

Faculdade de Engenharia da Universidade do Porto



IPBrick Cloud Environment

Luís Filipe Félix Martins

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major Telecomunicações

Orientador: Prof. João Manuel Couto das Neves
Proponente: iPortalMais Serviço de Internet e Redes Lda.

Junho de 2010

A Dissertação intitulada

“IPBRICK CLOUD ENVIRONMENT”

foi aprovada em provas realizadas em 20/Julho/2010

o júri

Presidente Professora Doutora Maria Teresa Magalhães da Silva Pinto de Andrade
Professora Auxiliar do Departamento de Engenharia Electrotécnica e de
Computadores da Faculdade de Engenharia da Universidade do Porto



Professor Doutor Bruno Alexandre Fernandes Dias
Professor Auxiliar do Departamento de Informática da Universidade do Minho



Professor Doutor João Manuel Couto das Neves
Professor Auxiliar Convidado do Departamento de Engenharia Electrotécnica e de
Computadores da Faculdade de Engenharia da Universidade do Porto (Orientador)

O autor declara que a presente dissertação (ou relatório de projecto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extractos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são correctamente citados.



Autor - Luís Filipe Félix Martins

Faculdade de Engenharia da Universidade do Porto

Resumo

A tecnologia tem marcado o passo do desenvolvimento das organizações. A evolução tecnológica tem prestado o seu contributo fazendo com que as empresas possam ter uma estrutura, seja a nível de intranet, de segurança ou das comunicações, bem solidificada, dinamizando e simplificando as tarefas do dia a dia. Contudo, todos estes serviços criam uma dependência face aos recursos físicos, nomeadamente os servidores, envolvendo custos muito altos. Surge assim o conceito de Cloud Computing. Este conceito traz às organizações a possibilidade de poder usufruir de mais serviços obtendo menos servidores físicos.

O Projecto IPBrick Cloud Environment consiste na criação de uma plataforma de gestão simples e intuitiva que, recorrendo à virtualização, permite ter um menor número de servidores físicos para gerir uma série de servidores virtuais procurando assegurar a alta disponibilidade dos mesmos, trazendo benefícios às organizações ao nível financeiro, de manutenção, de preservação e de espaço disponibilizado para os servidores.

Abstract

Technology has influenced the development of the organizations. Technological evolution has been given its contribution giving to companies the possibility to have a strong structure, whether the level of intranet, security or communications, stimulating and simplifying the tasks of daily life. However, all these services create a dependence on physical resources, including servers, involving high costs. Here arises the concept of Cloud Computing. This concept brings to the enterprises the possibility to enjoy more services getting fewer physical servers.

The Project IPBrick Cloud Environment is the creation of a simple and intuitive management platform, using virtualization, allowing the enterprises to have the lowest number of physical servers to manage a number of virtual servers foreseeing the high availability, bringing benefits to organizations in financial, maintenance, preservation and space available for servers.

Agradecimentos

Este espaço é dedicado a todas as pessoas que deram a sua contribuição para que este trabalho fosse realizado.

Gostaria de agradecer ao meu orientador, Prof. Dr. João Manuel Couto das Neves, pela forma como orientou o meu trabalho. Agradeço também o esforço desenvolvido na leitura e sugestões de revisão deste documento.

Ao Engenheiro Miguel Ramalhão pela sua disponibilidade, auxílio e orientação durante o desenvolvimento da dissertação.

Aos meus amigos que, sempre que necessário, deram a sua opinião, ajuda e principalmente motivação.

À minha namorada um agradecimento muito especial pela motivação, apoio e compreensão em alturas complicadas.

À minha família, especialmente aos meus pais e avós, pelo apoio durante todo o curso.

Por último, agradeço a todos aqueles que não foram mencionados, mas que de alguma forma também contribuíram para a elaboração deste trabalho.

Índice

Resumo	i
Abstract.....	iii
Agradecimentos	v
Índice.....	vii
Lista de Figuras	xi
Lista de Tabelas	xv
Acrónimos e Abreviaturas.....	xvi
Capítulo 1	1
Introdução.....	1
1.1 - Enquadramento	1
1.2 - Motivação	2
1.3 - Objectivos.....	2
1.4 - Estrutura da dissertação.....	3
Capítulo 2	4
Apresentação do problema.....	4
2.1 - Descrição do Problema.....	4
2.2 - História da Virtualização	5
2.3 - Virtualização de Servidores	5
2.4 - <i>Cloud Computing</i> (Computação em Nuvem)	10
2.4.1 - Modelos de Implementação.....	11
2.4.1.a - <i>Software as a Service</i> (SaaS)	11
2.4.1.b - <i>Platform as a Service</i> (PaaS)	11
2.4.1.c - <i>Infrastructure as a Service</i> (IaaS)	11
2.5 - IPBrick	11
2.5.1 - IPBrick.I	12
2.5.2 - IPBrick.C	13
2.5.3 - IPBrick.GT	13
2.6 - Conclusão	13
Capítulo 3	15
Revisão de Conceitos e Tecnologias	15

3.1 - Ferramentas de Virtualização	15
3.1.1 - KVM	15
3.1.2 - VMware	16
3.1.3 - Xen	16
3.1.4 - Hyper-V	16
3.1.5 - Análise comparativa das ferramentas	17
3.1.6 - XenServer vs. VMware vSphere	18
3.1.6.a - Arquitectura do Sistema	18
3.1.6.b - Instalação, configuração e administração do sistema	19
3.1.6.c - Sistemas operativos suportados pelos sistemas	20
3.1.6.d - Integração com unidades de armazenamento	20
3.1.6.e - <i>Disaster Recovery</i>	20
3.1.6.f - Alta disponibilidade	20
3.1.6.g - Manutenção do Sistema	21
3.2 - Linguagens de Programação	23
3.2.1 - PHP	23
3.2.2 - Javascript	23
3.2.3 - HTML	23
3.3 - Protocolos	24
3.3.1 - XML-RPC	24
3.3.2 - iSCSI	24
3.3.2.a - <i>iSCSITarget</i>	24
3.3.2.b - <i>Logical Unit Number</i>	24
3.3.2.c - Endereço	25
3.3.2.d - Segurança	25
3.3.3 - VNC	25
3.4 - PostgreSQL	25
3.5 - Plataformas existentes	26
3.5.1 - XVP	26
3.5.2 - XenCenter	28
3.6 - Conclusão	29

Capítulo 4 30

Desenvolvimento da Plataforma 30

4.1 - Metodologia utilizada	30
4.2 - Requisitos do sistema	31
4.2.1 - Casos de Uso	31
4.2.1.a - Identificação dos Casos de Uso	32
4.2.1.b - Descrição dos Casos de Uso	32
4.3 - Exploração da ferramenta XenServer	36
4.4 - Arquitectura do sistema	38
4.5 - Base de dados	38
4.6 - Problemas encontrados	39
4.7 - Conclusão	42

Capítulo 5 43

Resultados 43

5.1 - Cenários de teste	43
5.1.1 - Configurações	45
5.1.1.a - Configuração dos Servidores XenServer	45
5.1.1.b - Configuração do Servidor IPBrick	45
5.2 - Análise dos Resultados	45
5.2.1 - Adição do primeiro servidor	45
5.2.2 - Pools	47
5.2.3 - Hosts	49
5.2.4 - Máquinas Virtuais	50

5.2.5 - Repositório de dados.....	52
5.2.6 - Alta disponibilidade	53
5.3 - Compatibilidade.....	53
5.4 - Resultados Obtidos	55
5.5 - Conclusão	56
Capítulo 6	57
Conclusões	57
6.1 - Conclusão	57
6.2 - Trabalho Futuro	58
Anexo A.....	59
Referências	69

Lista de Figuras

Figura 2.1 - Hypervisor (VMM) do tipo “Hosted”	6
Figura 2.2 - Comunicação entre as máquinas virtuais e o Hardware com VMM “Hosted”	7
Figura 2.3 - Hypervisor (VMM) do tipo “Bare-Metal”	7
Figura 2.4 - Comunicação entre as máquinas virtuais e o Hardware com VMM “Bare-Metal”	8
Figura 2.5 - Comunicação entre as máquinas virtuais e o Hardware com VMM “Bare-Metal”	8
Figura 2.6 - Virtualização Completa	9
Figura 2.7 - Paravirtualização	9
Figura 2.8 - Cloud Computing.....	10
Figura 2.9 - Solução IPBrick	12
Figura 3.1 - Tradução binária efectuada pelo VMware ESX [24].....	18
Figura 3.2 - Arquitectura do Hypervisor do XenServer [24]	19
Figura 3.3 - Máquina Virtual Linux num sistema Linux	26
Figura 3.4 - Máquina Virtual Windows num MAC OS X	27
Figura 3.5 - XVPWEB.....	27
Figura 3.6 - Acedendo às propriedades de uma máquina virtual no XVPWEB	28
Figura 3.7 - XenCenter	29
Figura 4.1 - Caso de Uso Adicionar Host	33
Figura 4.2 - Caso de Uso Criar Máquina Virtual.....	34
Figura 4.3 - Caso de Uso Criar Repositório de dados	35
Figura 4.4 - Caso de Uso Activar Alta disponibilidade.....	35
Figura 4.5 - Resultado da função de teste	37
Figura 4.6-Arquitectura do sistema.....	38

Figura 4.7 - Esquema da base de dados.....	39
Figura 4.8 - Família do Processador	41
Figura 4.9 - Activação da flag de Paravirtualização	41
Figura 5.1 - Cenário Inicial	43
Figura 5.2 - Primeiro Cenário	44
Figura 5.3 - Segundo Cenário	44
Figura 5.4 - Adição do primeiro servidor	46
Figura 5.5 - Pools	47
Figura 5.6 - Hosts	49
Figura 5.7 - Máquinas Virtuais	50
Figura 5.8 - Repositório de dados	52
Figura 5.9 - Plataforma no browser Firefox	53
Figura 5.11 - Plataforma no browser Google Chrome	54
Figura 5.10 - Plataforma no browser Internet Explorer	54
Figura 5.12 - Plataforma no browser Safari	55
Anexo A 1 - Login	59
Anexo A 2 - Menu Inicial (Sem servidores adicionados)	59
Anexo A 3 - Formulário Adicionar Servidor	60
Anexo A 4- Configurações	60
Anexo A 5 - Pools (Com um Host em cada Pool).....	61
Anexo A 6 - Formulário Editar Pools	61
Anexo A 7 - Pool (Alta disponibilidade desactivada)	61
Anexo A 8 - Pool (Alta disponibilidade activada).....	62
Anexo A 9 - Hosts	62
Anexo A 10 - Informações sobre o Host (Master e único na Pool)	62
Anexo A 11 - Informações sobre o Host (Slave).....	63
Anexo A 12 - Formulário de Editar Host	63
Anexo A 13 - Máquinas Virtuais.....	63

Anexo A 14-Formulário de Criar Máquina Virtual (Seleccionar Pool)	64
Anexo A 15-Formulário de Criar Máquina Virtual (Especificar Parâmetros)	64
Anexo A 16 - Criar Máquina Virtual (Instalação).....	64
Anexo A 17 - Informações sobre a Máquina Virtual	65
Anexo A 18 - Formulário de Adicionar uma Interface de Rede Virtual	65
Anexo A 19 - Formulário de Adicionar um Disco Virtual	65
Anexo A 20 - Formulário de Editar Máquina Virtual	65
Anexo A 21 - Repositórios de dados	66
Anexo A 22 - Formulário de Adicionar Repositório de Dados (Especificar parâmetros)	66
Anexo A 23 - Formulário de Adicionar Repositório de Dados (Escolha do IQN)	66
Anexo A 24- Formulário de Adicionar Repositório de Dados (Escolha da LUN).....	67
Anexo A 25 - Informações sobre o Repositório de Dados	67
Anexo A 26 - Alta disponibilidade (Desactivada)	67
Anexo A 27 - Alta disponibilidade (Activada)	68

Lista de Tabelas

Tabela 3.1- Comparação das Plataformas	17
Tabela 3.2 - Comparação das funcionalidades oferecidas pelo XenServer e pelo VMware vSphere [24].....	22
Tabela 3.3- Comparação dos custos associados ao XenServer e ao VMware vSphere [24]	23
Tabela 4.1 - Casos de Uso	32
Tabela 4.2 - Templates disponibilizados pelo XenServer	40
Tabela 5.1 - Requisitos XenServer	45

Acrónimos e Abreviaturas

AIX	<i>Advanced Interactive Executive</i>
API	<i>Application Programming Interface</i>
AT&T	<i>American Telephone and Telegraph</i>
BSD	<i>Berkeley Software Distribution</i>
CHAP	<i>Challenge-Handshake Authentication Protocol</i>
CPU	<i>Central Processing Unit</i>
CRM	<i>Customer Relationship Management</i>
CSS	<i>Cascading Style Sheet</i>
DHCP	<i>Dynamic Host Configuration Protocol</i>
DNS	<i>Domain Name System</i>
E/S	<i>Entrada/Saída</i>
ERP	<i>Enterprise Resources Management</i>
EUI	<i>Extended Unique Identifier</i>
HP-UX	<i>Hewlett-Packard UNIX</i>
HTML	<i>Hyper Text Markup Language</i>
HTTP	<i>HyperText Transfer Protocol</i>
HVM	<i>Hardware Virtualization Mode</i>
IaaS	<i>Infrastructure as a Service</i>
IDS	<i>Intrusion Detection System</i>
Intel-VT	<i>Intel - Virtualization Technology</i>
IP	<i>Internet Protocol</i>
IQN	<i>iSCSI Qualified Name</i>
iSCSI	<i>Internet Small Computer System Interface</i>
KVM	<i>Kernel-based Virtual Machine</i>
LAN	<i>Local Area Network</i>
LUN	<i>Logical Unit Number</i>
NAS	<i>Network-Attached Storage</i>
NAA	<i>Network Address Authority</i>
NFS	<i>Network File System</i>
NIC	<i>Network Interface Card</i>
PaaS	<i>Platform as a Service</i>
PBX	<i>Private Branch Exchange</i>
PCI	<i>Peripheral Component Interconnect</i>
PHP	<i>Hypertext Preprocessor</i>
PV	<i>Paravirtualizado</i>

SaaS	<i>Software as a Service</i>
SAN	<i>Storage Area Network</i>
SDK	<i>Software Development Kit</i>
SSH	<i>Secure Shell</i>
SSL	<i>Secure Sockets Layer</i>
TI	<i>Tecnologia da Informação</i>
UcoIP	<i>Unified Communications Over IP</i>
USB	<i>Universal Serial Bus</i>
VHD	<i>Virtual Hard Disk</i>
VMFS	<i>Virtual Machine File System</i>
VMI	<i>Virtual Machine Interface</i>
VMM	<i>Virtual Machine Monitor</i>
VNC	<i>Virtual Network Computing</i>
VPN	<i>Virtual Private Network</i>
WAN	<i>Wide Area Network</i>
XHTML	<i>Extensible Hypertext Markup Language</i>
XML-RPC	<i>Extensible Markup Language - Remote Procedure Call</i>

Capítulo 1

Introdução

Neste Capítulo introdutório abordar-se-á o enquadramento desta dissertação, no que toca ao tema e ao contexto da mesma, a motivação bem como apresentar-se-á os objectivos e a estrutura do documento.

1.1 - Enquadramento

No âmbito do projecto final do Mestrado Integrado de Engenharia Electrotécnica e de Computadores no ramo de Telecomunicações e com especialização em Redes e Serviços de Comunicações, foi desenvolvida esta dissertação, de forma a documentar o trabalho desenvolvido ao longo deste último ano do curso. Esta dissertação tem como tema IPBrick Cloud Environment por um lado devido ao facto de se basear nas tecnologias de virtualização para a concretização do conceito que emergiu recentemente, o *Cloud Computing*, e por outro lado por ser um trabalho que irá ser embebido na tecnologia desenvolvida pela empresa, onde foi feito o estágio do último semestre, a IPBrick, que consiste num sistema operativo para servidores.

É neste contexto, dentro da área de I&D da iPortalMais, que nasceu o tema desta dissertação, que consiste no desenvolvimento de uma ferramenta que permita simplificar a criação, instalação, gestão e recuperação de ambientes de suporte ao "*Private Cloud Computing*" essencialmente para servidores IPBrick. A ferramenta projectada deverá ser capaz de, através de um ambiente gráfico simples, criar várias máquinas virtuais, com os parâmetros pretendidos (memória, número de processadores, tamanho do disco rígido e com a imagem .iso) em qualquer servidor físico disponível e permitir a gestão da política de falhas, ou seja, definir as operações a efectuar em caso de falha, como qual ou quais servidor(es) para que a(s) máquina(s) deverão migrar, caso o servidor onde se encontram falhe.

