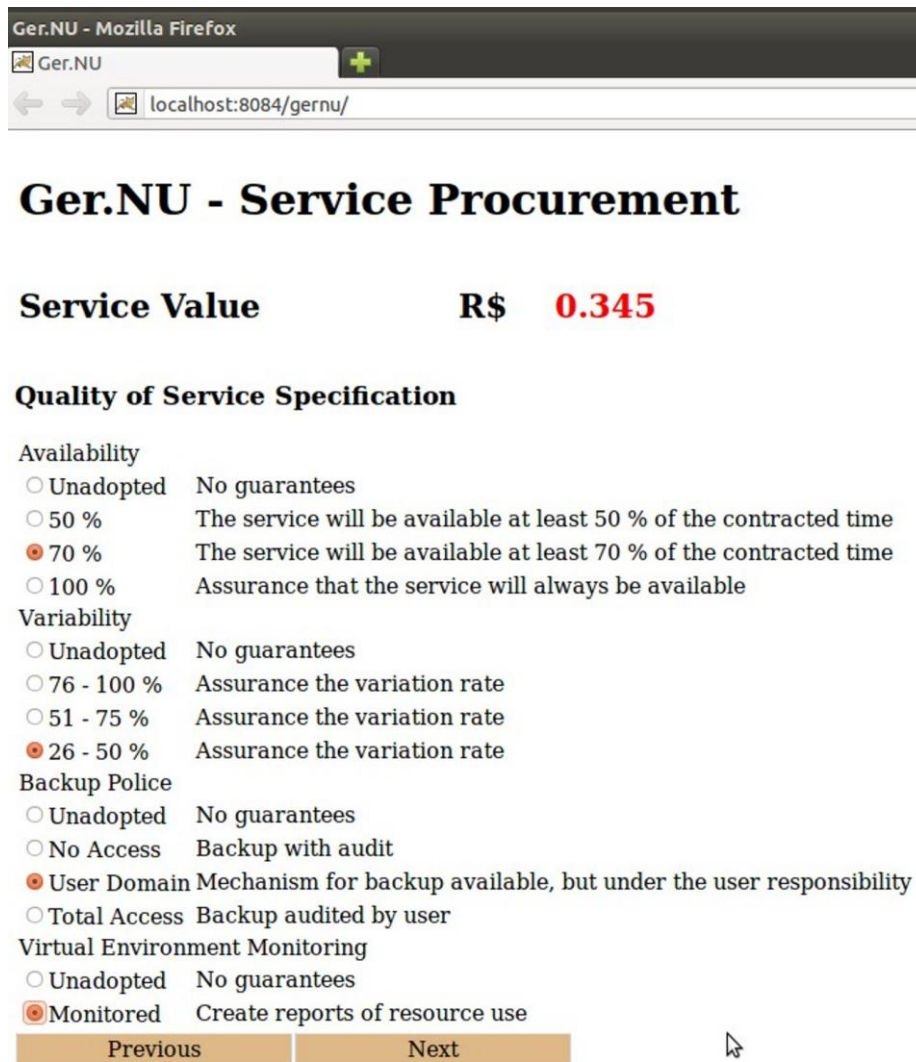


Figura 6.10: Especificação dos parâmetros de qualidade

Ger.NU - Service Procurement

Service Value **R\$ 0.345**

Priority Specification

Please, considering your expectations about the service, let us know which characteristics are more important for you. Which aspect is more relevant for your service? Define a relevance level between these criteria.

<input checked="" type="radio"/> Hardware Performance	<input type="radio"/> Availability	7
<input type="radio"/> Hardware Performance	<input checked="" type="radio"/> Variability	5
<input checked="" type="radio"/> Hardware Performance	<input type="radio"/> Backup Police	6
<input checked="" type="radio"/> Hardware Performance	<input type="radio"/> Virtual Environment Monitoring	5
<input type="radio"/> Availability	<input checked="" type="radio"/> Variability	6
<input checked="" type="radio"/> Availability	<input type="radio"/> Backup Police	5
<input checked="" type="radio"/> Availability	<input type="radio"/> Virtual Environment Monitoring	6
<input checked="" type="radio"/> Variability	<input type="radio"/> Backup Police	9
<input checked="" type="radio"/> Variability	<input type="radio"/> Virtual Environment Monitoring	9
<input type="radio"/> Backup Police	<input checked="" type="radio"/> Virtual Environment Monitoring	6

Previous Next

Figura 6.11: Especificação da prioridade dos requisitos

Login - Identification

Registered User New User

Name

Surname

CPF

Credit Card

Email

User

Password

Password Confirmation

Registered User New User

User

Password

Login Criar Usuário

Figura 6.12: Login e cadastro de usuários



Figura 6.13: Composição de custos

6.5 AVALIAÇÃO DO GERNU

A metodologia utilizada foi a aplicação de questionário fechado com o objetivo principal foi constatar se o GerNU efetivamente implementava o padrão proposto para especificação e provisionamento de serviços, além de criar corretamente o serviço que foi especificado por cada um dos usuários. Justifica-se o questionário fechado pela facilidade de interrogar várias pessoas em um curto espaço de tempo com perguntas relacionadas com os objetivos do trabalho, requerendo menos esforço por parte dos participantes. Além disto, apresenta maior uniformidade, rapidez e simplificação na análise das respostas, facilitando a categorização das respostas.

O processo de validação utilizado para avaliar o funcionamento do GerNU, consistiu na disponibilização da aplicação para um grupo de 15 usuários, constituído por pesquisadores em nível de alunos de graduação, graduados, mestres e doutores, quadro quantitativo descrito na tabela 6.1. Cada usuário especificou pelo menos três serviços, sendo apresentado o questionário (Anexo I) para que fosse realizado a avaliação do GerNU

Nível	Quantidade
Graduandos	3
Graduados	4
Mestres	6
Doutores	2

Tabela 6.1: Perfil dos usuário

Os usuários utilizaram o GerNU para especificar seus serviços, sendo que após a utilização do ambiente, cada um dos avaliadores respondeu um questionário (anexo 1), a partir do qual foram extraídos os resultados retratados nas imagens a seguir, sendo utilizados valores percentuais da amostra. As perguntas eram simples e objetivas e tinham as possíveis respostas presentes em cada imagem.

A figura 6.14 representa a opinião dos usuários quanto a usabilidade no ambiente do GerNU. Outro contexto avaliado está relacionado com a corretude do provisionamento do serviço, sendo este ajuizamento representado pela figura 6.15.

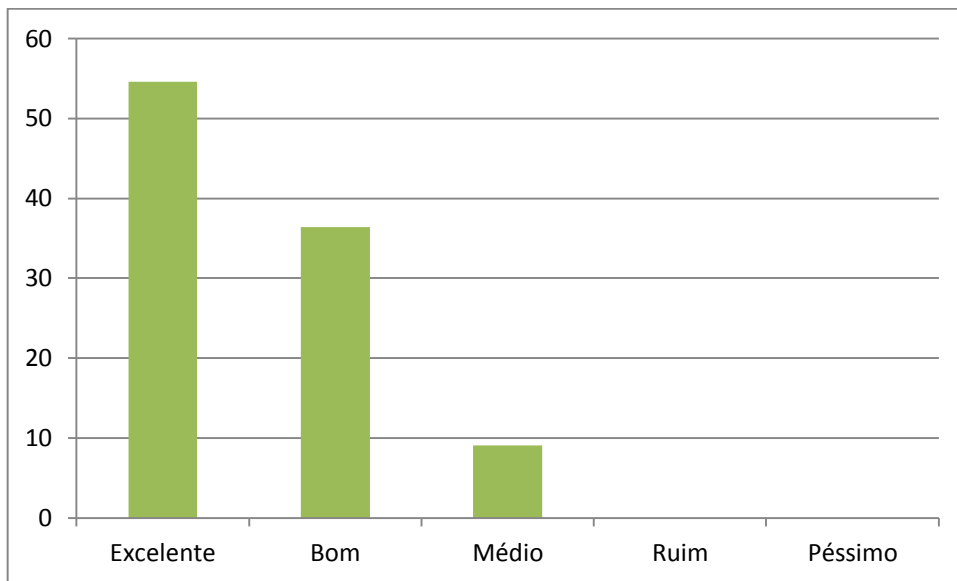


Figura 6.14: Avaliação de usabilidade

A figura 6.16 tem o objetivo de demonstrar a avaliação dos usuários quanto ao tempo despendido durante o processo de especificação e disponibilização do serviço, sendo que o tempo médio encontrado para a realização de todas as etapas foi de 4,5 minutos. Esta média pode ser considerada razoável, visto que a criação de toda estrutura é realizada do zero.

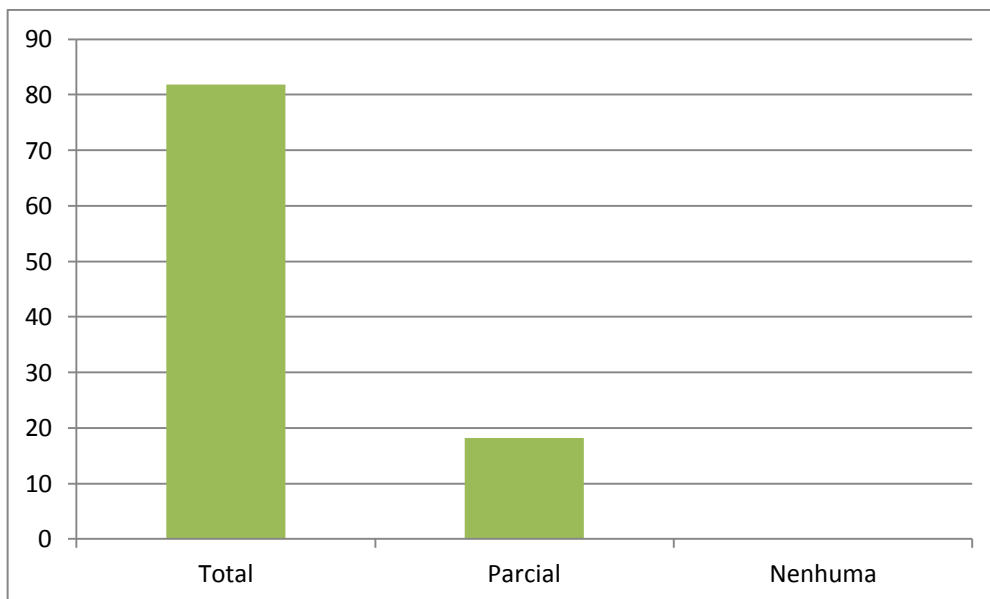


Figura 6.15: Avaliação da corretude

A figura 6.17 exibe os resultados relativos as considerações dos usuário relacionadas a flexibilidade proporcionada pelo GerNU para a realização da especificação do serviço.

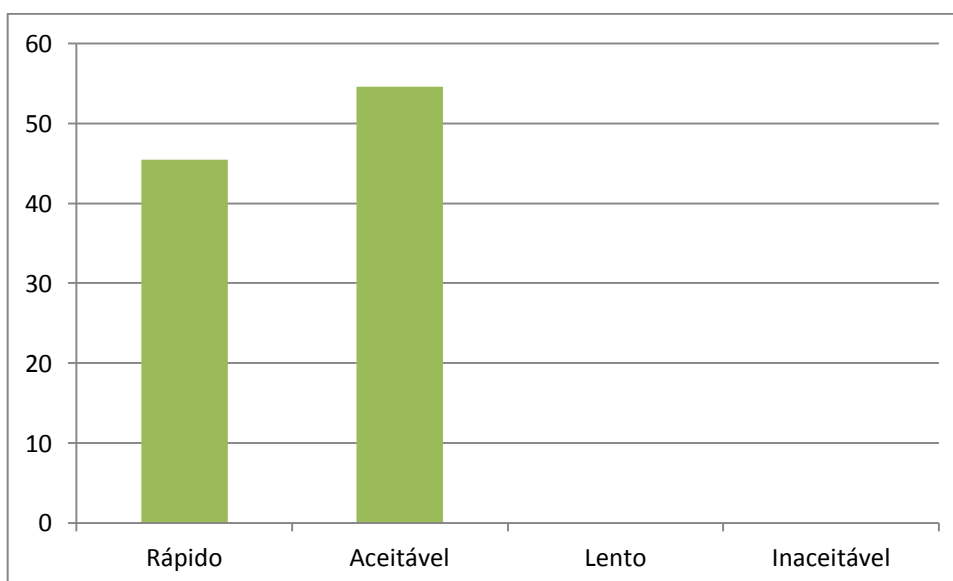


Figura 6.16: Avaliação do tempo de resposta

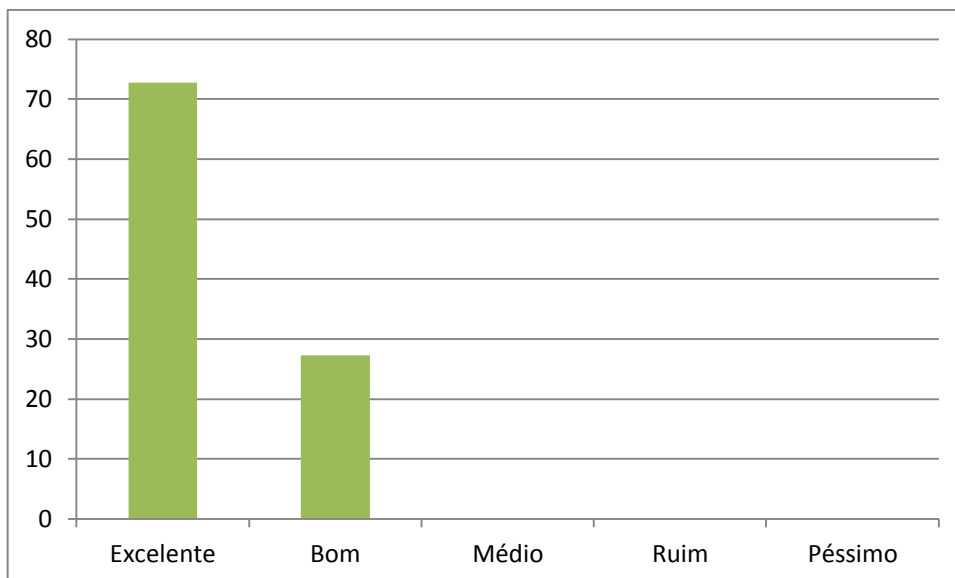


Figura 6.17: Avaliação do nível de flexibilidade para especificação do serviço

Outro aspecto considerado nesta avaliação, foi o nível de relevância que os usuários atribuíram ao fato de obterem descontos exatamente nos parâmetros que consideram mais importantes para a execução dos seus serviços, abordagem adotada pela política de negociação do GerNU, sendo os resultados exibidos a partir da figura 6.18.

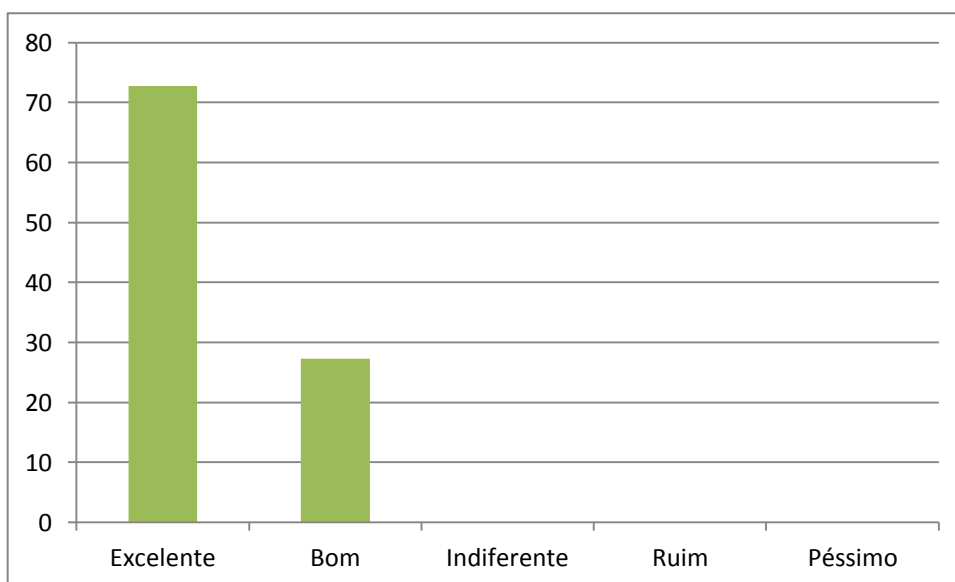


Figura 6.18: Avaliação da metodologia de negociação

A figura 6.19 apresenta um comparativo considerando os descontos oferecidos nos preços originais dos serviços durante as simulações. Percebe-se que

em todos os casos, os usuários foram favorecidos pela política de negociação adotada pelo GerNU.