Hoje em dia com o projecto IPBrick está facilitada a instalação de servidores de Intranet e Comunicações. Desta forma, este projecto tem o objectivo de explorar o mesmo tipo de simplificação para uma plataforma de *Private Cloud Computing*. Como resultado desta dissertação pretende-se chegar à concepção de uma plataforma de *Private Cloud Computing*

que possa ser manejada através de um interface de gestão tradicional a partir de um portal *Web*, onde seja possível gerir várias IPBricks virtuais. Sendo a meta final desta dissertação dentro da iPortalmats, reduzir o *know how* necessário para montar este tipo de subsistemas.

1.2 - Motivação

Hoje em dia, vivemos numa sociedade em constante evolução tecnológica juntamente com o aumento da produção e partilha de informação. Com vista a permitir a preservação e a partilha dessa mesma informação, as empresas recorrem à tecnologia como equipamentos de suporte. Os recursos físicos que as empresas adquirem para solucionar os seus problemas significam grandes custos tanto a nível de investimento, como de manutenção e preservação, e muitas das vezes as empresas deixam de ter capacidade, no que toca à infra-estrutura, para alocar um número elevado de equipamentos.

Devido a esta realidade, surge a necessidade de encontrar um recurso virtual que substitua os equipamentos físicos. Daqui nasce o conceito de *Cloud Computing* que funciona como uma forma de partilha de informação num suporte virtual. A motivação por esta dissertação, nasce da possibilidade de poder aprofundar os conhecimentos nas tecnologias de virtualização e conciliar o desenvolvimento da mesma num estágio, numa empresa Portuguesa que desenvolve tecnologia de ponta para as necessidades das empresas.

1.3 - Objectivos

Os objectivos que estão na base do desenvolvimento deste trabalho são a criação de uma plataforma *Web* simples e intuitiva que permita a adição e a gestão de servidores e máquinas virtuais, A interface deve ser projectada de maneira a que, após a instalação do servidor, permita todo o tipo de operações no mesmo como a junção deste a uma rede de servidores, definir se o servidor é *master* ou *slave* nessa rede, permitir a criação de máquinas virtuais no servidor ou nos restantes servidores da rede e acima de tudo colocar a informação de uma forma intuitiva para que se possa definir e procurar assegurar a alta disponibilidade na rede, ou seja, em caso de falha de algum servidor físico, as máquinas que se encontram no mesmo devem migrar para os restantes servidores presentes na rede, mantendo a informação e as definições da(s) máquina(s) virtual(ais).

A recuperação das máquinas virtuais da *Private Cloud* é um dos objectivos mais importantes, pois no caso de o servidor físico falhar, as máquinas deverão ser transferidas para um ou mais servidores físicos e devem ser preservadas todas as características e funcionalidades, para que as máquinas virtuais continuem a funcionar da mesma maneira, mas localizadas noutra máquina física. A plataforma de virtualização deverá funcionar como um Sistema Operativo, de tal forma que esta será primeiramente instalada e a partir da mesma serão criadas as máquinas virtuais. A plataforma deve ser completa no que toca à mobilidade, terá de ser capaz de importar, exportar e clonar máquinas virtuais com as respectivas configurações, permitir um sistema escalável e deverá garantir *Alta disponibilidade* das máquinas virtuais. As imagens com a extensão *isoe* os ficheiros que

contêm as máquinas virtuais devem encontrar-se num dispositivo externo (*storage*) sempre disponível.

1.4 - Estrutura da dissertação

Esta dissertação está estruturada em seis capítulos, o capítulo 1 - introdução - que já foi apresentado abordando o enquadramento do trabalho a desenvolver bem como a apresentação da empresa que propôs o tema desta dissertação e que disponibilizou a oportunidade de estágio.

No Capítulo 2 - Descrição teórica do problema - serão aprofundados os conceitos apresentados na introdução, a descrição do problema, a identificação das necessidades e a solução proposta. No Capítulo 3 -Revisão de Conceitos e Tecnologias - serão apresentadas as tecnologias passíveis de serem utilizadas, a abordagem às tecnologias de virtualização, ao *Cloud Computing* e o estudo de ferramentas que poderão ser utilizadas no âmbito do projecto.

O Capítulo 4 - Desenvolvimento da plataforma - Apresentar-se-á a metodologia utilizada, os requisitos do sistema, o trabalho realizado, os problemas que foram surgindo bem como os detalhes da implementação da solução projectada.

Os Resultados serão abordados no Capítulo 5, onde serão analisados e discutidos os resultados obtidos, verificando se estes foram de encontro com os requisitos estabelecidos. Por último o Capítulo 5 - Conclusões - Será feita uma análise de todo o trabalho realizado, dos resultados obtidos bem como será feita uma referência ao trabalho que ficará para futuro.

Capítulo 2

Apresentação do problema

Este Capítulo destina-se à formulação e definição teórica do problema que esteve na origem desta dissertação. Será abordada a descrição do Problema, bem como os meios e conceitos que irão ser utilizados, virtualização e *Cloud Computing*.

2.1 - Descrição do Problema

O sector das tecnologias é aquele que sofre mais mudanças e evolução em curtos espaços de tempo. As necessidades começam a emergir cada vez mais diversificadas fazendo com que as tecnologias tenham de abranger mais áreas. É no seio empresarial que essas necessidades começam a surgir mais, fazendo com que as empresas que desenvolvem soluções, tenham que acompanhar o ritmo de evolução e mais do que isso, estar sempre um passo à frente na inovação para que as suas soluções não fiquem obsoletas. Desta forma, a iPortalMais, como empresa que desenvolve soluções para as necessidades empresariais, é caracterizada por ter uma forte área de Investigação e Desenvolvimento mesmo para poder acompanhar a evolução e a inovação da área. Face aos problemas que têm vindo a surgir relacionados com a gestão de servidores físicos, tais como, o investimento em hardware, manutenção e reparação, problemas relacionados ainda, com o espaço disponibilizado para a alocação dos recursos físicos, surge a necessidade de aplicar o conceito de *Cloud Computing*.

A iPortalMais decidiu integrar este conceito dentro do leque das soluções que desenvolve e que disponibiliza às empresas. O *Cloud Computing* permite às organizações responder a algumas das suas necessidades relacionadas com a gestão dos servidores físicos, das ferramentas disponibilizadas na rede bem como com questões relacionadas com a alta disponibilidade dos serviços. A dissertação - IPBrick Cloud Environment, consiste na criação de uma solução conjugando as tecnologias de virtualização e o *Cloud Computing* para responder a essas mesmas necessidades. Para isso, será necessário criar uma plataforma que permita gerir, através da Web, todos esses servidores físicos e virtuais, a alta disponibilidade dos mesmos e em conjunto com as tecnologias IPBrick gerir também as ferramentas, aplicações e serviços disponibilizados.

Para isto, recorrer-se-á às tecnologias de virtualização para concretizar o conceito de Cloud Computing.

2.2 - História da Virtualização

Apesar de não se tratar de um conceito novo, a virtualização encontra-se em destaque na actualidade. A ideia surgiu no início dos anos 60, quando cada computador (*mainframe*) não suportava multitarefa, logo só poderia executar um programa de cada vez. Na altura, os softwares existentes não eram constituídos apenas pela aplicação mas sim também pelo Sistema Operativo completo, onde esta aplicação era executada [1]. A IBM foi a primeira a implementar, no CP-40, o primeiro Sistema Operativo com o conceito de virtualização completa, onde era possível emular até catorze pseudo máquinas simultaneamente, estas que mais tarde foram intituladas de máquinas virtuais [2].

Ao longo dos anos o interesse na virtualização começou a descer devido ao aparecimento de aplicações do tipo cliente/servidor. Outra causa para o decréscimo do interesse, foi a ascensão da plataforma x86, devido ao alto custo associado a um *mainframe* [3]. A grande adesão ao Windows e com a origem do Linux, a indústria foi forçada a adoptar esta plataforma para servidores e estações de trabalho, originando novos desafios [2]. Desafios como o aumento do custo devido à infra-estrutura física necessária, pois estes servidores têm de ser mantidos num local seguro e com as características necessárias para o seu bom funcionamento. Ocorreu também um aumento dos custos devido ao alto consumo de energia, porque os servidores devem-se encontrar sempre operacionais. Outro dos desafios é o aumento da complexidade das redes de computadores, aumenta o custo envolvido com a gestão da mesma, pois são necessários trabalhadores qualificados para efectuar este tipo de gestão. E por fim, o alto custo envolvido na manutenção de *desktops*, pois é preciso gerir os acessos e as políticas de segurança dos mesmos na rede. Estes deverão se encontrar sempre actualizados para não comprometer a segurança dos mesmos, o que eleva também os custos da sua manutenção [2].

Para combater estes desafios, a partir dos anos 90 voltou a surgir o interesse na virtualização, mas ao contrário dos *mainframes*, este tipo de arquitectura não se encontrava preparada para suportar virtualização completa. Passados alguns anos, superando vários obstáculos, a VMware no ano de 1999 lançou o seu primeiro produto para a virtualização em x86 chamado “Vmware Virtual Platform”.

Mais recentemente, em 2005, a Intel e a *Advanced Micro Devices (AMD)* lançaram processadores que permitem a virtualização assistida por hardware, o *Intel - Virtualization Technology* (Intel-VT) e o *AMD-V* respectivamente [2].

2.3 - Virtualização de Servidores

A virtualização, é uma forma de simular um recurso físico através da criação de um ambiente virtual, permitindo a utilização de vários sistemas operativos e aplicações. É uma

técnica que consiste na utilização de um único recurso físico para obter vários recursos virtuais.

A virtualização traz muitos benefícios, tais como a redução de custos com hardware e espaço físico, permitindo assim executar vários sistemas operativos numa única máquina, sem ter de investir em máquinas para utilizar os diferentes sistemas, o que reduz drasticamente os custos com *hardware* e reduz o espaço físico necessário para suportar o sistema pretendido. A confiança e disponibilidade são outros dos benefícios que a virtualização apresenta, ou seja, mesmo que um software falhe os outros serviços não são prejudicados. A possibilidade de poder dar um melhor uso aos recursos da máquina, é mais uma das vantagens da virtualização, isto é, alocar apenas os recursos necessários a cada Sistema Operativo, como *Central Processing Unit* (CPU), *Hard Disk Drive* (HDD), RAM, desta forma é conseguida a melhoria da segurança do sistema, fazendo com que os Sistemas Operativos possam ser isolados numa máquina virtualizada, resultando na criação de máquinas mais seguras. Esta vantagem reduz a necessidade de ter várias máquinas físicas a funcionar em diferentes níveis de segurança [4].

Por outro lado, a virtualização também tem contras, ou seja, também existem desvantagens associadas a esta solução. A nível da segurança como qualquer outro software, o *Virtual Machine Monitor* (VMM) também se encontra sujeito a vulnerabilidades, já que este também se trata de um software, logo em certas situações as máquinas virtuais tornam-se menos seguras que as máquinas físicas. Isto aplica-se, porque se o sistema operativo, onde as máquinas virtuais se encontram hospedadas, apresentar algum tipo de vulnerabilidade, estas também se encontram vulneráveis. No que toca à Gestão, todos os ambientes virtuais necessitam de ser geridos, pois todos precisam de ser monitorizados e configurados. Relativamente ao desempenho, a introdução de uma camada de software, o *hypervisor*, leva a um processamento superior ao que teríamos sem a virtualização [4].

Para virtualizar uma máquina, um dos principais componentes é o VMM, também conhecido por *hypervisor*, que permite executar vários sistemas operativos em simultâneo numa máquina física. Existem dois tipos de *hypervisors*, o *Hosted* ou Tipo 2 e o *Bare-Metal* ou Tipo 1. Na arquitectura *Hosted* o VMM é instalado sobre um Sistema Operativo, como se pode verificar na Figura 2.1.

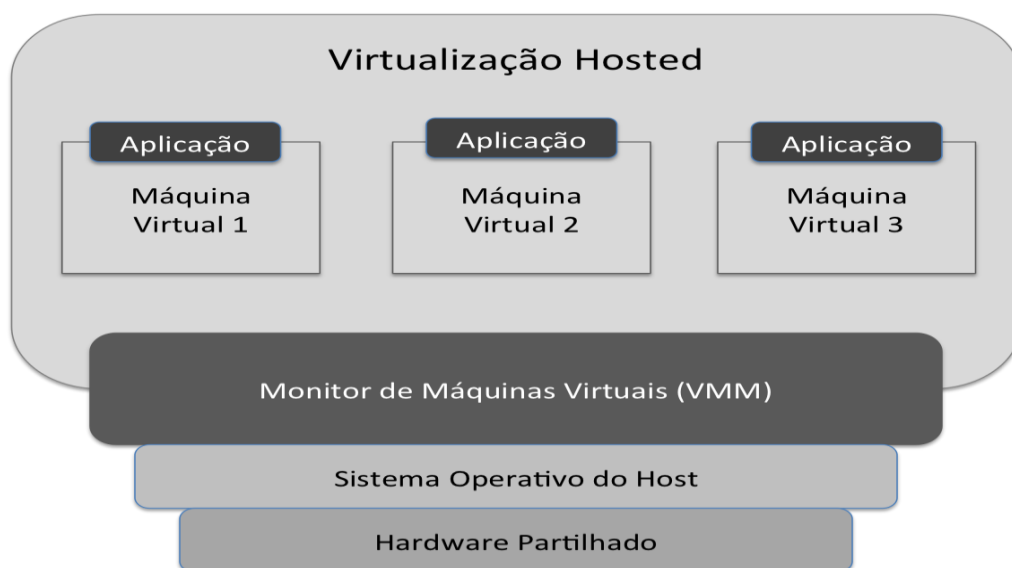


Figura 2.1 - Hypervisor (VMM) do tipo "Hosted"

Cada máquina virtual tem acesso a um número limitado de dispositivos Entrada/Saída (E/S). O VMM fornece uma visão virtual do hardware real para cada máquina virtual, pois o VMM não tem conhecimento da maioria dos dispositivos de E/S, como placas *Peripheral Component Interconnect* (PCI) de aquisição de dados, logo não apresenta estes dispositivos, apresentando apenas dispositivos genéricos como placas de rede e CD-ROM. Muitas soluções de virtualização suportam a funcionalidade do reconhecimento da porta *Universal Serial Bus* (USB), permitindo aos utilizadores o acesso aos dispositivos ligados a essa porta, a partir de máquinas virtuais diferentes [5].

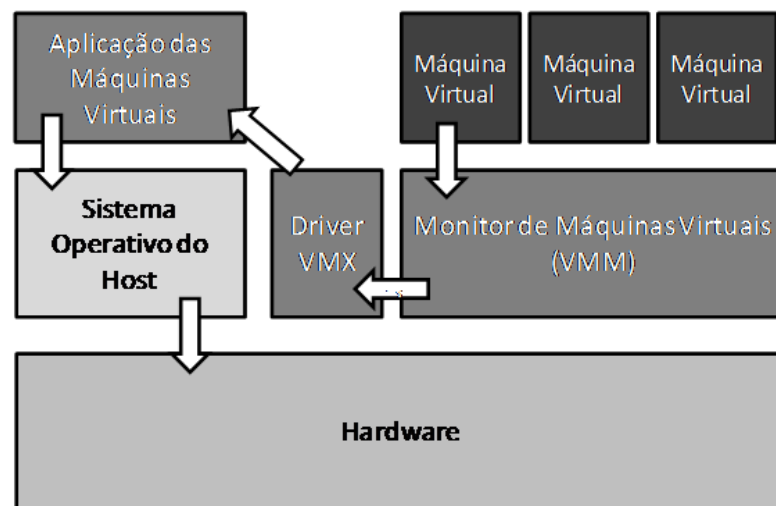


Figura 2.2 - Comunicação entre as máquinas virtuais e o Hardware com VMM "Hosted"

O software VMM é composto por diferentes componentes que fazem a comunicação entre a máquina virtual e os pedidos de E/S através do sistema operativo, como está exemplificado na Figura 2.2. Uma das vantagens desta arquitectura é a facilidade de instalação e configuração.

Na arquitectura *Bare-Metal* o VMM ou *hypervisor* é instalado para permitir a comunicação directa com o hardware, não confiando esta a um sistema operativo. Que se encontra apresentado na Figura 2.3.