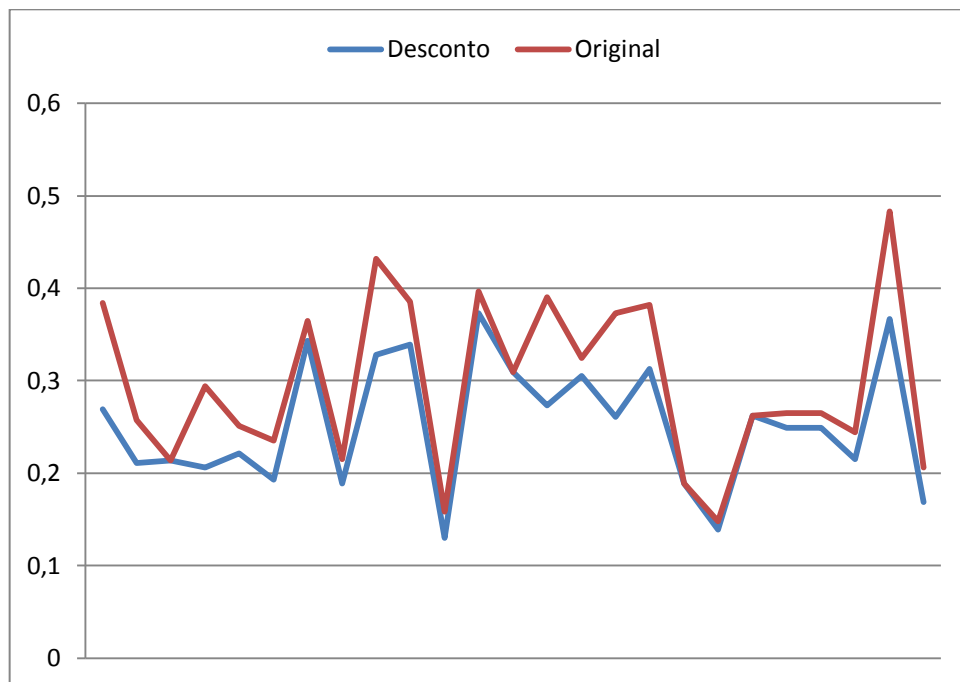


Figura 6.19: Comparativo de valores durante a negociação

6.6 CONCLUSÃO

Neste capítulo foram descritos os processos para avaliação e testes do GerNU. Foram descritas três possibilidades de implementação, sendo efetivamente testadas duas delas. Os primeiros teste foram realizados com o auxílio no Neblina, um *middleware* para nuvens e os demais aconteceram independentemente de terceiros, sendo utilizados apenas os recursos e a estrutura do próprio GerNU.

Também foram apresentados os resultados da avaliação realizada pelos diversos usuários, que através do GerNU puderam especificar um serviço e tê-lo disponível para utilização.

A avaliação geral da abordagem foi considerada satisfatória, concluindo-se que efetivamente apresenta soluções para os três problemas principais alencados como objetivos deste trabalho.

7. CONSIDERAÇÕES FINAIS

Considerando a falta de um padrão para o processo de aquisição de serviços em nuvens computacionais, esta tese propôs uma aplicação denominada GerNU, que além de implementar um processo de aquisição levando em consideração as reais necessidades de *hardware*, *software* e parâmetros de qualidade dos usuários, propõe que os serviços sejam dinamicamente criados e disponibilizados. Além destas particularidades, objetivando satisfazer os requisitos de qualidade contratados pelos usuários foram definidos parâmetros de qualidade relevantes ao contexto de nuvens, sendo possível garantir o contrato estabelecido entre as partes, que estabelece as responsabilidades de cada um.

A abordagem proposta neste trabalho também apresenta o diferencial de mesclar a negociação das garantias do contrato durante o processo de aquisição do serviço, servindo o artefato destas atividades como base para alocação dos recursos e para o monitoramento e asseguramento das garantias.

O objetivo do GerNU é possibilitar que o usuário especifique de forma flexível o seu serviço, definindo exatamente como deve ser o serviço que deseja contratar, sendo o mesmo automaticamente entregue e a partir de então seja possível monitorar sua qualidade.

Objetivando a criação de uma estrutura flexível e facilmente manipulável, foi especificado um modelo de ambiente virtual, sendo este responsável por armazenar as informações necessárias para instanciação de um serviço, contribuindo inclusive com a possibilidade de interoperabilidade entre provedores, sendo esta criada a partir do mapeamento de cada informação do modelo para uma estrutura virtual específica.

O modelo do ambiente virtual passa a ser o artefato mais importante no sistema, sendo que atividades como monitoramento e atualização acontecem baseadas nele, de forma que alterações no ambiente ocorrem primeiro no modelo para automaticamente serem refletidas no ambiente virtualizado.

Como um dos propósitos do GerNU é atender a exata especificação dos

usuários, foi desenvolvido uma abordagem para criação dinâmica de imagens virtuais baseado na utilização de um *preboot execution environment* que foi adaptado para o contexto de nuvens, visto que a estrutura do sistema não possui imagens previamente fabricadas.

Outro importante aspecto envolvido neste trabalho é negociação de SLA, sendo esta indispensável para o processo de aquisição de serviços em nuvem, desta forma, considerando que a automatização do processo de negociação de SLA ainda carece de amadurecimento, este trabalho apresenta uma proposta de negociação baseada em um processo de decisão multicritérios. A partir da especificação do usuário, os parâmetros do serviço são classificados, podendo-se oferecer descontos na característica que o usuário considera mais relevante durante a execução do seu serviço.

7.1 CONCLUSÕES

A partir das avaliações realizadas no GerNU apresentamos as principais conclusões sobre este trabalho.

Genericamente, a implementação de políticas que claramente tem o objetivo de beneficiar o usuário, aumentam consideravelmente o nível de satisfação do mesmo em relação a prestação do serviço, fato este ressaltado nos testes de validação realizados com o GerNU.

A criação de um processo flexível para especificação de serviços em nuvem apresenta um enorme potencial para aumentar o nível de satisfação do usuário em relação ao perfil personalizado que esta opção permite contratar, sendo que mais de 70% dos usuários consideraram como excelente o nível de flexibilidade disponibilizado pelo GerNU e os demais classificaram como bom. A satisfação do usuário facilmente pode ser revertida na contratação de novos serviços e em propaganda.

Observou-se também que o nível de satisfação dos clientes foi incrementado com o provimento de ofertas personalizadas, aumentando a possibilidade de efetivação do contrato. As propostas de valores imediatas facilitam a finalização do

processo e liberam a utilização do serviço pelo usuário. Considerar o que é relevante para o usuário em detrimento da racionalidade individual se mostrou como uma característica muito relevante para os usuários, principalmente pelos benefícios alcançados nos descontos que em média foram de 14,98% do valor original.

A padronização do processo de aquisição de serviços auxilia aos usuários a identificar uma série de requisitos que podem ser exigidos de um provedor quando for necessário contratar um serviço, funcionando como um *checklist* para que este processo seja realizado de forma efetiva e eficiente. Ainda pode se tornar uma medida de comparação para avaliar qual provedor pode oferecer uma melhor estrutura e conseqüentemente um serviço superior.

O GerNu representa ainda um esforço inicial para auxiliar o provisionamento de serviços, porém ele pode ser associado a *middlewares* de nuvem, que serão responsáveis pelo gerenciamento em si, enquanto que o GerNU atende o processo de especificação e criação dinâmica do serviço, especificando parâmetros de qualidade de o provedor deve implementar.

Objetivando retirar todas as dependências de terceiros, todos os mecanismos necessários para atender os objetivos propostos foram implementados a partir do GerNU. Este fato agrega flexibilidade e aumenta as possibilidades de utilização do sistema, inclusive podendo o mesmo ser utilizado apenas para a criação de imagens virtuais que não necessariamente serão utilizadas em ambientes de nuvens computacionais, podendo ser direcionadas apenas a ambientes virtualizados.

Os resultados dos experimentos realizados confirmaram que a solução apresentada pelo GerNU, para os três problemas principais apresentados na Introdução deste trabalho, resolveu corretamente cada um dos desafios, sendo que em mais de 80% dos testes, foi considerado que o serviço disponibilizado estava plenamente de acordo com os requisitos especificados pelo usuário. Desta forma, conclui-se que esta tese alcançou os objetivos propostos, fornecendo uma solução efetiva com uma boa avaliação de usabilidade e que pode ser facilmente adequada a outros padrões de imagens virtuais e ambientes de nuvem.

7.2 CONTRIBUIÇÕES

Até a presente data, foram geradas as seguintes publicações a partir dos resultados obtidos durante o desenvolvimento desta tese:

- Borges, H., de Souza, J., Schulze, B., Mury, A., abril 2012a. Automatic generation of platforms in cloud computing. In: Network Operations and Management Symposium (NOMS), 2012 IEEE. pp. 1311 - 1318.
- Borges, H., de Souza, J., Schulze, B., Mury, A., november 2012b. A Process for Clouds Services Procurement Based on Model and QoS. In: Latin America Conference on Cloud Computing and Communications, 2012 IEEE.

As publicações a seguir foram desenvolvidas em conjunto com a tese, sendo submetidas e aguardando parecer sobre a avaliação.

- Borges, H., de Souza, J., Schulze, B., Mury, A., A Negotiation Approach for Cloud Service Procurement, Simpósio Brasileiro em Rede de Computadores (SBRC).
- Borges, H., de Souza, J., Schulze, B., Mury, A. Dynamic Creation of Services Based on User's Requirements, Special Issue on Cloud Computing in Science & Engineering (CISE).
- Borges, H., de Souza, J., Schulze, B., Mury, A. Automatic Services Instantiation Based on a Pattern Specification, Journal of Network and Computer Applications (JNCA).

Além destes, atrelados a esta tese foram publicados os três relatórios técnicos a seguir.

- Borges, H.P., Souza, J.N., Schulze, B.R., Mury, A.R. Uma arquitetura baseada em Modelos – MDA. LNCC – Petrópolis. 2012.
- Borges, H.P., Souza, J.N., Schulze, B.R., Mury, A.R. Desenvolvimento automático de aplicações e plataformas de trabalho em nuvens computacionais. LNCC – Petrópolis. 2012.

- Borges, H.P., Souza, J.N., Schulze, B.R., Mury, A.R. Computação em Nuvem. LNCC – Petrópolis. 2012.

Estes relatórios estão disponíveis em:

http://www.lncc.br/departamentos/producaocientificageral.php?vMenu=2&vTipo=13&vCabecalho=pesq&vTitulo=lncc&vDepto=&idt_responsavel=&vAno=2012&ano=2012&anof=2012.

Por fim, considerando os pontos de vista, tanto do usuário quanto do provedor da nuvem, podemos destacar os benefícios a seguir:

Perspectiva do Usuário:

- O usuário não precisa ter muitos conhecimentos de como um ambiente de nuvem funciona e de aspectos relacionados com a manutenção da infraestrutura para poder usufruir dos benefícios deste paradigma.
- O processo de aquisição de um serviço é simples e intuitivo ocorrendo através de uma simples seleção de parâmetros, não sendo necessário a instalação e configuração de hardware localmente.
- O tempo para disponibilização do serviço é reduzido e todo o processo é realizado dinamicamente, com mínima intervenção humana, estando o mesmo menos suscetível a erros.
- O usuário pode fazer de forma clara e objetiva um prospecto em relação ao custo benefício de contratação e utilização de um serviço.

Perspectiva do provedor:

- O objetivo é desenvolver procedimentos o mais automatizados possível, reduzindo-se desta forma, a intervenção humana, assim as despesas relacionadas a alguns agentes humanos poderão ser reduzidas.
- O GerNU pode ser utilizado por vários usuários, quanto maior for este número, menores serão os custos com a manutenção da infraestrutura, principalmente porque ela é compartilhada entre todos os usuários.
- Arelado aos custos reduzidos, os serviços poderão ser oferecidos com menores taxas, sendo possível alcançar novos usuários, alcançando-se

maiores lucros.

- Em relação a gerenciamento e manutenção, todas as atividades são disponibilizadas transparentemente para todos os usuários ao mesmo tempo.
- A consolidação dos serviços deve permitir uma rápida localização e correção de problemas.

7.3 TRABALHOS FUTUROS

Com o objetivo de tornar o GerNU uma ferramenta mais completa, capaz de oferecer outras funcionalidades e recursos, vislumbra-se a seguir alguns possíveis trabalhos futuros.

Como alguns parâmetros de qualidade já foram especificados, considerando a natureza complexa da nuvem, se faz necessário criar mecanismos para computar cada um deles, sendo então especificadas políticas de monitoramento, calculando valores, armazenando, verificando tendências e violações de fato, inclusive definindo procedimentos para os casos previstos no contrato.

Durante o processo de negociação são armazenados diversos dados. A criação de mecanismos para a reutilização destas informações, podem auxiliar o monitoramento da qualidade dos serviços, revelando para quais requisitos devem ser aplicados maiores esforços para sua preservação, em conformidade com a relevância do usuário.

Outra possibilidade de utilização da proeminência de um parâmetro em relação a outro, está relacionada com a possibilidade de oferecer recursos extras com menores taxas, baseado nos requisitos mais importantes para a execução do serviço, de forma a incrementar a satisfação dos usuários.

Avaliação da melhor opção em relação ao escalonamento de recursos, se priorizar flexibilidade, criando-se um escalonador privado resolvendo-se toda a complexidade envolvida representa um melhor custo benefício do que a utilização de um escalonador já reconhecido.

Dado o tempo médio para disponibilização do serviço, considerando-se o cenário da internet é muito tempo, porém para todas as atividades que são realizadas, representa um tempo aceitável.

Dentro deste contexto, pretende-se definir abordagens para a reutilização de imagens virtuais que já foram criadas no ambiente do GerNU, assim poderá ser avaliado o grau de similaridade de uma dada especificação em relação as demais já construídas, podendo ser iniciado um processo de alteração, não sendo necessário uma construção completamente do zero. Este procedimento possibilitará a entrega de um serviço ainda mais rapidamente.

REFERÊNCIAS

- [Andrieux et al., 2007] Andrieux, A., Czajkowski, K., Dan, A., Keahey, K., Ludwig, H., Nakata, T., Pruyne, J., Rofrano, J., Tuecke, S., Xu, M. Web services agreement specification (ws-agreement). Grid Forum Document GFD.107, The Open Grid Forum, Joliet, Illinois, United States, 2007.
- [Alhamad et. al, 2010a] Alhamad, M., Dillon, T., Chang, E. Conceptual SLA framework for cloud computing. Digital Ecosystems and Technologies (DEST), 2010 4th IEEE International Conference on , vol., no., pp.606-610, 13-16 April 2010.
- [Alhamad et. al, 2010b] Alhamad, M., Dillon, T., Chang, E. SLA-Based Trust Model for Cloud Computing. Network-Based Information Systems (NBiS), 2010 13th International Conference on, vol., no., pp.321-324, 14-16 Sept. 2010.
- [Amazon, 2009] Amazon. Amazon EC2. Disponível em:<<http://aws.amazon.com/ec2/>>. Acessado em: 08 mai 2011.
- [Armbrust , 2009] Armbrust, M., Fox, M., Griffith, R., et al. Above the Clouds: A Berkeley View of Cloud Computing. University of Califórnia at Berkeley Technical Report, 2009.
- [Armstrong et al., 2009] Armstrong, D., Djemame, K. Towards Quality of Service in the Cloud. In Proc. of the 25th UK Performance Engineering Workshop, 2009.
- [Becker and Braunschweig, 2004] Becker, B., Braunschweig T. Choosing research priorities by using the analytic hierarchy process: an application to international agriculture. Swiss Centre for International Agriculture (ZIL), Swiss Federal Institute of Technology, ETHZentrum, SEC C7, CH-8092 Zürich, Switzerland, 2004.
- [Bernsme et al., 2011] Bernsme, K., Jaatun, M., Meland, P., Undheim, P. Security SLAs for Federated Cloud Services. Sixth International Conference on Availability, Reliability and Security, 2011.
- [Bolor et. al, 2010] Bolor, K., Chirkova, R., Salo, T., Viniotis, Y. Heuristic-Based Request Scheduling Subject to a Percentile Response Time SLA in a Distributed Cloud. GLOBECOM 2010, 2010 IEEE Global Telecommunications Conference, vol., no., pp.1-6, 6-10 Dec. 2010.