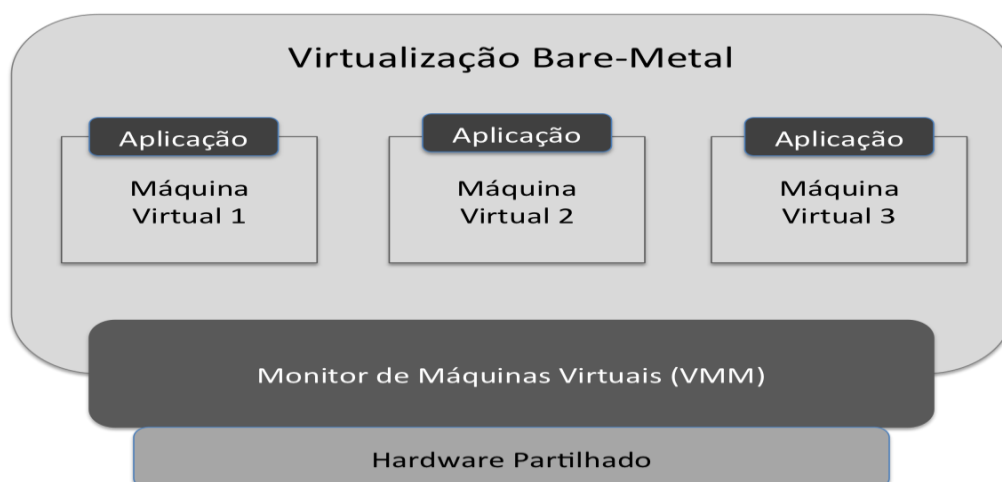


Figura 2.3 - Hypervisor (VMM) do tipo "Bare-Metal"

Este tipo de arquitectura fornece um grande número de opções para o acesso às funções de E/S através das máquinas virtuais. Como não se baseia num sistema operativo do *Host*, um *hypervisor*, usando esta arquitectura, pode comunicar directamente com dispositivos de E/S. O sistema de dispositivos de E/S que é compartilhado pelas máquinas virtuais na arquitectura *Bare-Metal*, deve ser acedido através de um Kernel que faz parte do *hypervisor*, onde podemos ver na Figura 2.4.

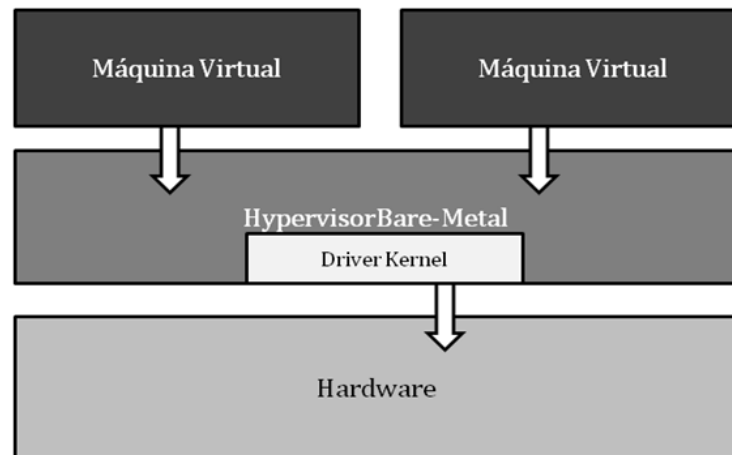


Figura 2.4 - Comunicação entre as máquinas virtuais e o Hardware com VMM “Bare-Metal”

Outra forma utilizada por esta arquitectura, é atribuir dispositivos individuais a máquinas virtuais específicas, o que proporciona o acesso directo dessas máquinas aos dispositivos de E/S atribuídos, que se encontra exemplificado na Figura 2.5. Este processo tem o nome de particionamento e pode melhorar muito o desempenho no que toca ao acesso aos dispositivos de E/S [5].

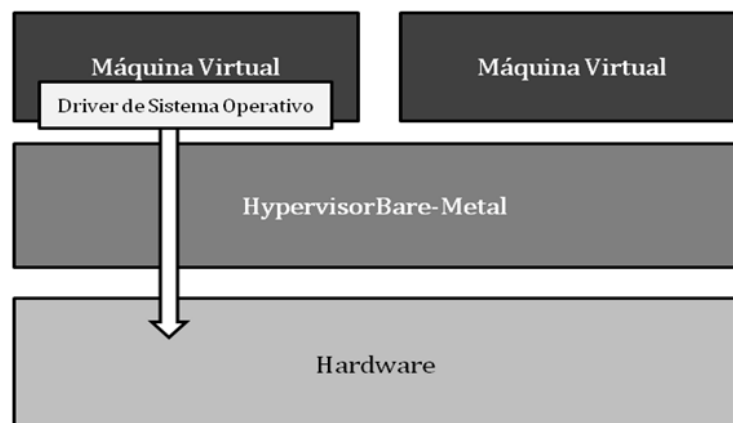


Figura 2.5 - Comunicação entre as máquinas virtuais e o Hardware com VMM “Bare-Metal”

Este tipo de arquitectura apresenta a grande vantagem de melhorar o acesso aos dispositivos E/S através do particionamento, mas estes *hypervisors* são mais difíceis de instalar e de configurar do que os *hypervisors hosted* [5].

Existem vários tipos de virtualização, virtualização completa, paravirtualização ou virtualização assistida pelo hardware[6]. Na virtualização completa, o sistema operativo da máquina virtual “não sabe” que está a ser executado num ambiente virtual, logo o *hypervisor* emula todo o hardware para as máquinas virtuais, necessitando estas apenas dos *devices drivers* para os dispositivos emulados, como se encontra representado na Figura 2.6. A grande vantagem deste tipo de virtualização é ser compatível com todos os sistemas operativos que podem ser executados no hardware emulado, sem que seja necessário efectuar qualquer alteração. Como o sistema operativo não tem conhecimento que se encontra a ser executado como uma máquina virtual, todas as instruções executadas nesse sistema devem ser analisadas pelo VMM e só depois são executadas no hardware, para que se possa evitar operações de hardware que prejudicam outras máquinas virtuais ou mesmo a máquina física [7] [8].

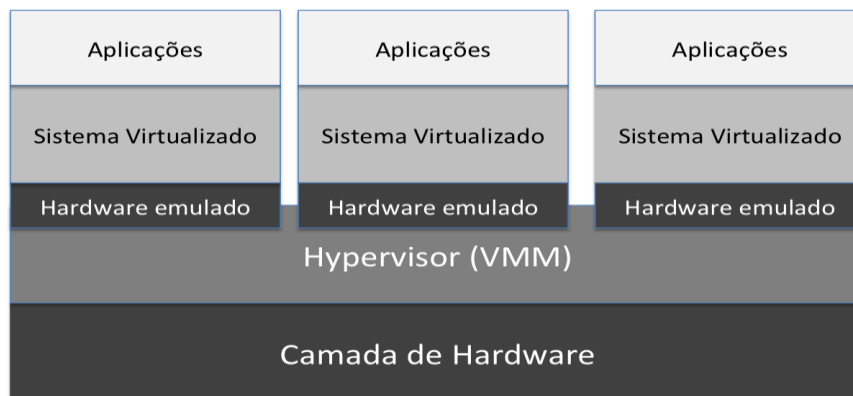


Figura 2.6 - Virtualização Completa

Na paravirtualização o VMM interage com o sistema operativo de uma maneira mais eficiente, mas neste caso o sistema operativo necessita de sofrer modificações para ser compatível, pois caso contrário, seria necessário recorrer à emulação o que iria penalizar o desempenho. Neste caso (Figura 2.7), como o sistema operativo é modificado de maneira a chamar o VMM sempre que executa uma determinada instrução que possa modificar o estado do sistema, o VMM não tem necessidade de testar todas as instruções, o que aumenta o desempenho do mesmo [4] [9].

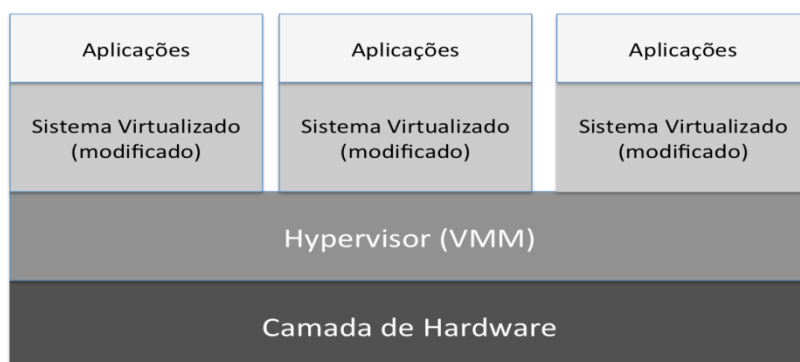


Figura 2.7 - Paravirtualização

Existe também a virtualização assistida pelo hardware, onde os processadores já possuem tecnologias que facilitam o trabalho do *hypervisor*, pois o próprio hardware já possui suporte para a execução das máquinas virtuais em modo isolado, dispensando assim algumas operações necessárias nos outros tipos de virtualização. Exemplos destes processadores são o Intel-VT e o AMD-V [8] [4].

2.4 - Cloud Computing (Computação em Nuvem)

O termo *Cloud* surge devido à utilização da imagem de uma nuvem para representar redes, como a internet. Hoje em dia, o termo *Cloud* já não é utilizado para representar um grande conjunto de ligações, routers, switchs e servidores, mas sim para representar serviços[10].

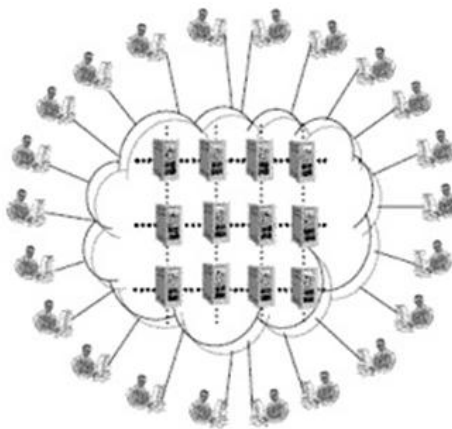


Figura 2.8 - Cloud Computing

Cloud Computing baseia-se na ideia de poder usar variadas ferramentas através da internet, evitando assim a instalação destas no nosso computador, logo o software deixa de estar presente nas máquinas locais e passa a estar em servidores localizados remotamente. O acesso a estes recursos poderá ser feito em qualquer lugar e em qualquer plataforma. A ideia de ter tudo instalado e armazenado no nosso computador torna-se diferente num ambiente corporativo, pois é mais fácil o uso de aplicações disponíveis em servidores remotos e que possam ser utilizadas por qualquer terminal com permissões para tal [12].

Um dos grandes problemas para as organizações é o alto custo com Tecnologia da Informação (TI). A evolução das tecnologias, no âmbito da computação e da comunicação, fornece as condições perfeitas para a utilização do *Cloud Computing*, pois, hoje em dia, é possível obter uma ligação rápida à internet com um baixo custo associado.

O *Cloud Computing* vem assim trazer um leque alargado de vantagens, tais como a possibilidade, grande parte das vezes, de utilizar os recursos *offline*, ou seja, não é necessária a ligação à internet nem à rede, tornando-se os dados criados restritos a essa máquina utilizada, sendo possível mais tarde a partilha pela rede. Disponibiliza ainda a possibilidade do utilizador deste sistema só necessitar de saber como aceder e como o utilizar, pois todas as tarefas relacionadas com desenvolvimento, armazenamento, manutenção, actualização, *backup* e escalonamento ficam ao encargo do fornecedor do

serviço. Estas vantagens proporcionam redução dos custos para o utilizador, pois o controlo total das aplicações tem um preço. O software tem de estar instalado, configurado e depois actualizado com os últimos lançamentos para se usufruir ao máximo dele. Permite também o utilizador ter acesso a variadas aplicações independentemente do seu sistema operativo ou hardware disponível. Outra das características benéficas é a grande flexibilidade, pois no caso de ser preciso mais capacidade de processamento, basta efectuar uma actualização, sem ser necessário a troca de componentes ou até equipamentos inteiros para isso. O uso deste conceito é mais viável do que o uso de máquinas físicas, pois se um servidor parar de funcionar, outros servidores pertencentes à estrutura do serviço continuarão a fornecer o serviço [14].

Contudo, o *Cloud Computing* também traz desvantagens. Desvantagens essas que se prendem com aspectos como a segurança, a confiabilidade e a disponibilidade de serviços, pois como se trata de internet, existem riscos como atrasos e falhas no sistema [15].

2.4.1 - Modelos de Implementação

2.4.1.a - *Software as a Service (SaaS)*

Paralelamente ao *Cloud Computing* existe um conceito relacionado com o mesmo, conceito este que se designa por *Software as a Service* (Software como Serviço). Trata-se de um sistema onde o *software* é oferecido como um serviço. No modelo de *software*, vende-se um produto e a receita é proveniente, ou da sua venda, ou da sua concepção, enquanto no modelo de serviços, a solução é desenvolvida sem qualquer custo para o cliente e depois disponibilizada na internet para que vários clientes possam utilizar esse serviço, obtendo lucro através da taxa ao cliente de um certo valor dependendo da quantidade de tempo ou quantidade de utilização do serviço [16].

2.4.1.b - *Platform as a Service (PaaS)*

Este sistema facilita a implementação de aplicações sem o custo e a complexidade de gerir o hardware associado, fornecendo todas as facilidades necessárias para suportar o desenvolvimento, o suporte e a entrega de aplicações *Web* e serviços disponíveis na internet, sem a necessidade de efectuar *downloads* ou instalações do *software* [16].

2.4.1.c - *Infrastructure as a Service (IaaS)*

Este sistema fornece uma infra-estrutura informática, geralmente através de virtualização, como um serviço [16].

2.5 - IPBrick

A iPortalMais desenvolveu a IPBrick (Sistema Operativo de Rede, *Network Operating Systems*) que tem como concorrente as soluções para servidores da Microsoft, e que portanto tem um potencial enorme, e um mercado à escala mundial. Em simultâneo e no âmbito da comercialização da IPBrick, fornece ainda serviços de suporte, consultadoria, instalação e

configuração. Esta iniciativa deve-se em tudo à capacidade criativa e intelectual dos colaboradores que todos os dias pensam e desenvolvem a IPBrick.

A IPBrick nasceu do desenvolvimento de um conjunto de sistemas operativos e aplicações, num produto integrado, com uma instalação simplificada, facilidade de configuração e um sistema de recuperação rápido em caso de desastre.

Esta Tecnologia surge, face às críticas feitas por parte da Microsoft ao Linux, fundamentando-se na instalação lenta, na dificuldade de configuração e de recuperação em caso de desastre. Desta forma surge a IPBrick como um sistema operativo de instalação automática, configurável através de uma interface *Web* orientada para uma lógica funcional do sistema e o *disaster recovery* através de uma *Pen Drive* onde são guardadas as configurações e em caso de desastre permite recuperar o sistema, como mostra a Figura 2.9.



Figura 2.9 - Solução IPBrick

A IPBrick é um sistema operativo para servidores que abrange três grandes áreas, a intranet, as comunicações e a segurança.

2.5.1 - IPBrick.I

A solução de Intranet, designa-se por IPBrick.I que poderá funcionar como: servidor de *email*, de ficheiros, de domínio, de impressoras e de backup. Disponibiliza ainda ferramentas colaborativas de *groupware* como o *email*, agenda/calendário e livro de contactos que integram com o cliente de *email* (outlook e mozilla thunderbird). Estas ferramentas encontram-se acessíveis através de um browser comum por serem aplicações *Web-based* e poderão ser acedidas através de um PC como de um dispositivo móvel. A IPBrick.I pode ser usada como suporte a aplicações de negócio, como o *Customer Relationship Management* (CRM) sugarCRM, o *Enterprise Resources Management* (ERP) - Gestix - e o Sistema de Gestão Documental - iPortalDoc - que é igualmente desenvolvido pela iPortalMais. Existem outras aplicações que podem ser instaladas na IPBrick, tais como o Moodle - aplicação para a educação, o Zoneminder - aplicação de videovigilância e o Nagios - aplicação de monitorização.

2.5.2 - IPBrick.C

A IPBrick.C a nível das comunicações pode funcionar como um servidor *Virtual Private Network* (VPN), *Web*, *mail relay* e *webmail*. No que toca à segurança permite fazer de firewall, *Intrusion Detection System* (IDS), proxy, e tem ainda Anti-vírus e Anti-spam pré-instalados.

2.5.3 - IPBrick.GT

A IPBrick.GT é um servidor de Comunicações Unificadas por IP, que integra voz, vídeo, fax, *mail*, *Web* e *Istant Messaging* que pode funcionar como *Private Branch Exchange* (PBX) IP ou então como *Media Gateway* da central existente. Esta solução permite ainda a gravação de comunicações para a Gestão Documental - iPortalDoc e também dinamizar os registos automáticos das comunicações no SugarCRM. Esta solução fez com que surgisse o conceito inovador criado pela iPortalMais *Unified Communications over IP- UcolP* - que faz com que o utilizador seja o centro das comunicações, pois o utilizador através do seu endereço de *email* tem acesso à sua página *Web* pessoal, pode realizar *Instant Messaging* e aceder ao email.

Com IPBrick.GT as empresas poderão transformar a sua página *Web*, numa página interactiva colocando lá um *Webphone* e um *Webchat*, poderão integrar esta solução com as aplicações da LAN (SugarCRM e iPortalDoc) como foi acima referido e por fim os utilizadores irão usar o seu endereço de *email* para todas as formas de comunicação. Utilizando o *email* os utilizadores podem ler e enviar *emails*, enviar mensagens (*mail to sms*), ler e enviar faxes (*fax to mail* e *mail to fax*) e ainda ouvir mensagens de *voice-mail*.