- [Brandic et. al, 2010] Brandic, I., Emeakaroha, V.C., Maurer, M., Dustdar, S., Acs, S., Kertesz, A., Kecskemeti, G. LAYSI: A Layered Approach for SLA-Violation Propagation in Self-Manageable Cloud Infrastructures. Computer Software and Applications Conference Workshops (COMPSACW), 2010 IEEE 34th Annual, vol., no., pp.365-370, 19-23 July 2010.
- [Brantner et al., 2008] Brantner, M., Florescu, D., Graf, D., Kossmann, D., and Kraska, T. (2008). Building a database on s3. In Proceedings of the 2008 ACM SIGMOD international conference on Management of data - SIGMOD '08, page 251, New York. ACM Press.
- [Buyya, 1999] Buyya, R. (1999) "High Performance Cluster Computing: Architectures and Systems", Prentice Hall, volume 1.
- [Buyya et al., 2008] Buyya, R., Yeo, C., Vnugopal, S. Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities. 2008.
- [Buyya et al., 2009] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., and Brandic, I. (2009). Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Gener. Comput. Syst., 25(6):599{616.
- [Butler et al., 2011] Butler, J., Yahyapour, R., Theilmann, W. Motivation and Overview, Capítulo 1, pp-3-11. In: Wieder, P., Butler, J., Theilmann, W., Yahyapour, R. (Eds.) Service Level Agreements for Cloud Computing. Eds. Springer New York, 2011, 357p.
- [Callou et al., 2010] Callou, G., Sousa, E., Maciel, P., Tavares, E., Araujo, C., Silva, B., Rosa, N., Marwah, M., Sharma, R., Shah, A., et al. (2010). Impact analysis of maintenance policies on data center power infrastructure. In Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on, pages 526–533. IEEE.
- [Carr, 2008] Carr, N. Big Switch: Rewiring the World,from Edison to Google. Norton & Company, 2008.
- [Cearley, 2009] Cearley, D. et al – Hype Cycle for Applications Development – Gartner Group Reporter number G00147982. Disponível em: <<http://www.gartner.com>>. Acessado em: 08 mai 2011.

- [Cern-Esa, 2011] CERN-ESA. Strategic plan for a scientific cloud computing infrastructure for europe. page 24, CERN, 2011.
- [Chavez et al., 1997] Chavez A., Dreilinger D., Guttman R., Maes P.: A Real-Life Experiment in Creating an Agent Marketplace. In: Proceedings of the Second International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology (PAAM'97) (1997).
- [Chan et al., 2004] Chan, A., Kwok W., Duffy, V. Using AHP for determining priority in a safety management system. *Industrial Management & Data Systems*, Volume 104, Number 5 , pg 430–445, 2004.
- [Chen et al., 2004] Chen E., Kersten G. E., Vahidov R.: An E-marketplace for Agent-supported Commerce Negotiations. In: Proceedings of 5th World Congress on the Management of eBusiness (2004).
- [Chen and Yang, 2008] Chen, J. and Yang, Y. A Taxonomy of Grid Workflow Verification and Validation. *Concurrency and Computation: Practice and Experience*, vol. 20, pp. 347-360, 2008.
- [Chen and Zang, 2009] Chen, W. N. and Zhang, J. An Ant Colony Optimization Approach to a Grid Workflow Scheduling Problem With Various QoS Requirements. *IEEE Trans. on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 39, pp. 29-43, 2009.
- [Cheng et al., 2007] Cheng, S., Chang, C., Zhang, L., Kim, T. Towards competitive web service market. In: 11th IEEE International Workshop on Future Trends of Distributed Computing Systems, p 213219.
- [Chhetri et al., 2007] Chhetri M.B., Mueller I., Goh S.K., Kowalczyk R.: ASAPM An Agent-based Framework for Adaptive Management of Composite Service Lifecycle. In: Proceedings of the IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology - Workshops, 2007 (2007).
- [Creese, 2010] Creese, G. SaaS vs. Software: The Release Cycle for SaaS Is Usually (Not Always) Faster. Gartner blog. Gartner, Inc. Retrieved 24 April 2011. (<http://blogs.gartner.com/guy-creese/2010/05/18/saas-vs-software-the-development-cycle-for-saas-is-usually-not-always-faster/>).
- [Cochinwala et al., 2005] Cochinwala, M., Shim, H., Wullert, R. A model-driven approach to rapid service introduction. *Integrated Network Management*, 2005. IM

2005. 2005 9th IFIP/IEEE International Symposium on , vol., no., pp. 659- 672, 15-19.
- [Correia and Brito, 2010] Correia, A. and Brito, F. Defining and Observing the Compliance of Service Level Agreements: A Model Driven Approach, Quality of Information and Communications Technology (QUATIC), 2010 Seventh International Conference on the, vol., no., pp.165-170, Sept. 29 2010-Oct. 2 2010.
- [Sun et. al, 2010] Sun, D., Chang, G., Guo, Q., Wang, C., Wang, X. A Dependability Model to Enhance Security of Cloud Environment Using System-Level Virtualization Techniques. Pervasive Computing Signal Processing and Applications (PCSPA), 2010 First International Conference on , vol., no., pp.305-310, 17-19 Sept. 2010.
- [Dawoud, 2010] Dawoud, W., Takouna, I., Meinel, C. Infrastructure as a service security: Challenges and solutions. Informatics and Systems (INFOS), 2010 The 7th International Conference on , vol., no., pp.1-8, 28-30 March 2010.
- [Di Nitto et al., 2007] Di Nitto E., Di Penta M., Gambi A., Ripa G., Villani M.L.: Negotiation of Service Level Agreements: An Architecture and a Search-Based Approach. In: Proceedings of the 7th International Conference on Service Oriented Computing, ICSOC 2007 (2007).
- [Distefano et al., 2011] Distefano, S., Puliafita, A., Rak, M., Venticinque, S., Villano, U., Cuomo, A., Di Modica, G., Tomarchio, O. QoS Management in Cloud@Home Infrastructures. Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2011 International Conference on , vol., no., pp.190-197, 10-12 Oct. 2011.
- [Ebeling, 1997] Ebeling, C. An Introduction to Reliability and Maintainability Engineering. Illinois, Waveland Press, 1997.
- [Elfatraty et al., 2004] Elfatraty A., Layzell P.: Negotiating in Service-Oriented Environments. Communications of the ACM 47 (8), 103–108 (2004).
- [Emeakaroha et. al, 2010] Emeakaroha, C., Brandic, I., Maurer, M., Dustdar, S. Low level Metrics to High level SLAs - LoM2HiS framework: Bridging the gap between monitored metrics and SLA parameters in cloud environments. High Performance Computing and Simulation (HPCS), 2010 International Conference on, vol., no., pp.48-54, June 28 2010-July 2 2010.

- [Energy Department U.S., 2011] Energy Department U.S. The magellan report on cloud computing for science. <http://magellan.alcf.anl.gov/>, 2011.
- [Entrialgo et al., 2011] Entrialgo, J., García, D. F., García, J., García, M., Valledor, P., and Obaidat, M. S. (2011). Dynamic adaptation of response-time models for qos management in autonomic systems. *J. Syst. Softw.*, 84:810{820.
- [Eucalyptus, 2011] Eucalyptus. Eucalyptus Systems. Disponível em: <<http://www.eucalyptus.com/>>. Acessado em: 08 mai 2011.
- [Favre, 2004] Favre, J. Towards a Basic Theory to Model Driven Engineering. Workshop in Software Model Engineering (WISME 2004), 2004.
- [Feng et al., 2004] Feng, Y.J., Lu H., Bi K. An AHP/DEA Method for Measurement of the Efficiency of R&D Management Activities in Universities. International Federation of Operational Research Societies, International Transactions in Operational Research, Res. 11, pg 181–191, Published by Blackwell Publishing Ltd, 2004.
- [Fernandes et al., 2011] Fernandes, J. F., Schulze, B., and Mury, A. R. (2011). Neblina - espaços virtuais de trabalho para uso em aplicações científicas. In Simpósio Brasileiro de Redes de Computadores - SBRC 2011 - Salão de Ferramentas.
- [Ferretti et al., 2010] Ferretti, S., Ghini, V., Panzieri, F., Pellegrini, M., Turrini, E. QoS ware Clouds. *Cloud Computing (CLOUD)*, 2010 IEEE 3rd International Conference on, vol., no., pp.321-328, 5-10 July 2010.
- [Fito et al., 2010] Fito, J. O., Presa, I. G., and Guitart, J. (2010). Sla-driven elastic cloud hosting provider. *Parallel, Distributed, and Network-Based Processing, Euromicro Conference on*, 0:111-118.
- [Fowler, 2004] Fowler, M. *UML Distilled: A Brief Guide to the Standard Object Modeling Language*. 3a. edição. Addison-Wesley, 2004.
- [Frankova, 2007] Frankkova, G. *Service Level Agreements: Web Services and Security*. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin, 2007, vol. 4607.
- [Future Internet, 2012] Future Internet Assembly, Bled Declaration on Future Internet April 2012, <http://www.future-internet.eu/index.php?id=47>.

- [Garschhammer et al., 2001] Garschhammer, M., Hauck, R., Hegering, H., Kempter, B., Langer, M., Nerb, M., Radisic, I., Roelle, H., Schmidt, H. Towards Generic Service Management Concepts - A Service Model Based Approach. In Proc. of the 7th International IFIP/IEEE Symposium on Integrated Management (IM 2001), pp. 719–732, May, 2001.
- [Godse, 2009] Godse, M., Mulik, S. An Approach for Selecting Software-as-a-Service (SaaS) Product. Cloud Computing, 2009. CLOUD '09. IEEE International Conference on , vol., no., pp.155-158, 21-25 Sept. 2009.
- [Goiri et al., 2010] Goiri, I., Julia, F., Fitó, O., Macías, M., Guitart, J. Resource-Level QoS Metric for CPU-Based Guarantees in Cloud Providers. Economics of Grids, Clouds, Systems, and Services, ser. Lecture Notes in Computer Science, J. Altmann and O. Rana, Eds. Springer Berlin / Heidelberg, 2010, vol. 6296, pp. 34-47.
- [Grandzol, 2005] Grandzol, R. Improving the Faculty Selection Process in Higher Education: A Case for the Analytic Hierarchy Process. Bloomsburg University of Pennsylvania. IR Applications Volume 6, August 24, 2005.
- [Happe et al., 2011] Happe, J., Theilmann, W., Edmonds, A., Kearney, K. A Reference Architecture for Multi-Level SLA Management, Capítulo 2, pp-13-26. In: Wieder, P., Butler, J., Theilmann, W., Yahyapour, R. (Eds.) Service Level Agreements for Cloud Computing. Eds. Springer New York, 2011, 357p.
- [Hasselmeyer et al., 2007] Hasselmeyer, P., Koller, B., Kotsiopoulos, I., Kuo, D., Parkin, M. Negotiating slas with dynamic pricing policies. In: Proceedings of the SOC@ Inside07.
- [Huang et al., 2004] Huang, S., Chang I., Li S., Lin, M. Assessing risk in ERP projects: identify and prioritize the factors. Industrial Management & Data Systems, Volume 104, Number 8, 2004, pp. 681–688.
- [Iosup et al., 2010] Iosup, A., Ostermann, S., Yigitbasi, N., Prodan, R., Fahringer, T., Epema, D. Performance analysis of cloud computing services for many-tasks scientific computing. IEEE Trans. on Parallel and Distrib. System, 2010.
- [Iqbal et al., 2010] Iqbal, W., Dailey, M., Carrera, D. SLA-Driven Dynamic Resource Management for Multi-tier Web Applications in a Cloud. Cluster, Cloud and Grid

- Computing (CCGrid), 2010 10th IEEE/ACM International Conference on, vol., no., pp.832-837, 17-20 May 2010.
- [IBM, 2011] IBM, Understanding Quality of Service for Web Services. available at: <http://www.ibm.com/developerworks/library/wsquality.html>, accessed on 1st jully 2011.
- [Kim et al., 2010] Kim, H., Lee, H., Kim, W., Kim, Y. A Trust Evaluation Model for QoS Guarantee in Cloud Systems. International Journal of Grid and Distributed Computing, Vol.3, No.1, March, 2010.
- [Karaenke and Kirn, 2010] Karaenke P., Kirn S.: Towards Model Checking and Simulation of a Multi-tier Negotiation Protocol for Service Chains (extended abstract). In: Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2010) (2010).
- [Kersten and Lo, 2001] Kersten, G., Lo, G. Negotiation support systems and software agents in e-business negotiations. In: The First International Conference on Electronic Business, Hong Kong, December, 2001.
- [Kersten et al., 2003] Kersten G.E., Lo G.: An Integrated Negotiation Support System and Software Agents for E-Business Negotiation. International Journal of Internet and Enterprise Management 1 (3) (2003).
- [Kersten and Noronha, 1999] Kersten G.E., Noronha S.J.: WWW-based Negotiation Support: Design, Implementation and Use. Journal of Decision Support Systems 25 (2) (1999).
- [Keller et al., 2003] Keller, A., Ludwig, H. The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services. Journal of Network and Systems Management , 11(1):57–81, 2003.
- [Kuo and Zuo, 2003] Kuo, W. and Zuo, M. (2003). Optimal reliability modeling: principles and applications. Wiley.
- [La, 2009] La, H., Choi, S., Kim, S. Technical Challenges and Solution Space for Developing SaaS and Mash-Up Cloud Services. e-Business Engineering, 2009. ICEBE '09. IEEE International Conference on , vol., no., pp.359-364, 21-23 Oct. 2009.

- [Laprie et al., 1992] Laprie, J., Avizienis, A., and Kopetz, H. (1992). Dependability: Basic concepts and terminology. Springer-Verlag New York, Inc. Secaucus, NJ, USA.
- [Leemis, 1995] Leemis, L. (1995). Probabilistic Models and Statistical Methods. Prentice Hall.
- [Leitner et al., 2010] Leitner, P., Michlmayr, A., Rosenberg, F., and Dustdar, S. (2010). Monitoring, prediction and prevention of sla violations in composite services. In 2010 IEEE International Conference on Web Services, pages 369–376. IEEE.
- [Lisserman, 2010] Lisserman, M. SaaS And The Everlasting Security Concerns. Forrester Research, 2010.
- [Li et. al, 2009] Li, J., Chinneck, J., Woodside, M., Litoiu, M., Iszlai, G. Performance model driven QoS guarantees and optimization in clouds. Software Engineering Challenges of Cloud Computing, 2009. CLOUD '09. ICSE Workshop on , vol., no., pp.15-22, 23-23 May 2009.
- [Liu, 2010] Liu, F., Guo, W., Zhi, Q., Chou, W. SaaS Integration for Software Cloud. Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on , vol., no., pp.402-409, 5-10 July 2010.
- [Liu et al., 2011] Liu, X., Yang, Y., Yuan, D., Zhang, G., Li, W., Cao, D. A Generic QoS Framework for Cloud Workflow Systems. Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on , vol., no., pp.713-720, 12-14 Dec. 2011.
- [Luck et al., 2002] Luck, I., Vogel, S., Krumm, H. Model-based Configuration of VPNs. In Proc. of 8th IEEE/IFIP Network Operations and Management Symposium (NOMS 2002), pp. 589-602, April 2002.
- [Ludwig et al., 2003] Ludwig H., Keller A., Dan A., King R.P., Franck R.:Web Service Level Agreement (WSLA) Language Specification 1.0 (wsla-2003/01/28) (2003) .
- [Macias and Guitart, 2010] Macias, M., Guitart, J. Maximising revenue in cloud computing markets by means of economically enhanced sla management. In: Tech. Rep. UPCDAC-RR-CAP-2010-22, Universitat Politècnica de Catalunya, Computer Architecture Department.