A iPortalMais disponibiliza ainda outras soluções dedicadas utilizando a tecnologia IPBrick, como a IPBrick.SOHO - Solução para pequenas e Médias empresas que integra todas as funcionalidades que foram descritas acima. A IPBrick.KAV - Que consiste numa solução de segurança na rede para os acessos *Web*, para a rede, intranet e correio electrónico com o Anti-vírus kaspersky instalado. A IPBrick.LIVE - Solução para *corporate TV's*, permite gerir painéis informativos em zonas públicas bem como *playlists* de filmes, notícias, publicidade e meteorologia; IPBrick.H - Hotspot Controller que consiste numa solução para gerir *hotspots* em zonas públicas, permite a configuração de tarifas a cobrar pelo acesso, páginas consultadas e cartões personalizados e a IPBrick. SCHOOL - um servidor escolar, que fornece um desktop multi-sessão baseado num conjunto de terminais ligeiros a um servidor IPBrick.

Esta tecnologia em alguns casos apenas está disponível em *Appliance* - solução completa, mas a IPBrick.I e IPBrick.C poderá ser instalada em hardwares como Intel, DELL, ASUS, Compaq, HP, Tsunami, IBM, Fugitsu Siemens entre outros [17].

2.6 - Conclusão

De acordo com o que foi apresentado neste Capítulo, será colocado em prática o conceito de Cloud Computing através da virtualização de servidores, para concretizar o objectivo desta dissertação, que consiste na criação de uma Plataforma de gestão de servidores físicos e virtuais. Como foi verificado no ponto - Virtualização de servidores, existem dois tipos

dehypervisor, Bare-Metale Hosted. O tipo de *hypervisor* escolhido para ser utilizado foi o *Bare-Metal*.

A plataforma irá servir para criar e gerir os servidores físicos e máquinas virtuais. Deverá ser capaz de gerir várias *Pools* de servidores, ou seja, definir pequenas redes com conjuntos independentes de servidores, definir a alta disponibilidade dessa *Pool* e ainda criar repositórios para as mesmas. Nestes repositórios deverão constar todos os dados das máquinas virtuais para efeitos de alta disponibilidade, ou seja, caso uma máquina falhe, se o conteúdo dessa máquina se encontrasse no servidor, toda a informação da mesma iria perder-se, assim se estiver num repositório, esta máquina poderá ser recuperada em caso de falha.

Deve possibilitar a alteração dos estados das máquinas, realizando acções como iniciar, reiniciar, desligar e migrar as máquinas virtuais. A alteração dos estados também se deverá aplicar aos *Hosts*, efectuando acções como reiniciar e desligar permitindo também configurar as definições de rede do servidor.

Capítulo 3

Revisão de Conceitos e Tecnologias

Neste capítulo - Revisão de Conceitos e Tecnologias, serão apresentadas ferramentas e as tecnologias já existentes que serão utilizadas como base para a elaboração da plataforma, através de conceitos que servirão para defini-las. A tecnologia é uma ferramenta baseada em computação que permite informatizar vários tipos de processos e objectos.

Este capítulo remete-se assim para a fase de pesquisa, estudo e análise das plataformas e ferramentas existentes de forma a determinar quais serão utilizadas no âmbito desta dissertação. É onde será definida a forma como se irá desenvolver a plataforma de gestão, passando pelos meios utilizados (ferramentas de virtualização) e ferramentas complementares (linguagens de programação). Serão abordadas as plataformas existentes de referência.

3.1 - Ferramentas de Virtualização

3.1.1 - KVM

Kernel-based Virtual Machine(KVM) trata-se de uma infra-estrutura de virtualização do kernel Linux. Este suporta virtualização total usando processadores como o Intel VT ou o AMD-V[13].

Desta forma verifica-se que o KVM necessita de processadores com instruções especiais para o seu funcionamento. Trata-se de uma solução completa baseada em Qemu. A diferença entre o Qemu e o KVM, é que o Qemu é um emulador e o KVM oferece virtualização usando os recursos de virtualização disponíveis nos processadores, enquanto o Qemu emula vários tipos de CPU. O KVM é uma *framework* de virtualização livre que se encontra disponível no kernel Linux desde a versão 2.6.20. Ao contrário de outras tecnologias de virtualização, é possível executar Windows e outros sistemas operativos sem haver necessidade de uma modificação específica. A máquina virtual é tratada como um processo Linux, desta forma é possível utilizar comandos de gestão de processos, tais como ps, kill entre outros de forma a poder controlar a máquina virtual [18] [19].

3.1.2 - VMware

O VMware é desenvolvido pela VMware Inc., localizada em Palo Alto, Califórnia, Estados Unidos. O nome é um jogo de palavras com *Virtual Machine* (máquina virtual), que é o nome técnico dado a um sistema operativo executado pelo VMware[13].

A VMware é um dos fornecedores mais populares na virtualização para a arquitectura x86, fornecendo *softwares* para virtualizar desde ambientes *desktop* como ambientes de *data centers*. Nos produtos fornecidos pela VMware, encontramos os que são executados sob um sistema operativo como o VMware Workstation, VMware Server, VMware Fusion e VMware Player. Também fornecem os que são executados como sistema operativo como o VMware ESX.

O VMware vSphere trata-se de uma plataforma de virtualização de servidores que inclui o *hypervisor* VMware ESX e ferramentas de gestão associadas. Os servidores são geridos através de uma consola de gestão capaz de gerir vários servidores em simultâneo que se designa por vCenter Server [7] [20].

3.1.3 - Xen

O Xen é um software livre de virtualização para as arquitecturas x86, x86-64, IA-32 e IA-64 (*Intel Architecture*, 32-bit e 64-bit respectivamente). Este permite, com o mesmo hardware, a execução de vários sistemas operativos em simultâneo. Foi concebido no âmbito de um projecto na Universidade de Cambridge, liderado pelo Ian Pratt, fundador da XenSource, Inc, esta que mais tarde em 2007 foi comprada pela Citrix System Inc[13].

O Xen é um dos exemplos mais populares de paravirtualização. Este tipo de virtualização tem um ganho de desempenho em relação à virtualização total, pois o sistema operativo visitante é modificado para que todas as tarefas protegidas sejam reencaminhadas para o VMM ao contrário da virtualização total, onde o sistema operativo visitante tenta executar as mesmas tarefas protegidas, mas como estas se encontram no domínio do sistema operativo *host* não podem ser executadas, logo o VMM tem de intervir para realizar a sua execução ou a sua simulação.

Citrix XenServer é uma plataforma de virtualização grátis baseada no Xen *hypervisor* que é um software livre e inclui o XenCenter, uma consola capaz de gerir vários servidores e máquinas virtuais, possui vários *templates* para criar novas máquinas virtuais, possibilita *snapshots*, possibilita migração das máquinas virtuais, através do XenMotion, e oferece suporte para o armazenamento compartilhado[21] [22].

3.1.4 - Hyper-V

O Hyper-V é uma ferramenta de virtualização da Microsoft, Bare Metal que deve ser instalado directamente no hardware eliminando a necessidade de um *Host*. Sendo assim, um produto especializado e independente que contém o *hypervisor* Windows, o modelo de driver do Windows Server e os componentes de virtualização que permitem desenvolver testes e desenvolvimentos, a consolidação de escritórios remotos e a virtualização de estações de trabalho hospedadas. Esta ferramenta tem as características necessárias para o desenvolvimento desta dissertação, contudo, tem certas limitações, ou seja, para poder

usufruir em pleno da ferramenta seria necessário ter sempre o Windows Server 2008, obrigando sempre as empresas a optar por uma tecnologia que não é livre para a resolução dos seus problemas. Outra das limitações é o facto de suportar poucas distribuições Linux a nível de sistemas operativos *guests*, ou seja, para além de apenas suportar Suse Linux Enterprise Server e Red Hat Enterprise Linux, ainda tem um entrave no que toca ao número de processadores virtuais que se pode dedicar a estas máquinas que neste caso é um [13] [23].

3.1.5 - Análise comparativa das ferramentas

Após a apresentação das ferramentas que poderão ser utilizadas, efectuou-se uma análise comparativa das mesmas, com o intuito de seleccionar aquela que iria mais de encontro com os objectivos para a criação da plataforma. Para isso foram enumeradas algumas características que estão visíveis na tabela:

	KVM	Microsoft Hyper-V R2	Citrix XenServer 5.5	Vmware vSphere 4
Hypervisor Bare-Metal	Não	Sim	Sim	Sim
Guests Windows	Sim	Sim	Sim	Sim
Guests Linux	Sim	Sim*	Sim	Sim
Live Migration	Sim	Sim	Sim	Sim **
Alta disponibilidade	Não	Sim	Sim ***	Sim **

Tabela 3.1- Comparação das Plataformas

*Apenas SUSE Linux Enterprise Server e Red Hat Enterprise Linux.

**Versão VMware vSphere Advanced

*** Versão Citrix XenServer Essentials

Com base na tabela apresentada, verifica-se que a nível de *hypervisor* o KVM[19] não é *bare-metal*, isto é, tecnologia que é instalada directamente no servidor físico, sem ser necessário ter um sistema operativo instalado, como também não disponibiliza a Alta disponibilidade ao contrário das restantes ferramentas. Com excepção ao KVM, todas as ferramentas são do tipo *bare-metal* e contêm Alta disponibilidade, contudo o Xenserver[24] e o VMware vSphere[25] apenas a disponibilizam em versões específicas dos produtos que estão designadas na tabela acima. Todas as ferramentas permitem *Live Migration* e suportam *Guests* Windows e Linux, tirando o VMware vSphere que apenas disponibiliza *Live Migration* na versão VMware vSphere Advanced, e o Hyper-V R2[23] que a nível de *Guests* Linux suportados, apenas suporta SUSE Linux Enterprise Server e RED Hat Enterprise Linux.

Desta forma, as ferramentas que nesta fase foram excluídas foram o Hyper-V R2 e o KVM. O Hyper-V por estar sempre muito dependente do Windows Server 2008, como é descrito em cima quando é apresentada a ferramenta e por suportar poucas distribuições Linux, visto que é um ponto importante, pois as máquinas virtuais que irão ser criadas serão máquinas Linux. O KVM foi excluído por não ser do tipo Bare-Metal, que se trata de um requisito necessário e por não oferecer o serviço de Alta disponibilidade.

3.1.6 - XenServer vs. VMware vSphere

Devido ao facto do KVM e do Hyper-V não irem de encontro com aquilo que é pretendido resta-nos as ferramentas XenServer e VMware vSphere. Desta forma, irá ser comparado ambas as ferramentas em pontos cruciais para o desenvolvimento do trabalho e verificar qual delas se adequa mais ao mesmo.

A análise irá se basear em características como a arquitectura do sistema; as características ligadas à instalação, configuração e administração do sistema; quais os sistemas operativos suportados pelos sistemas; se permite a integração com unidades de armazenamento; a possibilidade de *disaster recovery*; a alta disponibilidade e a manutenção do sistema[24] [26].

3.1.6.a - Arquitectura do Sistema

Ambas as ferramentas são *hypervisors* do tipo 1 - *bare-metal* - ou seja, são tecnologias que são instaladas directamente no servidor físico, sem ser necessário ter um sistema operativo instalado, pois estas funcionam como tal. Estas tecnologias oferecem um desempenho significativamente melhor do que soluções que necessitam ser instaladas sob um sistema operativo, tais como VMware GSX Server (GSX) e o Microsoft Virtual Server.

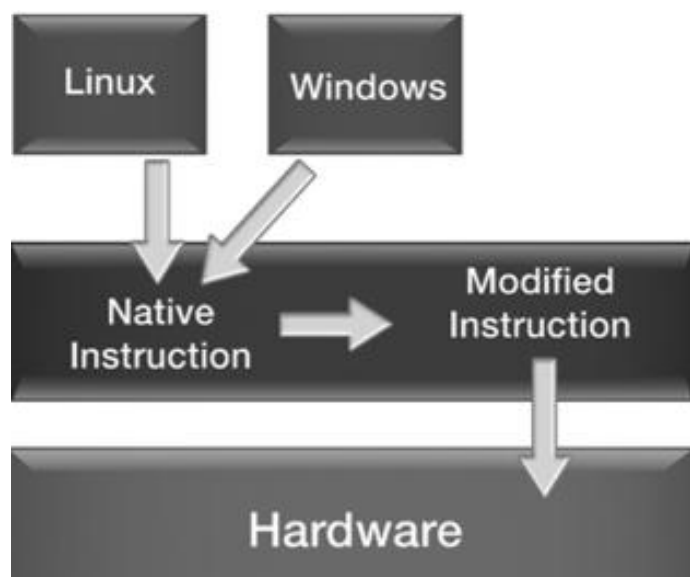


Figura 3.1 - Tradução binária efectuada pelo VMware ESX [24]

O VMware ESX faz parte da primeira geração das arquitecturas de virtualização *bare-metal*. A abordagem que o ESX faz à virtualização pode ser demonstrada como uma tradução binária tal como representa a Figura 3.1. Exemplificando, ao fazer um pedido de suspensão do sistema operativo para o processador, aquilo que se irá suceder é a suspensão de uma máquina virtual específica, em vez de ser suspenso todo o sistema, libertando assim recursos para as outras máquinas virtuais, logo o ESX leva o sistema operativo convidado a pensar que está a ser executado num hardware físico. Como a primeira camada que interage com o hardware é o *software* o VMware tem feito investimentos consideráveis no desenvolvimento de *drivers* próprios para suportar a variedade de hardware contida nos servidores comerciais.

A arquitectura do XenServer é bastante diferente da do VMware. O XenServer é considerado a base para quase duas dezenas de produtos de virtualização comerciais, alimentando assim o maior motor do mundo de virtualização - *Amazon Elastic Compute Cloud*. Enquanto o VMware utiliza a tradução binária o XenServer usa uma combinação de paravirtualização e virtualização assistida por hardware. Ao contrário do VMware o XenServer foi a primeira solução de paravirtualização que permite ao sistema operativo convidado estar plenamente consciente que se encontra a ser executado sob um hardware virtualizado. Assim conseguimos ter um tipo de virtualização cujo desempenho é mais optimizado. O XenServer suporta distribuições de Linux como o Red Hat Enterprise Linux; o Novell SUSE; o Debian; o Oracle Enterprise Linux e o CentOS.

A VMware começou a adoptar a paravirtualização como a tecnologia base da *Virtual Machine Interface* (VMI) do ESX. Consequentemente através da análise do desempenho da VMI a VMware acredita que a paravirtualização é superior à tradução binária. Contudo apenas alguns sistemas operacionais Linux estão habilitados a executar VMI em vSphere, como algumas versões do SUSE e algumas versões Fedora.

Por outro lado a abordagem do XenServer para *drivers* de dispositivos é significativamente diferente da abordagem da VMware, como podemos verificar na Figura 3.2.

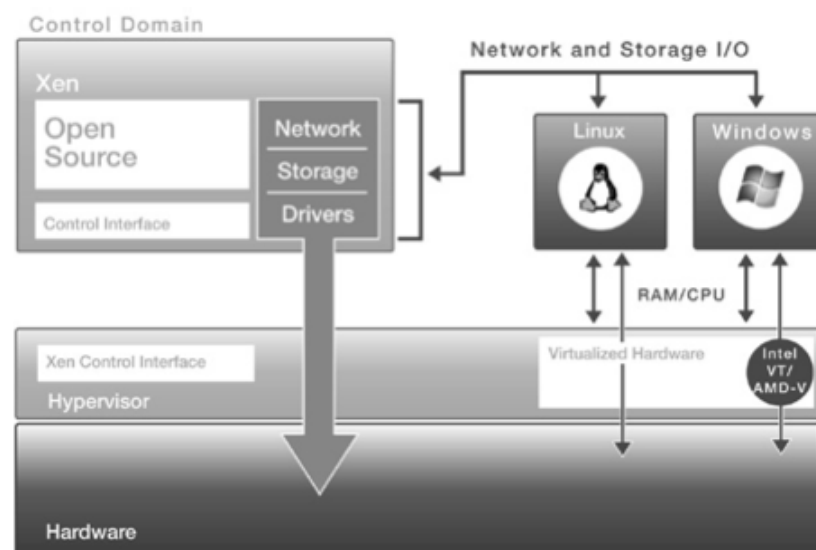


Figura 3.2 - Arquitectura do Hypervisor do XenServer [24]

3.1.6.b - Instalação, configuração e administração do sistema

O processo de instalação do XenServer é muito simples. O XenServer pode ser instalado na máquina através de um CD-ROM ou a partir da rede. O XenCenter é uma consola de administração que possui uma interface gráfica muito simples e pode ser instalada em qualquer PC (*Personal Computer*) com ambiente Windows ou servidor. As configurações do sistema são mantidas num sistema de armazenamento interno de dados e são replicadas por todos os servidores que se encontram a ser geridos em conjunto, formando uma *Pool* de recursos para garantir a alta disponibilidade dos sistemas de gestão do núcleo de serviço, não recorrendo a um servidor de base de dados externo para a suposta gestão do núcleo.