- [Maciel et al., 2010] Maciel, P. R. M., Trivedi, K., Mathias JR, R., and Kim, D. (2010). Dependability Modeling In: Performance and Dependability in Service Computing: Concepts, Techniques and Research Directions. Ed. Hershey: IGI Global, Pennsylvania, USA.
- [Mahbub et al., 2011] Mahbub, K., Spanoudakis, G., Tsigkritis, T. Service Level Agreements for Cloud Computing: Translation of SLAs into Monitoring Specifications. p. 79-101. Springer New York, 2011.
- [Malkowski et al., 2010] Malkowski, S., Hedwig, M., Jayasinghe, D., Pu, C., and Neumann, D.(2010). Cloudxplor: a tool for configuration planning in clouds based on empirical data. In Proceedings of the 2010 ACM Symposium on Applied Computing, SAC '10, pages 391{398, New York, NY, USA. ACM.
- [Marchione et al., 2009] Marchione, F., Fantinato, M., Toledo, M., Gimenes, I. Price definition in the establishment of electronic contracts for web services. In: Proceedings of the 11th International Conference on Information Integration and Web-based Applications & Services,, ACM, p 217224.
- [Marinos, 2009] MARINOS, A., Briscoe, G. Community Cloud Computing. Lecture Notes in Computer Science. Springer. 2009.
- [Marwah et al., 2009] Marwah, M., Sharma, R., Shih, R., Patel, C., Bhatia, V., Mekanapurath, M., Velumani, R., and Velayudhan, S. (2009). Data analysis, visualization and knowledge discovery in sustainable data centers. In Proceedings of the 2nd Bangalore Annual Compute Conference on 2nd Bangalore Annual Compute Conference, pages 1–8. ACM.
- [Marwah et al., 2010] Marwah, M., Maciel, P., Shah, A., Sharma, R., Christian, T., Almeida, V., Araújo, C., Souza, E., Callou, G., Silva, B., et al. (2010a). Quantifying the sustainability impact of data center availability. ACM SIGMETRICS Performance Evaluation Review, 37(4):64–68.
- [MCTI, 2011] MCTI. Estratégia nacional de ciência, tecnologia e inovação 2012 - 2015. <http://www.mct.gov.br/>, 2011.
- [Miezner et al., 2009] Miezner, R., Metzger, A., Leymann, F., Pohl, K. Variability Modeling to Support Customization and Deployment of Multi-Tenant-Aware Software as a service Application. IEEE PESOS'09, May 18-19, 2009.

- [Mihailescu et al., 2011] Mihailescu, M., Rodriguez, A., and Amza, C. (2011). Enhancing application robustness in infrastructure-as-a-service clouds. In Proc. First International Workshop on Dependability of Clouds, Data Centers and Virtual Computing Environments (DCDV 2011) in conjunction with The 41st Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2011), Hong Kong, China.
- [Moon et. al, 2010] Moon, H., Chi, Y., Hacigümüş, H. SLA-Aware Profit Optimization in Cloud Services via Resource Scheduling. Services (SERVICES-1), 2010 6th World Congress on, vol., no., pp.152-153, 5-10 July 2010.
- [Murakami and Almeida, 2003] Murakami, M; Almeida, M; FEA – USP. Decisão Estratégica em TI: Um estudo de caso. 2003. 154 p. Dissertação (Mestrado) – Faculdade de Economia, Administração e Contabilidade da Universidade de São Paulo.
- [Naur and Randell, 1969] Naur, P., Randell, B. Software Engineering: Report on a Conference sponsored by the NATO Science Committee. Scientific Affairs Division, NATO, 1969.
- [Nathuji et al., 2010] Nathuji, R., Kansal, A., Ghaffarkhah, A.Q-Clouds: Managing Performance Interference Effects for QoS-Aware Clouds. Proc. 5th European Conf. Computer Systems, ACM Press, 2010, pp. 237-250.
- [Nash et al., 1950] Nash Jr.J.F.: The Bargaining Problem. Journal of the Econometric Society 18 (2) (1950).
- [Nacional, 2011] National Center for Manufacturing Sciences. Modeling and digital simulation among u.s.manufacturers: The case for digital manufacturing. www.ncms.org, 2011.
- [Ncho and Aimeur, 2004] Ncho A., Aimeur E.: Building a Multi-Agent System for Automatic Negotiation in Web Service Applications. In: Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems (2004).
- [Nimbus, 2011] Nimbus. Nimbus Project. Disponível em: <<http://www.nimbusproject.org/>>. Acessado em: 09 mai 2011.

- [Nist, 2009] NIST. The NIST Definition of Cloud Computing. Disponível em: <http://csrc.nist.gov/publications/drafts/800-145/Draft-SP-800-145_cloud-definition.pdf>. Acessado em: 08 mai 2011.
- [NSF, 2012] NFS, N., 2012. Strategic Technology for CyberInfrastructure. URL <http://www.nsf.gov/pubs/2012/nsf12086/nsf12086.jsp>
- [Open Group, 2004] The Open Group. SLA Management Team, SLA Management Handbook. Enterprise Perspective, 2004.
- [Patel, 2009] Patel, P., Ranabahu, A., Sheth, A. Service Level Agreement in Cloud Computing. Cloud Workshops at OOPSLA09, 2009. Disponível em: <<http://knoesis.wright.edu/aboutus/visitors/summer2009/PatelReport.pdf>>. Acessado em 12 mai 2010.
- [Patterson, 2002] Patterson, D. (2002). A simple way to estimate the cost of downtime. In Proc. 16th Systems Administration Conf. | LISA, pages 185–8.
- [Pedersen et al., 2011] Pedersen, J.M., Riaz, M.T., Celestino, J., Dubalski, B., Ledzinski, D., Patel, A. Assessing Measurements of QoS for Global Cloud Computing Services. Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference on , vol., no., pp.682-689, 12-14 Dec. 2011.
- [Pervez, 2010] Pervez, Z., Khattak, A., Sungyoung L., Young-Koo L. Dual Validation Framework for Multi-Tenant SaaS Architecture. Future Information Technology (FutureTech), 2010 5th International Conference on , vol., no., pp.1-5, 21-23 May 2010.
- [Prasad et. al, 2010] Prasad, K., Faruquie, T., Subramaniam, L., Mohania, M., Venkatachaliah, G. Resource Allocation and SLA Determination for Large Data Processing Services over Cloud. Services Computing (SCC), 2010 IEEE International Conference on , vol., no., pp.522-529, 5-10 July 2010.
- [Pruyne and Singhal, 2008] Pruyne, J., Singhal, S. A Model-based Approach to Service-Oriented Computing, Hewlett-Packard Labs. Technical Report HPL-2008-174, October 2008, <http://www.hpl.hp.com/techreports/2008/HPL-2008-174.html>.
- [Rausand e Hoyland, 2004] Rausand, M. and Hoyland, A. (2004). System reliability theory: models, statistical methods, and applications. Wiley-IEEE.

- [Rego, 2012] Rego, Paulo. FairCPU: uma arquitetura para provisionamento de máquinas virtuais utilizando características de processamento. Fortaleza: UFC, 2012. 81p. Dissertação (Pós-Graduação em Ciência da Computação) - Universidade Federal do Ceará, Departamento de Computação, Fortaleza, Março / 2012.
- [Ritu and Manu, 2011] Ritu, S., Manu, S. Cloud SaaS and Model Driven Architecture. International Conference on Advanced Computing and Communication Technologies (ACCT 2011).
- [Rodosek, 2003] Rodosek, G. A Generic Model for IT Services and Service Management. In Proc. of 8th IEEE/IFIP Integrated Network Management (IM 2003), pp. 171-184, March 2003.
- [Romano et al., 2011] Romano, L., De Mari, D., Jerzak, Z., Fetzer, C. A Novel Approach to QoS Monitoring in the Cloud. Data Compression, Communications and Processing (CCP), 2011 First International Conference on , vol., no., pp.45-51, 21-24 June 2011.
- [Saaty, 1986] Saaty, Thomas. Axiomatic foundation of the analytic hierarchy process. Management Science 32 (7). 1986. 841-855. URL <http://mansci.journal.informs.org/content/32/7/841.abstract>
- [Sahner et al., 1996] Sahner, R., Trivedi, S., Puliafito, A., Performance and Reliability of Computer Systems: An Example-Based Approach Using the SHARPE Software Package. Boston: Kluwer Academic Publishers, 1996.
- [Sanz et al., 2008] Sanz, M., Acuna, J., Cuesta, C., Esperanza, M. Defining Service-Oriented Software Architecture Models for a MDA-based Development Process at the PIM level. Wicsa, pp.309-312, Seventh Working IEEE/IFIP Conference on Software Architecture (WICSA 2008), 2008.
- [Schad et al., 2010] Schad, J., Dittrich, J., and Quian_e-Ruiz, J.-A. (2010). Runtime measurements in the cloud: Observing, analyzing, and reducing variance. PVLDB, 3(1):460{471.
- [Schnjakin et al., 2010] Schnjakin, M., Alnemr, R., and Meinel, C. (2010). Contract-based cloud architecture. In Proceedings of the second international workshop on Cloud data management, CloudDB '10, pages 33{40, New York, NY, USA. ACM.
- [Selic, 2003] Selic, B. The pragmatics of model-driven development. IEEE Software, v.20, n.5, p. 19-25. 2003.

- [Shu and Meina, 2010] Shu, Z. and Meina, S. An architecture design of life cycle based SLA management. *Advanced Communication Technology (ICACT)*, 2010 The 12th International Conference on , vol.2, no., pp.1351-1355, 7-10 Feb. 2010.
- [Snell et al., 2011a] Snell, A., Willard, C., Segervall, L. User site census: Primary applications at hptc sites. <http://www.intersect360.com/>, INTERSEC360, 2011.
- [Snell et al., 2011b] Snell, A., Willard, C., Segervall, L. Top ten hpc research insights from 2011. <http://www.intersect360.com/>, 2011.
- [Sousa et al., 2009] Sousa, F., Moreira, L., Machado, J. *Computação em Nuvem: Conceitos, Tecnologias, Aplicações e Desafios*. In: Antônio Costa de Oliveira;Raimundo Santos Moura;Francisco Vieira de Souza. (Org.). III Escola Regional de Computação Ceará, Maranhão e Piauí (ERCEMAPI). 1 ed. Teresina: SBC, 2009, v. 1, p. 150-175.
- [Srinivasmurthy et al., 2009] Srinivasmurthy, V., Manvi, S., Gullapalli, R., Sathyamurthy, D., Reddy, N., Dattatreya, H., Singhal, S., Pruyne, J. *Web2Exchange: A Model-Based Service Transformation and Integration Environment*. *IEEE International Conference on Services Computing*, 2009.
- [Stapelberg, 2008] Stapelberg, R. (2008). *Handbook of reliability, availability, maintainability and safety in engineering design*. Springer Verlag.
- [Tao et al., 2010] Tao, Q., Chang, H., Yi, Y., Gu, C. A trustworthy management approach for cloud services QoS data. *Machine Learning and Cybernetics (ICMLC)*, 2010 International Conference on , vol.4, no., pp.1626-1631, 11-14 July 2010.
- [Taurion, 2009] Taurion, C. *Computação em Nuvem: Transformando o mundo da tecnologia da informação*. Rio de Janeiro: Brasport, 2009.
- [Theilmann et. al, 2010] Theilmann, W., Winkler, U., Happe, J., Abril, I. *Managing On-Demand Business Applications with Hierarchical Service Level Agreements*. *Future Internet - FIS 2010*. Springer Berlin / Heidelberg, 2010.
- [Vaquero et al., 2009] Vaquero, L., L. Rodero-Marino, J. Caceres, M. Lindner, A break in the clouds: towards a cloud definition, *SIGCOMM Computer Communication Review*, 39 (2009), 137-150.
- [Van et. al, 2009] Van, H., Tran, F., Menaud, J. *SLA-Aware Virtual Resource Management for Cloud Infrastructures*. *Computer and Information Technology*,

2009. CIT '09. Ninth IEEE International Conference on, vol.1, no., pp.357-362, 11-14 Oct. 2009.
- [Vecchiola, 2009] Vecchiola, C., Chu, X., Buyya, R. Aneka: A Software Platform for .NET-based Cloud Computing. High Speed and Large Scale Scientific Computing. IOS Press. Amsterdam. 2009.
- [Vinek et al., 2011] Vinek, E., Beran, P.P., Schikuta, E. Classification and Composition of QoS Attributes in Distributed, Heterogeneous Systems. Cluster, Cloud and Grid Computing (CCGrid), 2011 11th IEEE/ACM International Symposium on , vol., no., pp.424-433, 23-26 May 2011.
- [Xu et al., 2009] Xu, M., Cui, L., Wang, H., Bi, Y. A Multiple QoS Constrained Scheduling Strategy of Multiple Workflows for Cloud Computing. Parallel and Distributed Processing with Applications, 2009 IEEE International Symposium on , vol., no., pp.629-634, 10-12 Aug. 2009.
- [Xiao et al., 2009] Xiao, X., Hailong, S., Xiang, L., Chao, Z. A Basing on Model-Driven Framework of Service-Oriented Software Production Line. Computational Intelligence and Design, 2009. ISCID '09. Second International Symposium on , vol.2, no., pp.139-145, 12-14 Dec. 2009.
- [Xiaoyan et al., 2010] Xiaoyan, J., Yong, Z., Shijun, L. A Well-designed SaaS Application Platform Based on Model-driven Approach. Gcc, pp.276-281, 2010. Ninth International Conference on Grid and Cloud Computing, 2010.
- [Xie et al., 2004] Xie, M., Dai, Y., and Poh, K. (2004). Computing systems reliability: models and analysis. Springer Us.
- [Weiss, 2000] Weiss G.: Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. The MIT Press (2000).
- [Wiboonrat, 2008] Wiboonrat, M. (2008). An Optimal Data Center Availability and Investment Trade-Offs. In Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, pages 712–719. IEEE.
- [Winkler and Gilani, 2011] Winkler, U., Gilani, W. Service Level Agreements for Cloud Computing: Model-Driven Framework for Business Continuity Management. p. 227-250. Springer New York, 2011.

- [Yang et al., 2009] Yang, F., Shanmugasundaram, J., Yerneni, R. (2009). A scalable data platform for a large number of small applications. In CIDR 2009, Fourth Biennial Conference on Innovative Data Systems Research, Asilomar, CA, USA, Online Proceedings, pages 1-10.
- [Yaqub et al., 2011] Yaqub, E., Wieder, P., Kotsokalis, C. Service Level Agreements for Cloud Computing: A Generic Platform for Conducting SLA Negotiations. p. 187-206. Springer New York, 2011.
- [Yu and Buyya, 2005] Yu, J and Buyya, R. A Taxonomy of Workflow Management Systems for Grid Computing. Journal of Grid Computing, vol. 3, no 3-4, pp. 171-200, 2005.
- [Zhang et al., 2011] Zhang, Y., Zheng, Z., Lyu, M.R. Exploring Latent Features for Memory-Based QoS Prediction in Cloud Computing. Reliable Distributed Systems (SRDS), 2011 30th IEEE Symposium on , vol., no., pp.1-10, 4-7 Oct. 2011.
- [Zheng et al., 2010] Zheng, Z., Zhang, Y., Lyu, M.R. CloudRank: A QoS-Driven Component Ranking Framework for Cloud Computing. Reliable Distributed Systems, 2010 29th IEEE Symposium on , vol., no., pp.184-193, Oct. 31 2010-Nov. 3 2010.
- [Zhu and Wang, 2009] Zhu, X. and Wang, S. Software Customization Based on Model-Driven Architecture Over SaaS Platforms. IEEE 978-1-4244-4639-1/09, 2009.

APÊNDICE A - QUESTIONÁRIO DE AVALIAÇÃO

Solicitamos que leia com atenção cada questão e especifique qual a opção retrata melhor a sua experiência durante a utilização do GerNU para especificar seus serviços.

Pergunta 1: Considerando a usabilidade do sistema, qual a opção decreve melhor a sua experiência de utilização em relação ao nível de facilidade para uso?

Excelente Bom Médio Ruim Péssimo

Pergunta 2: Considerando a corretude do sistema, qual a opção decreve melhor a sua experiência de utilização em relação ao nível de correção do serviço disponibilizado (requisitos do serviço em conformidade com a especificação) ?

Total Parcial Nenhum

Pergunta 3: Considerando o tempo de resposta do sistema, qual a opção decreve melhor a sua experiência de utilização em relação ao tempo despendido para disponibilização do serviço, considerando que toda a estrutura é criada do zero?

Rápido Aceitável Lento Inaceitável

Pergunta 4: Considerando a flexibilidade para especificação do serviço, qual a opção decreve melhor a sua experiência de utilização em relação as opções disponibilizadas pelo GerNU?

Excelente Bom Médio Ruim Péssimo

Pergunta 5: Considerando o processo de negociação, qual a opção decreve melhor a sua experiência de utilização em relação a estratégia de negociação com o objetivo de beneficiar o usuário?

Excelente Bom Médio Ruim Péssimo