Ao par do XenServer, o VMware ESX é instalado na máquina, contudo utiliza o vCenter Server para gerir os vários servidores e as suas respectivas configurações. O vCenter Server é executado como um serviço do Windows num servidor de gestão em separado, requerendo assim uma base de dados externa para armazenar e gerir as configurações do núcleo. Para redundância e alta disponibilidade dos serviços de gestão do núcleo a VMware recomenda a adição do seu software de *clustering* intitulado de vCenter Server Heartbeat.

3.1.6.c - Sistemas operativos suportados pelos sistemas

Tanto o VMware como o Xenserver suportam os sistemas operativos mais populares como Windows e Linux, ou seja, Windows XP, Windows Vista, Windows 2000 Server, Windows Server 2003, Windows Server 2008, Red Hat Linux, SUSE Linux entre outros.

3.1.6.d - Integração com unidades de armazenamento

O armazenamento é considerado uma das coisas mais importantes para a implementação de servidores virtuais. Ambas as soluções fornecem suporte de armazenamento de máquinas virtuais em discos locais, em *Storage Area Network* (SAN) ou *Network-Attached Storage* (NAS). O uso destas tecnologias para o armazenamento de dados é necessário para mais tarde usufruir de recursos como alta disponibilidade e *live migration*.

O XenServer tem uma abordagem completamente diferente para o armazenamento, ou seja, não impõe o seu próprio sistema de arquivo no sistema de armazenamento, como se verifica no VMware que utiliza o *Virtual Machine File System* (VMFS), mas sim aproveita as capacidades de armazenamento nativo. Por exemplo com um arquivo baseado em sistema de armazenamento partilhado, tais como *Network File System* (NFS), as máquinas virtuais do XenServer são armazenadas directamente usando o formato Microsoft *Virtual Hard Disk* (VHD).

3.1.6.e - Disaster Recovery

O *Disaster recovery* implica a duplicação da infra-estrutura do servidor virtual e de dados em instalações remotas para a sua recuperação no caso de algum evento impossibilitar o funcionamento da infra-estrutura inicial.

Tanto no XenServer como no VMware é possível disponibilizar as máquinas virtuais através do local primário ou do local que foi destinado para o *Disaster recovery*. Em ambos os casos a solução de virtualização assenta na tecnologia de replicação SAN para manter os arquivos e as configurações actuais da máquina virtual num local de *backup*. A VMware oferece o *Site Recovery Manager*, um produto que é essencialmente um *workflow* de *Disaster recovery* para os sistemas virtualizados com ESX. Da mesma forma, o XenServer inclui o Workflow Studio para complementar o XenServer na automação de *Disaster recovery* tanto para servidores virtuais como físicos.

3.1.6.f - Alta disponibilidade

A alta disponibilidade permite que máquinas virtuais possam ser reiniciadas numa outra máquina física caso a máquina original falhe a executar a máquina virtual de forma inesperada. Isto pode reduzir o tempo de indisponibilidade do sistema e a necessidade de

intervenção administrativa. A VMware e o XenServer têm características de alta disponibilidade que oferecem políticas de gestão do comportamento das máquinas virtuais após falha da máquina que as acolhe. A VMware inclui um sistema de tolerância de falhas, uma característica que mantém espelhado instâncias de máquinas virtuais que se encontram a ser executadas em máquinas distintas. No caso de falha da máquina que as suporta é espelhada a instância mantendo assim a continuidade de trabalho.

3.1.6.g - Manutenção do Sistema

Tanto o XenServer como o VMware requerem alguma manutenção periódica para aplicar actualizações de software. Porque ambas as soluções oferecem recursos de *live migration* actualizações para o *hypervisor*, que podem ser realizadas sem incorrer em qualquer tempo de inactividade para as máquinas virtuais[24] [26].

No decorrer da análise da comparação destas duas tecnologias, foi efectuado um levantamento mais funcional das características das mesmas bem como os custos envolvidos [24].

22 Ferramentas de Virtualização

	XenServer 5.5	VMware vSphere 4.0
Pricing model	Per server: No restriction on number of processors	Per processor: Penalizes use of more powerful servers and higher consolidation ratios
Bare-metal deployment	Free	Yes
P2V and V2V migration tools	Free	Yes
Multi-server management	Free	vCenter Server \$6-\$8k extra
Shared storage support	Free	Yes: Standard Edition and higher
Resource pools	Free	Yes: Standard Edition and higher
VM snapshots	Free	Yes: Standard Edition and higher
Real-time performance monitoring	Free	Yes: Standard Edition and higher
Live motion	Free	Yes: Advanced Edition and higher
VM backup enablement	Free	Yes: Advanced Edition and higher
8 vCPU support	Free	Yes: Enterprise Plus Edition only
Disaster recovery enablement	Free	Site Recovery Manager: \$2k extra, per processor
High performance for Windows	Free	Virtualization Review called VMware "the pokiest" and XenServer "the Porsche."
High performance for Linux	Free	VMware has limited support for paravirtualization
High performance for XenApp	Free	Tolly Group: XenServer beats VMware by 41%
Maintenance requirements	5 patches for XenServer 4.0 since release in August 2007	Hundreds of patches for VI 3.5 since release in December 2007
VM high availability	Yes, Enterprise Edition	Yes: Standard Edition and higher
High availability of core management services	Yes, Enterprise Edition	vCenter Server Heartbeat \$12-\$16k extra
Historical performance monitoring	Yes, Enterprise Edition	Yes: Standard Edition and higher
Administrator alerts	Yes, Enterprise Edition	Yes: Standard Edition and higher
StorageLink	Yes, Enterprise Edition	Nothing comparable
Provisioning services for VMs	Yes, Enterprise Edition	Nothing comparable
VM load balancing	Yes, Enterprise Edition	Yes: Enterprise Edition and higher
Lab management	Yes, Platinum Edition	Lab Manager: \$1,500+ extra, per processor
Stage management	Yes, Platinum Edition	Stage Manager: \$1,500+ extra, per processor
XenCenter role-based administration	Coming soon ¹	Yes; Standard Edition and higher
Memory overcommit	Coming soon ¹	Yes; Standard Edition and higher
VM fault tolerance	Products from Marathon Technologies and Stratus	Yes: Advanced Edition and higher (for VMs with 1 vCPU)

Tabela 3.2 - Comparação das funcionalidades oferecidas pelo XenServer e pelo VMware vSphere [24]

5 Servers – XenServer 5.5		5 Servers – VMware vSphere 4.0*	
Capabilities: Multi-server management, resource pools, XenMotion, HA, workload balancing		Capabilities: Multi-server management, resource pools, vMotion, HA, DRS	
5x Citrix XenServer	Free	vCenter Server + 3 Year Gold Support:	\$7,670
5x Essentials for XenServer, Enterprise:	\$12,500	vCenter Server Heartbeat, 3 Yr Gold Support:	\$15,347
(Includes 1 year Subscription Advantage)	\$1,500 x 3 years	vSphere Enterprise, 3 Yr Gold Support :	\$88,304
Technical Support:	= \$4,500	(20 processors)	
SA Renewal:	\$325/host x 2 years = \$3,250		
Total 3-year cost:	\$20,250	Total 3-year cost:	\$111,321

Tabela 3.3- Comparação dos custos associados ao XenServer e ao VMware vSphere [24]

3.2 - Linguagens de Programação

3.2.1 - PHP

Hypertext Preprocessor (PHP) é uma linguagem de programação muito utilizada para gerar conteúdo dinâmico na *World Wide Web*. Esta surgiu por volta de 1995, criada por Rasmus Lerdorf, com o nome de *Personal Home Page Tools*. Mais tarde em 1997, Zeev Suraski e Andi Gutmans desenvolveram o PHP 3, a primeira versão que se aproxima do PHP que conhecemos, apresentando como maior característica a grande capacidade de extensibilidade. Hoje em dia, a versão utilizada é a versão PHP 5, que foi lançada em 2004, que introduziu um novo modelo de orientação a objectos, melhorando muito esta potencialidade em relação à versão anterior. Trata-se de uma linguagem muito parecida, em termos de sintaxe e funções, com a linguagem C e C++. Esta simplifica a interacção das páginas desenvolvidas com as bases de dados pretendidas e trata-se de uma linguagem modular, o que é ideal para o uso em servidores *Web*[13] [27].

3.2.2 - Javascript

Javascript é uma linguagem de programação que foi criada pela Netscape em 1995. Apesar de ser muito semelhante com Java é muito diferente quanto ao uso, sendo o Javascript mais orientado para páginas *Web*. Trata-se de uma linguagem orientada a objectos, muito usada em páginas *Web* no lado do cliente, que leva a uma melhor apresentação e interactividade das páginas *Web*. O *browser* de quem acede a páginas que usam este tipo de linguagem, é o elemento que fica encarregue de interpretar e executar estes *scripts*, sem ter de apelar aos recursos de um servidor, o que aumenta a rapidez do processo[13] [28].

3.2.3 - HTML

HyperText Markup Language (HTML) trata-se de uma linguagem usada para criar páginas *Web* e foi desenvolvido por Tim Berners-Lee em 1990. O objectivo inicial era tornar possível a troca de informação e documentação de pesquisas, entre cientistas. Este possibilita apresentar informações através da internet. Actualmente encontra-se a ser desenvolvido o *Extensible Hypertext Markup Language* (XHTML), que usa uma sintaxe mais rigorosa e menos ambígua para que o código HTML se torne mais simples para ser processado[13] [29].

3.3 - Protocolos

3.3.1 - XML-RPC

O *Extensible Markup Language - Remote Procedure Call* (XML-RPC) foi criado por Dave Winer na UserLand Software em 1995 na Microsoft. No entanto, a Microsoft apoiou pouco o seu desenvolvimento adicionando poucas funcionalidades.

O XML-RPC é um protocolo de RPC codificado em XML, é um protocolo simples, definido com poucas linhas de códigos ao contrário da maioria dos sistemas de RPC, onde os documentos padrões são frequentemente milhares de páginas e exige apoio de softwares para serem usados.

O XML-RPC funciona através de pedidos *HyperText Transfer Protocol* (HTTP) de um cliente para um servidor que contenha este protocolo implementado. O cliente neste caso não passa de uma simples chamada de um método de um sistema remoto. Podem ser passados vários parâmetros de entrada de forma a obter o(s) valor(es) pretendidos. O XML-RPC pode ser usado para transportar objectos ou estruturas como parâmetros de entrada ou saída [13] [30].

3.3.2 - iSCSI

iSCSI - “eye-scuzy”, abreviatura de *Internet Small Computer System Interface*, é um protocolo de armazenamento baseado na rede *Internet Protocol* (IP). É utilizado para facilitar a transferência de dados na intranet. Este protocolo pode ser utilizado para transferir dados nas *Local Area Networks* (LANs), *Wide Area Networks* (WANs) ou pela Internet. É um protocolo muito usado nas SANs (*Storage Area Networks*). Ao contrário do *Fibre Channel*, este utiliza a infraestrutura da rede já existente sem ser necessário material especializado.

3.3.2.a - iSCSITarget

O protocolo iSCSI refere-se a um recurso de armazenamento localizado num servidor de iSCSI, normalmente um dos muitos nós de armazenamento iSCSI que se encontram no servidor, como uma *target*. Uma *target* iSCSI geralmente representa um disco rígido que serve como repositório e que funciona nas redes Ethernet. Uma *Target* iSCSI não deve ser confundida com o termo iSCSI, pois este trata-se de um protocolo e não de uma instância de armazenamento num servidor.

3.3.2.b - Logical Unit Number

Na terminologia SCSI, *Logical Unit Number* - LUN representa o número de unidade lógica. Um LUN trata-se de um dispositivo SCSI individual e endereçável que faz parte de um dispositivo SCSI físico (*target*). Num ambiente iSCSI, os LUNs são unidades de disco numeradas.

3.3.2.c - Endereço

O iSCSI aceita três formatos para o nome: *iSCSI Qualified Name* (IQN) com o formato “iqn.yyyy-mm.{nome de domínio invertido}: texto opcional para melhorar organização de recursos” como por exemplo ‘iqn.2001-04.com.acme:storage.tape.sys1.xyz’. *Extended Unique Identifier* (EUI), onde o formato é “eui.(endereço EUI-64bit)” como demonstra o exemplo ‘eui.02004567A425678D’ e por ultimo *T11Network Address Authority*(NAA)que apresenta o formato naa.(identificador NAA 64 ou 128 bits) como é apresentado no exemplo ‘naa.52004567BA64678D’. O mais frequente dos três é o IQN.

3.3.2.d - Segurança

As iSCSI *targets* podem apresentar autenticação usando o protocolo *Challenge-Handshake Authentication Protocol* (CHAP). O protocolo CHAP é vulnerável a ataques de dicionário, *spoofing*, ou ataques de reflexão mas é possível prevenir a maioria destes ataques, respeitando as devidas regras de utilização de CHAP em iSCSI[13] [31].

3.3.3 - VNC

Desenvolvido originalmente pela Olivetti & Oracle Research Labs em Cambridge, Inglaterra, que foi posteriormente adquirida pela *American Telephone and Telegraph*(AT&T) e, finalmente, fecharam, *Virtual Network Computing*(VNC) tem evoluído como o software de controlo remoto mais utilizado. Isto é devido ao facto de ser gratuito, à sua disponibilidade para um grande número de diferentes sistemas operativos, a sua simplicidade, robustez e boa compatibilidade com versões anteriores. O suporte multi-plataforma é especialmente importante para os administradores de rede, pois permite administrar quase todos os sistemas a partir de uma única ferramenta. Existe também uma applet Java, que possibilita o acesso a qualquer computador localizado remotamente acedendo a um browser, sem ser necessário a instalação de software.

A versão VNC, original da AT&T, não é muito usada, porque existem diversos produtos com melhorias significativas disponíveis, como o RealVNC, VNC tight e UltraVNC. UltraVNC é o mais usado neste momento.

O VNC foi originalmente desenvolvido para ser utilizado dentro de uma LAN, logo existe questões de segurança quando utilizado através da internet. Quando se pretende aceder a um computador localizado remotamente deve-se considerar um túnel VPN ou proteger todo o tráfego do VNC através do uso de *Secure Sockets Layer* (SSL)[32] [33].

3.4 - PostgreSQL

PostgreSQL trata-se de uma base de dados *open-source* e foi originalmente desenvolvido no Departamento de Ciência de Computação da Universidade de Berkeley. Este apresenta mais de 15 anos de desenvolvimento activo e uma arquitectura comprovada que lhe valeu a reputação de confiabilidade, integridade de dados e exactidão. Pode ser utilizado em todos os principais sistemas operativos, incluindo Linux, UNIX (*Advanced Interactive Executive* (AIX), *Berkeley Software Distribution* (BSD), *Hewlett-Packard UNIX* (HP-UX), Mac OS X,

Solaris, Tru64) e Windows. Este possui suporte para chaves estrangeiras, *joins*, vistas e *triggers*. Inclui vários tipos de dados como INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL, e TIMESTAMP suportando também o armazenamento de imagens, sons e vídeo. Perante estas vantagens acima descritas o sistema operativo IPBrick utiliza esta base de dados como principal do sistema[34] [35].

3.5 - Plataformas existentes

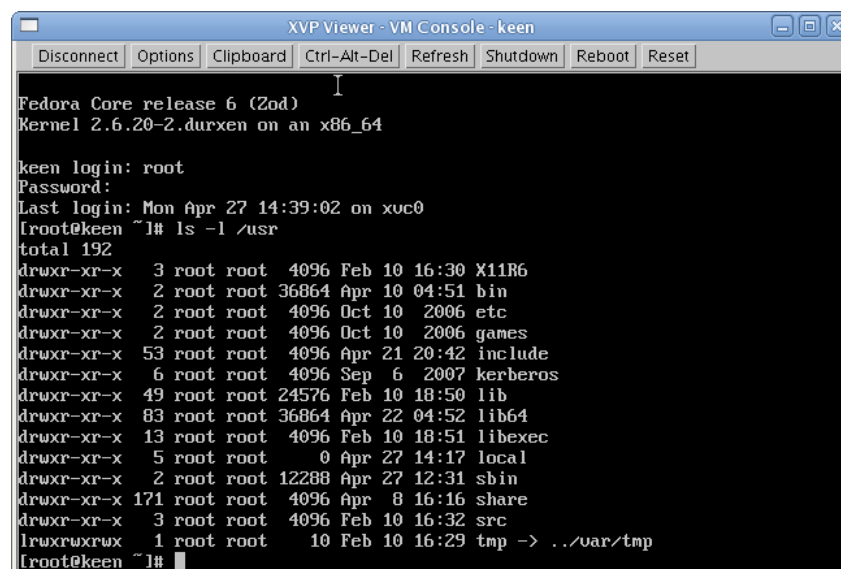
3.5.1 - XVP

O XVP engloba um conjunto de programas de código aberto com o objectivo de gerir várias máquinas virtuais presentes no Citrix XenServer e foi desenvolvida por Colin Dean. Usando esta interface WEB, temos acesso às consolas das máquinas virtuais, podemos visualizar as propriedades das máquinas virtuais, executar operações nas máquinas virtuais (iniciar, reiniciar, desligar, suspender, entre outras) e migrar máquinas virtuais entre *Hosts* numa *Pool*. Esta plataforma é suportada em browsers presentes nos sistemas Linux, Mac OS X e Windows. Os vários componentes que a compõem são: *xvp*, *xvpviewer* e o *xvpweb*.

O *xvp* (*Xen VNC Proxy*) é um servidor proxy que oferece um acesso protegido por *password* às consolas das máquinas virtuais que se encontram em funcionamento no Citrix XenServer.

Pode ser utilizado para ter acesso às consolas através de qualquer sistema operativo que tenha um cliente VNC, e não apenas do Windows como o XenCenter. Usando as *passwords* do VNC, podemos fornecer um acesso selectivo às consolas das máquinas virtuais. O XVP encontra-se escrito em C e funciona nas distribuições mais recentes do Linux.

O *xvpviewer* trata-se de um cliente VNC feito em Java. É baseado no TightVNC viewer, mas com algumas alterações para permitir certas operações nas máquinas virtuais como desligar e reiniciar e para suportar o *scroll* do rato. É compatível com Linux, Mac OS X e Windows como podemos confirmar nas figuras 3.3 e 3.4.



```

XVP Viewer - VM Console - keen
[Disconnect] [Options] [Clipboard] [Ctrl-Alt-Del] [Refresh] [Shutdown] [Reboot] [Reset]

Fedora Core release 6 (Zod)
Kernel 2.6.20-2.durxen on an x86_64

keen login: root
Password:
Last login: Mon Apr 27 14:39:02 on xvc0
[root@keen ~]# ls -l /usr
total 192
drwxr-xr-x  3 root root  4096 Feb 10 16:30 X11R6
drwxr-xr-x  2 root root 36864 Apr 10 04:51 bin
drwxr-xr-x  2 root root  4096 Oct 10 2006 etc
drwxr-xr-x  2 root root  4096 Oct 10 2006 games
drwxr-xr-x 53 root root  4096 Apr 21 20:42 include
drwxr-xr-x  6 root root  4096 Sep  6 2007 kerberos
drwxr-xr-x 49 root root 24576 Feb 10 18:50 lib
drwxr-xr-x 83 root root 36864 Apr 22 04:52 lib64
drwxr-xr-x 13 root root  4096 Feb 10 18:51 libexec
drwxr-xr-x  5 root root    0 Apr 27 14:17 local
drwxr-xr-x  2 root root 12288 Apr 27 12:31 sbin
drwxr-xr-x 171 root root  4096 Apr  8 16:16 share
drwxr-xr-x  3 root root  4096 Feb 10 16:32 src
lrwxrwxrwx  1 root root    10 Feb 10 16:29 tmp -> ../var/tmp
[root@keen ~]#
  
```

Figura 3.3 - Máquina Virtual Linux num sistema Linux

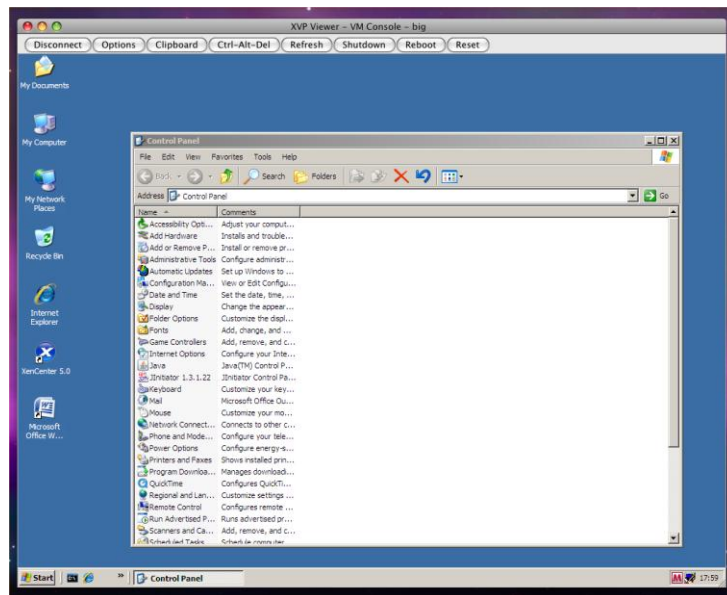


Figura 3.4 - Máquina Virtual Windows num MAC OS X

O xvpweb é um *Web-based front end* que fornece uma visão geral do estado das *Pools*, *Hosts* e Máquinas Virtuais que se encontram no XenServer, apresentando várias funcionalidades que permitem a visualização das consolas das máquinas virtuais e a realização de operações como iniciar, reiniciar, suspender e migrar máquinas virtuais como podemos visualizar nas figuras 3.5 e 3.6.

Este usa o xvp (correndo na mesma máquina que o servidor *Web*) e o xvpviewer, utilizado como uma applet no xvpweb.

O xvpweb utiliza HTML, *Cascading Style Sheet* (CSS), PHP, Javascript e usa a biblioteca de Javascript jQuery. Para ser executado necessita de um servidor *Web* Apache no Linux, com PHP, CURL e mcrypt.

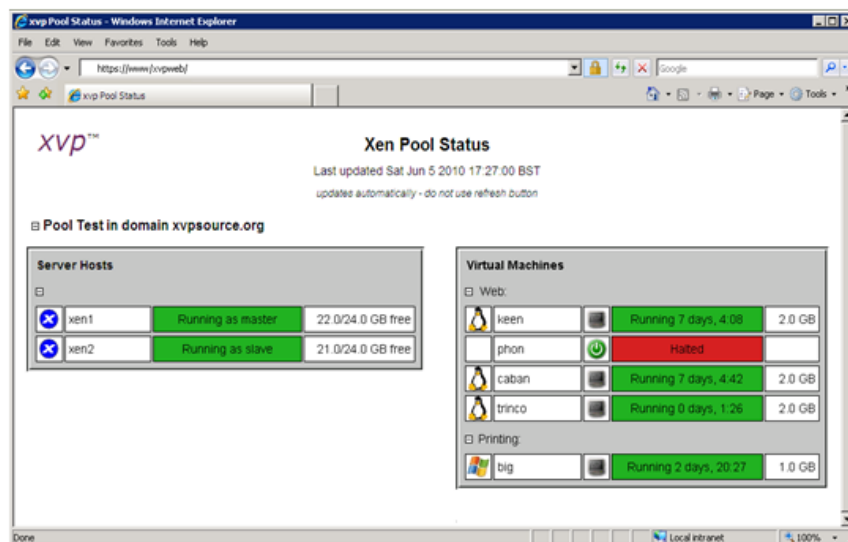


Figura 3.5 - XVPWEB



Figura 3.6 - Acedendo às propriedades de uma máquina virtual no XVPWEB

O xvpdiscover é um pequeno programa, que funciona em Linux, que pode escrever uma configuração no xvp.conf adequada para o xvp e para o xvpweb.

Faz isso conectando-se a um dos servidores XenServer e consultando os detalhes da *Pool*, dos *Hosts* e das máquinas virtuais e criar o ficheiro de configuração consoante a informação que conseguiu recolher[36].

3.5.2 - XenCenter

A aplicação XenCenter serve para gestão remota dos servidores XenServer, contudo apenas funciona no Windows Server 2003, Windows Server 2008, Windows XP ou Windows Vista.

É uma plataforma que proporciona a inserção de novos servidores XenServer e a sua gestão na mesma. É possível adicionar servidores a *Pools*, editar configurações e de rede dos servidores bem como os seus atributos. A nível das *Pools* permite a configuração de alta disponibilidade definindo quais as políticas de gestão das máquinas. Permite ainda a criação e gestão de máquinas virtuais, como podemos verificar na Figura 3.7.

Ao contrário de outras plataformas de gestão, o XenCenter distribui a gestão dos dados entre os servidores numa *Pool* de recursos para garantir que não haja um único ponto de falha de gestão. Se um servidor de gestão falhar, qualquer outro servidor na *Pool* poderá assumir o papel de gestor.

O Xencenter tem como requisitos a nível de Sistema Operativo - Windows XP, Windows Server 2003, Windows Server 2008 e Windows Vista SP1e SP2; Framework -.NET framework - versão 2.0 ou superior; relativamente ao CPU - 750 MHz mínimo, 1GHz ou superior recomendado; no que toca à memória RAM - 1 GB mínimo, 2 GB ou mais recomendado, Disco - 100 MBmínimo e Interface de rede - 100Mbps/sou mais rápida.

O Xencenter é uma aplicação Windows que serve de cliente. Deve ser instalada numa máquina remota que esteja na mesma rede que o servidor XenServer[37] [38].

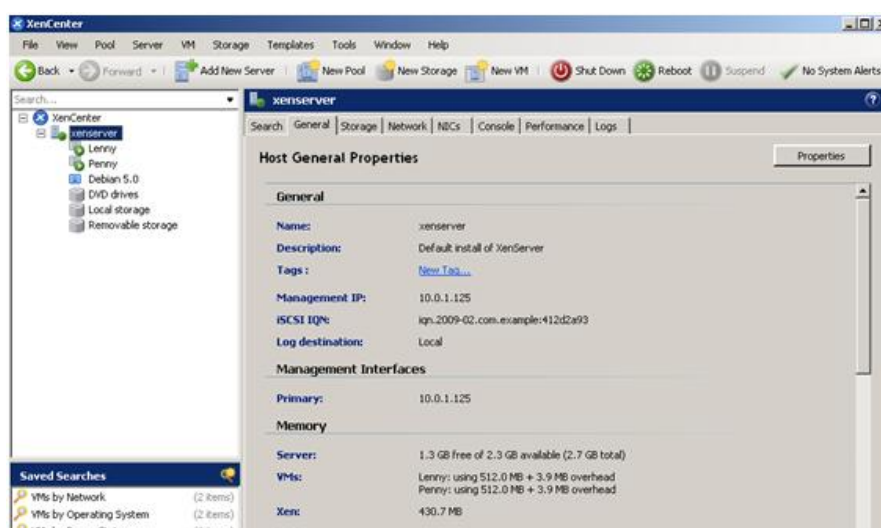


Figura 3.7 - XenCenter

3.6 - Conclusão

Com o intuito de desenvolver uma nova plataforma, de acordo com o que foi referido no capítulo 2, será utilizada a virtualização como forma de concretizar o conceito de *Cloud Computing* para a IPBrick. Desta forma, este capítulo foi iniciado com algumas ferramentas existentes de virtualização tais como o KVM, VMware, Xen e Hyper-V. Como foi explicado o KVM e o Hyper-V não foram tidos em conta devido ao facto de não irem de encontro com os requisitos como foi explicado neste capítulo, na análise comparativa das ferramentas. Foi efectuada uma análise das soluções VMware vSphere e XenServer, de forma a comparar ambas e verificar qual a mais indicada para ser utilizada no desenvolvimento da plataforma *Web* para a gestão de servidores físicos e máquinas virtuais.

As linguagens de programação (PHP, Javascript e HTML), passíveis de serem utilizadas, também foram apresentadas bem como os protocolos (XML-RPC e iSCSI e VNC). Após ter sido referenciado as linguagens de programação e protocolos a utilizar, foi feita referência a duas plataformas existentes que complementam o conceito da plataforma a criar dentro dos objectivos desta dissertação, ou seja, a plataforma que irá ser criada deve corresponder às funcionalidades que as plataformas XVP e XenCenter apresentam. É por estas plataformas não satisfazerem as necessidades das empresas, no que toca à gestão e configuração de *Pools*, *Hosts* e máquinas virtuais, que emerge a necessidade de ser desenvolvida uma plataforma que o faça. O XenCenter funciona apenas em sistemas operativos Windows e o XVP, apesar de ser uma plataforma *Web*, não efectua uma gestão que se enquadra nas necessidades das empresas, pois não fornece as acções necessárias.

Em suma, através das ferramentas abordadas, deverá ser possível o desenvolvimento da plataforma IPBrick Cloud Environment para a gestão dos servidores IPBrick. Para o efeito foi escolhida a ferramenta XenServer pelo facto de ser uma versão grátis com muitas mais funcionalidades do que a da VMware e dentro das opções que têm de ser pagas é aquela que fica mais económica. Foi escolhida também pelo facto de ser uma tecnologia que deriva de tecnologias *open-source*, existindo muita mais informação disponibilizada sobre a ferramenta.

Capítulo 4

Desenvolvimento da Plataforma

Neste capítulo - Desenvolvimento da Plataforma - será abordada a metodologia utilizada na execução deste trabalho bem como serão apresentadas todas as etapas da concepção da Plataforma. Serão estipulados os requisitos que o sistema deverá cumprir, ou seja, os pontos aos quais a solução deverá responder recorrendo a casos de uso para os exemplificar. Será mostrada a forma como foi efectuada a exploração da ferramenta - XenServer, apresentada a arquitectura da rede que foi projectada, demonstrada a base de dados da plataforma e por fim expostos os problemas encontrados ao longo do desenvolvimento do trabalho.

4.1 - Metodologia utilizada

Em todos os trabalhos efectuados é necessário seguir uma metodologia na execução do mesmo. Seja na elaboração de trabalhos científicos, ou até em trabalhos mais simples do quotidiano, estabelecem-se linhas orientadoras para definir o plano de trabalho. Desta forma foi definida uma metodologia para o desenvolvimento desta dissertação, fundamentada na metodologia para a elaboração de trabalhos científicos. Esta metodologia encontra-se dividida em quatro fases: formulação da pergunta, realização da pesquisa, desenvolvimento do trabalho e interpretação dos resultados.

A fase inicial é sempre a formulação da pergunta. Antes de ser iniciada a investigação deve-se definir e estruturar muito bem qual o cerne da nossa questão. Os primeiros capítulos desta dissertação foram dedicados à formulação, descrição e definição de objectivos da pergunta que deu origem a este tema. Posteriormente é que se poderá dar início à fase de investigação. É pertinente que exista a exploração da temática e de trabalhos já realizados, para evitar cair no óbvio elaborando um trabalho que acrescenta muito pouco, sendo apenas mais uma fonte de informação igual a muitas outras.

A investigação é uma forma de actualização, servindo de base para a inovação na concepção de qualquer trabalho. Com efeito, esta dissertação reflecte, como se pode ver no Capítulo anterior - Revisão de Conceitos e Tecnologias - que a fase de arranque deste projecto foi a investigação, que serviu de apoio para a tomada de decisão no momento de escolha da ferramenta a utilizar - XenServer - para a concretização da plataforma de gestão.

Este Capítulo - Desenvolvimento da plataforma - remete-se para a fase do desenvolvimento do trabalho.

4.2 - Requisitos do sistema

Com o intuito de fazer com que o sistema fosse de encontro com os objectivos esperados, foram definidos os requisitos que o sistema deveria responder.

Os requisitos do sistema que foram considerados para o desenvolvimento da plataforma, foram a instalação automática do sistema, ou seja, a partir de um CD-ROM a ferramenta instala-se automaticamente, sem ser necessária qualquer tipo de intervenção ou qualquer tipo de conhecimento por parte do utilizador para efectuar a instalação. Para alcançar a simplicidade da instalação deverá ser alterado o conteúdo do CD-ROM de instalação.

Outro dos requisitos foi a possibilidade de a configuração e gestão do sistema ser feita de uma forma simples e intuitiva com uma interface *user-friendly*.

A plataforma deverá ser capaz de criar e gerir *Pools*, onde serão adicionados os servidores, nesses mesmos servidores deverá ser possível a criação e parametrização de máquinas virtuais. Também deverá ser possível adicionar repositórios às *Pools* que serão bastante importantes para o bom funcionamento do sistema, pois para além de servirem de repositório das máquinas virtuais, proporcionam também a migração das máquinas pois estas estão sempre acessíveis por todos os servidores na *Pool*, pelo facto de o repositório ser partilhado. Um dos requisitos mais pertinentes é a disponibilização da funcionalidade de alta disponibilidade, bem como a parametrização da mesma, especificando que máquinas deverão se encontrar sempre ligadas e as que deverão não ser reiniciadas caso o servidor físico, onde se encontram, falhe.

4.2.1 - Casos de Uso

Na concepção de um sistema são sempre definidos requisitos, de maneira a que o mesmo corresponda às necessidades do utilizador. Para a concepção desta plataforma, foram levantados os requisitos que deveriam estar incluídos nas funcionalidades do sistema, como foi evidenciado no ponto 4.2. Requisitos do Sistema. Uma das ferramentas utilizadas na Engenharia de Software como apoio para a definição dos requisitos são os Casos de Uso.

A definição de Casos de Uso é uma ferramenta muito útil no momento de estipular o comportamento do sistema. Os Casos de Uso representam a interacção entre o utilizador (humano ou máquina) e o sistema. Essa interacção é marcada por uma acção que é efectuada pelo utilizador e que provoca um certo comportamento do sistema, fazendo-o executar uma funcionalidade do mesmo. Ou seja, cada caso de uso descreve um cenário possível dessa interacção.

Para isso, foram identificados os Casos de Uso principais desta plataforma com base nos requisitos do sistema. Os Casos de Uso estão organizados pelo objecto ao qual poderá ser exercida a acção: Sessão, *Pool*, *Host*, Máquina Virtual, Interface de Rede Virtual, Disco Virtual, Repositório de Dados e Alta disponibilidade.

4.2.1.a - Identificação dos Casos de Uso

Neste Ponto, foram identificados os Casos de Uso Principais, para que no ponto seguinte se possa proceder à sua descrição. Os Casos de Uso identificados foram:

Sessão	Efectuar login, Editar utilizador, Seleccionar Idioma, Efectuar logout
Pool	Editar Pool, Eliminar Pool, Visualizar Pool
Host	Adicionar Host, Editar Host, Visualizar Host, Adicionar Host a uma Pool, Retirar Host de uma Pool
Máquina Virtual	Criar Máquina Virtual, Eliminar Máquina Virtual, Editar Máquina Virtual, Visualizar Máquina Virtual, Migrar Máquina Virtual ,Reiniciar Máquina Virtual (Normal), Reiniciar Máquina Virtual (Forçado), Desligar Máquina Virtual (Normal), Desligar Máquina Virtual (Forçado)
Interface de rede virtual	Adicionar Interface de rede virtual, Eliminar Interface de rede virtual
Disco virtual	Adicionar disco virtual, Eliminar disco virtual
Repositório de dados	Adicionar repositório de dados, Visualizar repositório de dados, Eliminar repositório de dados
Licenciamento	Aplicar licença
Alta disponibilidade	Activar Alta disponibilidade, Desactivar Alta disponibilidade, Configurar Alta disponibilidade

Tabela 4.1 - Casos de Uso

4.2.1.b - Descrição dos Casos de Uso

Após a identificação dos Casos de Uso, serão explicados em que cada um consiste, alguns recorrendo aos diagramas de Casos de Uso, e outros descrevendo apenas, visto que não será necessário utilizar os diagramas pelo facto de serem muito simples e comuns.

Sessão

Efectuar Login - O utilizador introduz o *username* e *password* para aceder à interface de gestão da plataforma.

Editar Utilizador - O utilizador define um novo *username* e uma nova *password*.

Seleccionar Idioma - O utilizador escolhe qual o idioma que pretende que o sistema seja apresentado.

Efectuar Logout - O utilizador efectua logout para encerrar a sua sessão no sistema.

Pool

Editar Pool - Através desta acção o utilizador consegue editar o nome, o *master* e o repositório de dados *default* da *Pool*.

Eliminar Pool - Com esta opção a *Pool* é eliminada, sendo igualmente eliminados os dados da base de dados, tanto da *Pool* como dos respectivos *Hosts* e repositórios de dados.

Visualizar Pool - Esta acção permite ao utilizador a visualização das informações da *Pool*.

Host

Adicionar Host

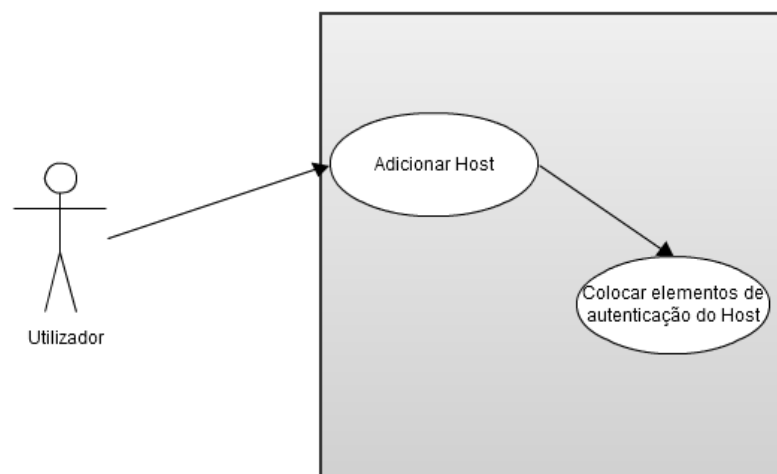


Figura 4.1 - Caso de Uso Adicionar Host

Editar Host - Através desta acção o utilizador consegue editar o nome, o IP, a *gateway*, o *Domain Name System* (DNS), a Máscara e o tipo de IP do *Host*.

Visualizar Host - Esta acção permite ao utilizador a visualização das informações do *Host*.

Adicionar Host a uma Pool - Acção que permite ao utilizador adicionar o Host a uma Pool já existente.

Retirar Host de uma Pool - A partir desta acção o utilizador pode retirar o Host de uma Pool.

Máquina Virtual

Criar Máquina Virtual

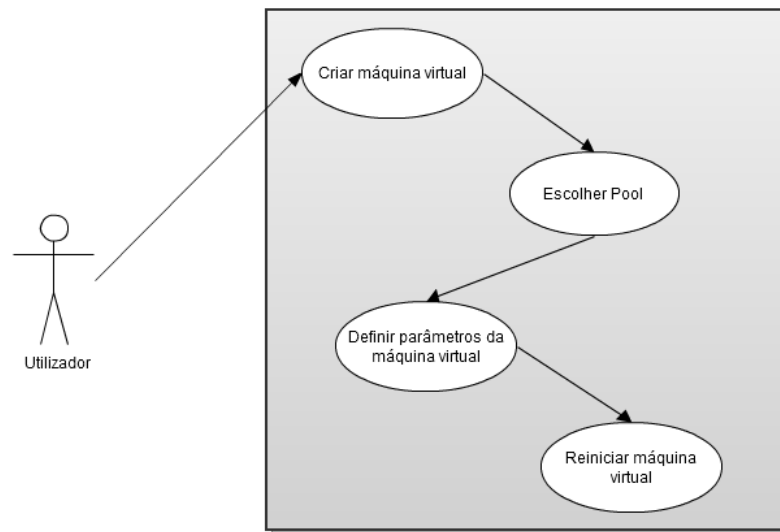


Figura 4.2 - Caso de Uso Criar Máquina Virtual

Eliminar Máquina Virtual - Ao seleccionar esta opção, o utilizador elimina a máquina virtual, escolhida, do servidor aonde ela se encontrava alojada.

Editar Máquina Virtual - Através desta acção o utilizador consegue editar o nome, a memória e o número de CPUs virtuais.

Visualizar Máquina Virtual - Esta acção permite ao utilizador a visualização das informações da Máquina Virtual.

Estes casos de uso tal como o nome indica permitem efectuar as respectivas acções:

Migrar Máquina Virtual, Reiniciar Máquina Virtual (Normal), Reiniciar Máquina Virtual (Forçado), Desligar Máquina Virtual (Normal) e Desligar Máquina Virtual (Forçado)

Interface de Rede Virtual

Adicionar Interface de rede virtual - Esta acção permite ao utilizador a criação de uma nova interface de rede virtual e consequente associação à máquina virtual.

Eliminar Interface de rede virtual - Ao seleccionar esta opção, o utilizador elimina a interface de rede virtual, escolhida, da máquina virtual à qual esta pertencia.

Disco Virtual

Adicionar disco virtual - Esta acção permite ao utilizador a criação de um novo disco virtual e consequente associação à máquina virtual.

Eliminar disco virtual - Ao seleccionar esta opção, o utilizador elimina o disco virtual, escolhido, da máquina virtual à qual este pertencia.

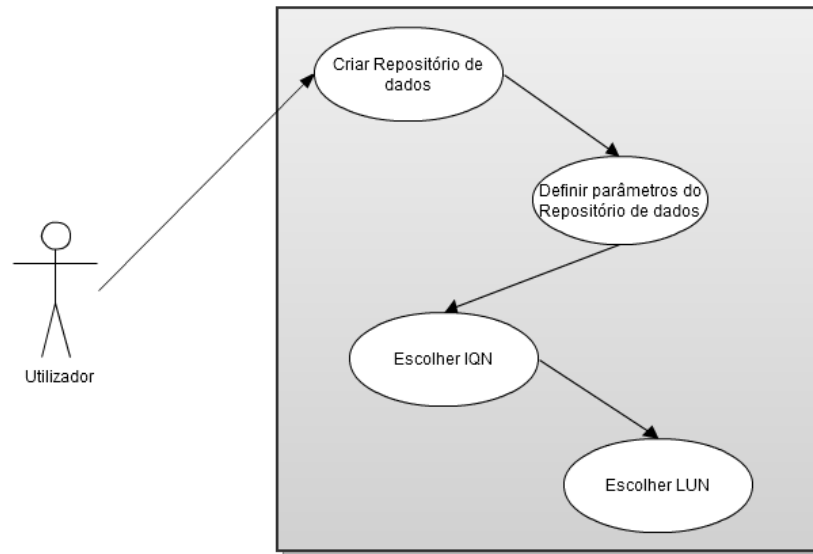
Repositório de dados**Criar repositório de dados**

Figura 4.3 - Caso de Uso Criar Repositório de dados

Visualizar repositório de dados - Esta acção permite ao utilizador a visualização das informações do Repositório de dados.

Eliminar repositório de dados - Com esta opção o Repositório de dados é eliminado, sendo igualmente eliminados os dados da base de dados.

Licenciamento

Aplicar Licença - A partir desta acção o utilizador faz upload de uma licença válida para os servidores XenServer.

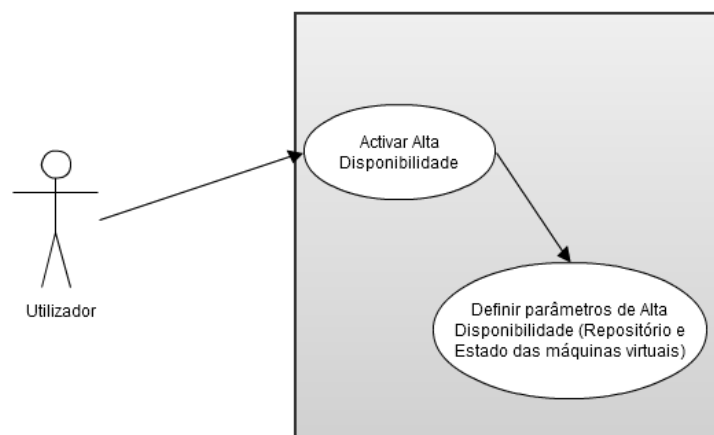
Alta disponibilidade**Activar Alta disponibilidade**

Figura 4.4 - Caso de Uso Activar Alta disponibilidade

Desactivar Alta disponibilidade - Esta acção permite ao utilizador desactivar a opção de alta disponibilidade na Pool pretendida

Configurar Alta disponibilidade - Acção que permite parametrizar os estados das máquinas virtuais para definir as prioridades em caso de falha.

4.3 - Exploração da ferramenta XenServer

Como foi referido na conclusão do capítulo anterior, a ferramenta seleccionada foi o XenServer. Após a escolha da ferramenta e de serem definidos os requisitos que a plataforma deveria cumprir, foi feito um levantamento exaustivo de informação para servir de apoio à compreensão do funcionamento da ferramenta.

Neste levantamento foi descoberto, na documentação do XenServer, um *Software Development Kit* (SDK) [39] sob a forma de máquina virtual, que depois de importada fornecia uma página *Web* com toda a informação sobre a *Application Programming Interface* (API) disponibilizada. Esta API fornece todo o tipo de informações para que, através de pedidos por XML-RPC, se pudesse comunicar com o servidor, adquirir todo o tipo de informação sobre o mesmo e efectuar operações como criar e gerir máquinas virtuais e todos os componentes associados e alterar dados dos servidores existentes.

Posteriormente à etapa de compreensão de como funcionaria a ferramenta e de quais as limitações da mesma, foi efectuado um estudo de como seria projectada a rede, que poderemos verificar no ponto - 4.4 Arquitectura de rede - deste capítulo. Com a rede projectada, iniciou-se a elaboração de alguns testes de comunicação com o servidor, de forma a verificar o que seria necessário enviar no pedido para obter a resposta pretendida.

O exemplo seguinte ilustra um desses testes, que neste caso é o estabelecimento de uma sessão no XenServer:

```
...
$action = $_REQUEST[fn];
$function = $action;
$params = array ();

switch ( $action ) {
    case 'session.login_with_password':
        //username
        $p1 = new XML_RPC_Value ( $_REQUEST[p1], "string" );
        // password
        $p2 = new XML_RPC_Value ( $_REQUEST[p2], "string" );
        $params = array ( $p1, $p2 );
        break;
}

...

//criação da mensagem

$message = new XML_RPC_Message ( $function, $params );

...

//envio da mensagem
```

```

$result = $client->send ( $message );
//selecção do campo value da resposta
$val = $result->value();
//passagem da resposta para array
$dataArray = XML_RPC_decode($val);
//impressão do resultado
echo "DATAARRAY<pre>"; print_r($dataArray); echo "</pre>";

```

O resultado deste teste é o seguinte, retornando se o pedido foi bem sucedido e a referencia da sessão estabelecida para usar noutras funções:

DATAARRAY

```

Array
(
    [Status] => Success
    [Value] => OpaqueRef:75019782-217b-fa73-3f68-824d76fa4829
)

```

Figura 4.5 - Resultado da função de teste

Depois de efectuados vários testes com várias funções, foram desenvolvidas um conjunto de funções que para além de fazerem os pedidos, tratavam as respostas e filtravam o que interessava. Segue um exemplo de uma dessas funções, onde é adquirido a referência da sessão estabelecida no XenServer:

```

//login no XenServer
function xapi_login ($ip, $user, $pass)
{
    global $client;
    $client = new XML_RPC_Client ( "/xmlrpc/server.php", $ip, 443 );
    $user = new XML_RPC_Value ( $user , "string");
    $pass = new XML_RPC_Value ( $pass , "string");
    // criar a mensagem
    $msg = new XML_RPC_Message ( "session.login_with_password" ,
array ($user,$pass));
    // envio da mensagem
    $result = $client->send ( $msg );
    $check = checkComm ($result);
    //se conseguir comunicar com o XenServer e não der erro
    if ($check == 'OK')
    {
        $val = $result->value();
        $status = $val->structmem("Status");
        //se o valor for diferente de "Success" imprime o erro
        if ($status->scalarval() != "Success")
        {
            $value = XML_RPC_decode($val);
            return $error = $value["ErrorDescription"]["0"] . ", "
. $value["ErrorDescription"]["1"];
        }
        $value = XML_RPC_decode($val);
    }
}

```

```

        //retorna o valor no campo "Value"
        return      $ssidval = $value["Value"];
    }
    else
        return false;
}

```

4.4 - Arquitectura do sistema

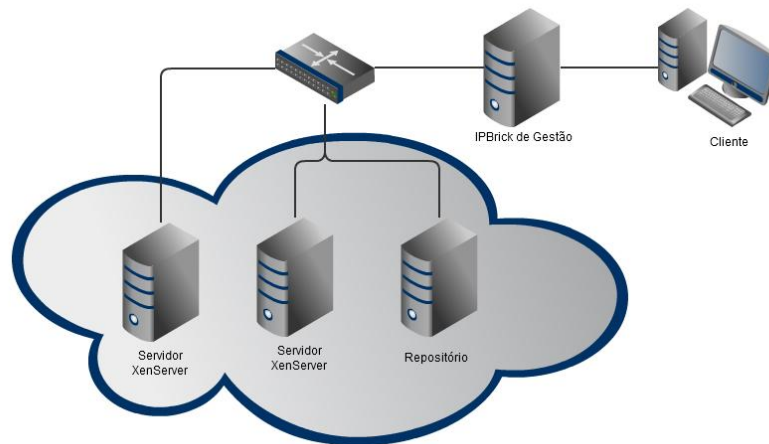


Figura 4.6-Arquitectura do sistema

A figura 4.6 representa a arquitectura de rede que foi projectada. Como se pode visualizar, a rede tem os servidores com o software XenServer instalado, que servem de alojamento das máquinas virtuais, o repositório da rede responsável por armazenar todo o conteúdo das máquinas virtuais criadas na rede, o servidor com o sistema operativo IPBrick instalado a servir de alojamento para a plataforma de gestão dos componentes da rede e contém também toda a informação, sobre a disposição da rede, guardada. A partir da estação cliente acede-se à plataforma alojada no servidor IPBrick para efectuar todo o tipo de gestão na rede disponível.

4.5 - Base de dados

Para facilitar a aquisição de dados e a recuperação em caso de falha foi criada uma base de dados onde são guardadas as informações da rede, à excepção das máquinas virtuais, pois são adquiridas em tempo real juntamente com todos os seus dados. A base de dados é constituída da seguinte forma:

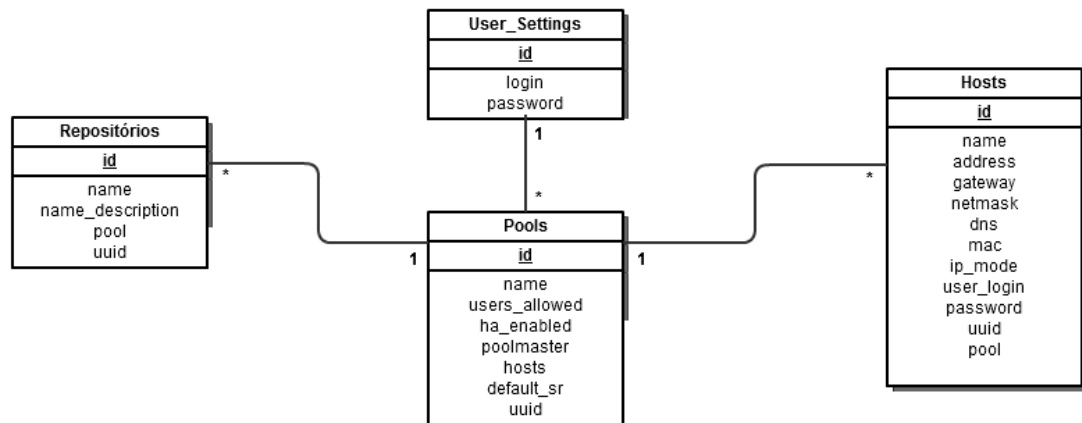


Figura 4.7 - Esquema da base de dados

4.6 - Problemas encontrados

É comum a todos os trabalhos elaborados, surgirem problemas. Ao longo do desenvolvimento da plataforma foram emergindo algumas barreiras que foram sendo contornadas e ultrapassadas de forma a poder atingir os objectivos definidos.

O XenServer suporta *guests* Linux e Windows, disponibilizando templates para ajudar na criação e instalação das máquinas virtuais. Existem *full-templates* que são aqueles que criam uma máquina virtual e instalam a distribuição pretendida. Os Físico para Virtual disponibilizam a passagem de uma máquina física para máquina virtual com a distribuição pretendida. Existem ainda os templates que permitem a instalação a partir do CD-ROM da distribuição e os templates que possibilitam a instalação a partir do repositório da distribuição disponibilizado pela rede (internet). Na tabela 4.2 são mostrados os templates disponibilizados para as várias distribuições linux[40].

Distribuição	Full-Template	Físico para Virtual	CD-ROM	Rede
Debian Etch 4.0	X			
Debian Lenny 5.0				X
Red Hat Enterprise Linux 3.6-3.8		X		
Red Hat Enterprise Linux 4.5-4.7			X	X
Red Hat Enterprise Linux 5.0-5.3 32-bit			X	X
Red Hat Enterprise Linux 5.0-5.3 64-bit			X	X
SUSE Linux Enterprise Server 9 SP1/2/3		X		
SUSE Linux Enterprise Server 9 SP4		X	X	X

SUSE Linux Enterprise Server 10 SP1/2 32-bit/64-bit			X	X
SUSE Linux Enterprise Server 11 32-bit/64-bit			X	X
CentOS 4.5, 4.6			X	X
CentOS 4.7				X
CentOS 5.0-5.3 32-bit			X	X
CentOS 5.0-5.3 64-bit			X	X
Oracle Enterprise Linux 5.0-5.2 32-bit			X	X
Oracle Enterprise Linux 5.0-5.2 64-bit			X	X

Tabela 4.2 - Templates disponibilizados pelo XenServer

Um dos problemas surgiu quando a adição da primeira máquina virtual IPBrick. A IPBrick tem algumas distribuições e a utilizada para este projecto assenta na distribuição Debian.

Para ser instalada a IPBrick no XenServer foi utilizado o template '*Other Install Media*'. Isto deve-se ao facto de o único template que o XenServer disponibiliza para o sistema operativo Debian tratar-se de um *full-template*, ou seja, cria automaticamente uma máquina virtual com o sistema operativo Debian instalado, o que não acontece em outros templates disponibilizados, pois permitem outras formas de instalação como se confirma na figura. Desta forma, não foi possível, através do template para o sistema operativo Debian, instalar o sistema operativo IPBrick, mas sim através do template '*Other install Media*' que é utilizado quando o sistema a instalar não se encaixa em nenhum dos templates disponibilizados.

Ao usar o template '*Other Install Media*' a máquina é instalada em modo *Hardware Virtualization Mode* (HVM), virtualização total, logo como o XenServer assenta num tipo de virtualização chamado Paravirtualização, não se consegue tirar total partido das funcionalidades das máquinas. Com a máquina a funcionar em modo HVM podia-se apenas efectuar operações como Reiniciar/Desligar e ambas forçadas, não sendo possível Reiniciar, Desligar, Suspendar nem migrar uma máquina virtual. Para além disso não se tinha acesso a informações como o IP das interfaces de rede virtuais que foram criadas para cada máquina virtual, não se conseguia criar interfaces de rede virtuais nem discos virtuais em tempo real, ou seja, se houvesse intenção de adicionar alguma interface de rede ou algum disco teria de se Reiniciar a máquina, de modo forçado, para estas alterações entrarem em vigor. Após muita pesquisa descobriu-se que mudando alguns parâmetros nessa máquina virtual podia-se ter a mesma a trabalhar em modo de Paravirtualização. Era necessário aplicar os seguintes comandos no XenServer:

xe vm-param-set uuid=<vm uuid> HVM-boot-policy="" (apaga o modo de inicialização HVM)

xe vm-param-set uuid=<vm uuid> PV-bootloader=pygrub (define pygrub como modo de inicialização)

xe vbd-param-set uuid=<vbd uuid> bootable=true (define o dispositivo de arranque)

Contudo, mesmo depois de colocados os comandos a máquina não iniciava, como era pretendido. Após mais alguma pesquisa foi descoberto o problema. O kernel da IPBrick, aquando a compilação, teria de ter a *flag* do Xen activada. Para isso foram efectuadas alterações na compilação do kernel a utilizar pela IPBrick, como mudar a família do processador, como mostra a Figura 4.8.

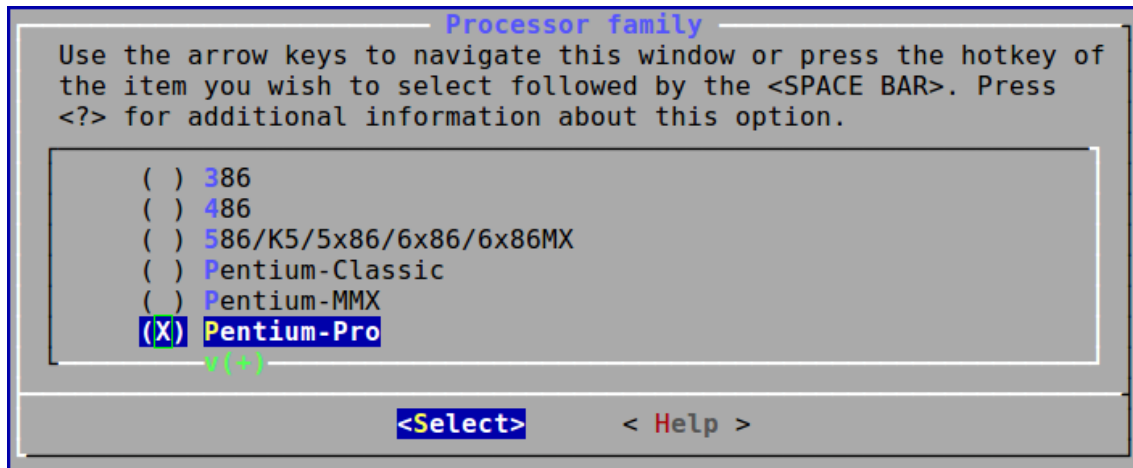


Figura 4.8 - Família do Processador

Como se pode verificar na Figura 4.8, foi alterada a família do processador para a “Pentium-Pro”, pois para o XenServer correr a máquina em modo Paravirtualizado (PV), é necessário o kernel ter uma *flag* de paravirtualização activa, e para a *flag* aparecer na compilação do kernel é necessário escolher um processador com suporte para a virtualização, como podemos visualizar na Figura 4.9.

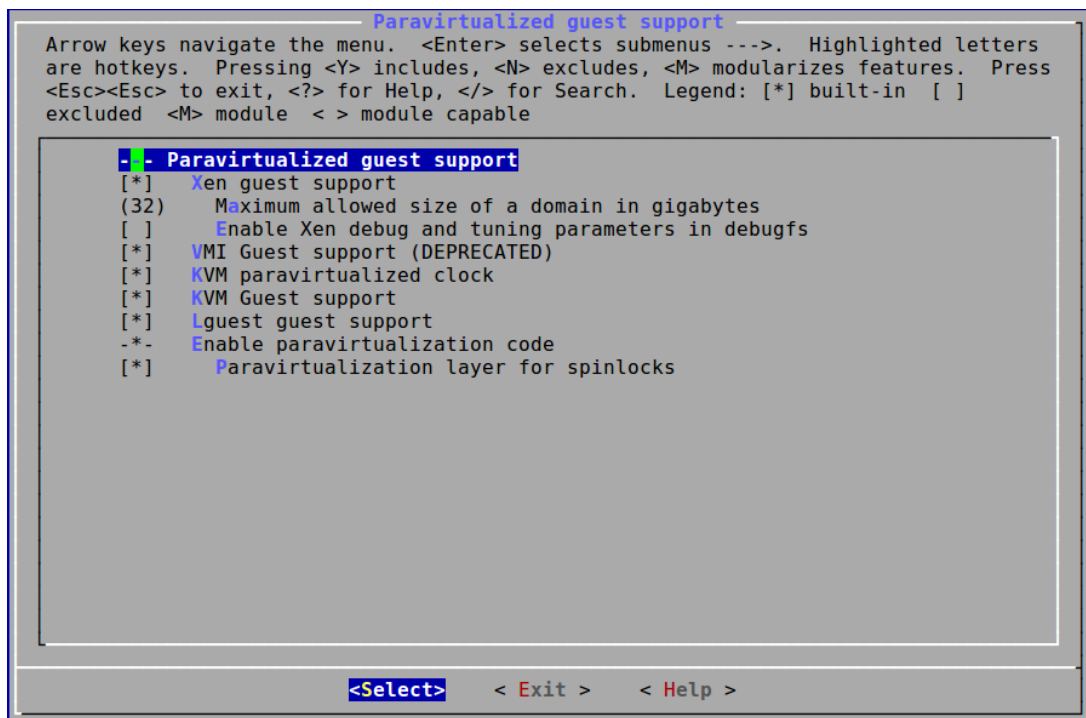


Figura 4.9 - Activação da flag de Paravirtualização

Outro problema que surgiu, foi quando se adicionou uma máquina virtual IPBrick, pois era necessário saber quando a instalação do sistema operativo se encontrava no fim. O sistema operativo IPBrick após instalação faz o pedido para se reiniciar a máquina, logo pretendia-se determinar qual o momento certo para efectuar essa operação e para proceder à alteração do modo da mesma, ou seja, do modo HVM para PV. Para o efeito, após ter sido descoberto que o XenServer dispunha de um servidor VNC, procedeu-se à criação de alguns scripts cuja função era a criação de um túnel entre a máquina, onde se encontrava a plataforma alojada, e o servidor que continha a máquina virtual, pois era a única forma de ter acesso às consolas. Devido a este problema foi desenvolvida uma solução. Após a adição de uma máquina virtual com o sistema IPBrick o utilizador é encaminhado para uma página *Web* onde é mostrada a consola da máquina virtual que permite a visualização da instalação através da utilização de uma applet java, que suporta o protocolo VNC, como cliente, aplicação disponibilizada pela empresa. Para além disso disponibiliza também um botão que serve para quando o sistema operativo finalizar a instalação, reiniciar a máquina e alterá-la com a finalidade de a mesma iniciar trabalhando em modo PV.

Outras das dificuldades sentidas, foi a complexidade da compreensão da API disponibilizada, pois a falta de exemplos de como utilizar a API fez com que o trabalho para a exploração da ferramenta XenServer fosse redobrado.

4.7 - Conclusão

Dentro da metodologia estabelecida, o desenvolvimento da plataforma, insere-se na fase em que são reunidas as ferramentas e as informações, recolhidas na fase de pesquisa - Revisão de Conceitos e Tecnologias - e começa-se a desmontar uma problemática, resultando na solução.

Ao longo do desenvolvimento da plataforma, como se pôde verificar neste capítulo, foram definidos requisitos e casos de uso para o sistema, montada a arquitectura da rede e a base de dados da plataforma e durante o desenvolvimento da mesma foram ultrapassadas barreiras que surgiram ao longo do desenvolvimento do projecto. As dificuldades não se prenderam com a impossibilidade de cumprir os requisitos nem com os problemas que foram surgindo, pois foram sempre encontradas formas de contornar a situação, a principal dificuldade agregou-se à escassez de informação sobre a utilização da API disponibilizada pela Xen, contudo, a informação foi sendo encontrada em fóruns onde continha testemunhos que ajudaram a desvendar algumas questões.

Capítulo 5

Resultados

Este capítulo- Resultados, remete-se para a solução obtida do trabalho desenvolvido. Com o intuito de avaliar os resultados e verificar se os objectivos definidos foram cumpridos, serão interpretados os resultados e apresentada a solução final - a plataforma.

5.1 - Cenários de teste

De modo a garantir as funcionalidades da plataforma, foram criados alguns cenários, de maneira a testar o comportamento e limitações do sistema. Isto é, variando o cenário, poderemos ter variações no comportamento do sistema e será sempre necessário assegurar os serviços disponibilizados independentemente do cenário. Com o intuito de testar as funcionalidades da plataforma, foram criados dois cenários que partiram da arquitectura do sistema que está representada na figura 5.1.

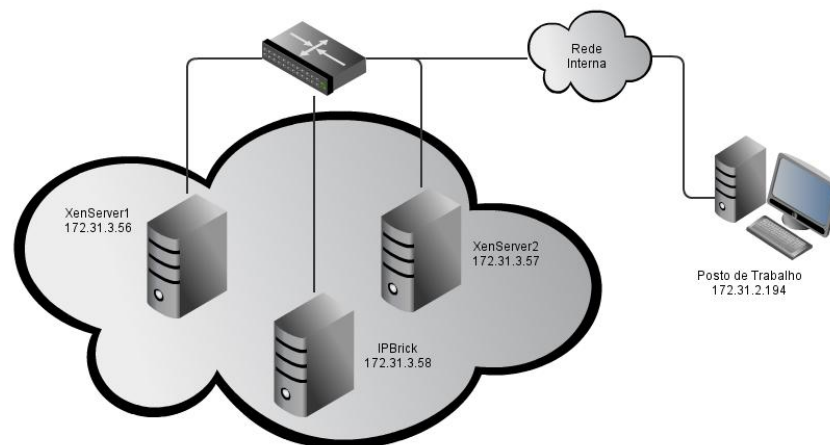


Figura 5.1 - Cenário Inicial

Com base na figura 5.1, poder-se-á verificar a existência de dois servidores com o software XenServer instalado, um servidor com o software IPBrick que para além de servir de alojamento para a plataforma desenvolvida, serve como repositório e uma estação cliente que servirá para aceder a plataforma de gestão.

Para validar todo o tipo de acções foram analisados os dois cenários. No primeiro cada servidor XenServer encontrava-se numa *Pool* como é mostrado na figura 5.2, para validar o sistema quando existe apenas um servidor em cada *Pool*.

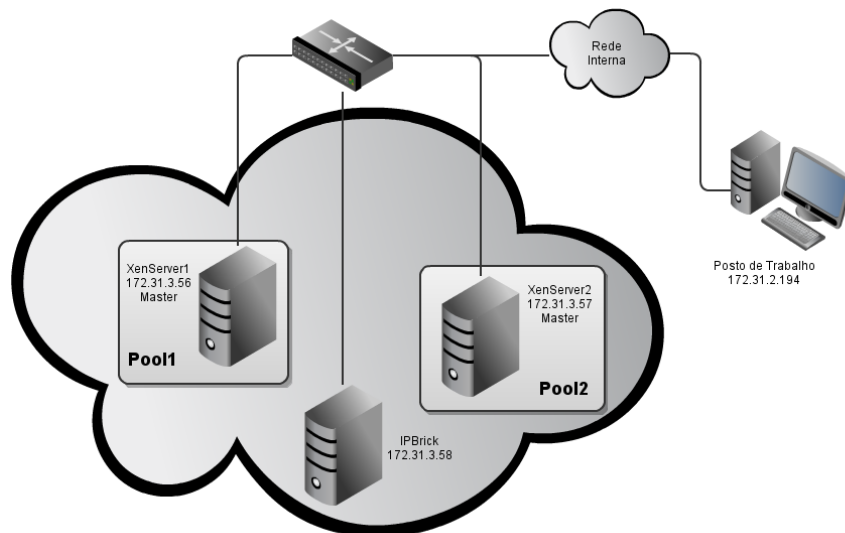


Figura 5.2 - Primeiro Cenário

No segundo os dois servidores XenServer encontravam-se inseridos na mesma *Pool*, como podemos confirmar na figura 5.3, para validar a plataforma quando cada *Pool* é composta por dois ou mais servidores.

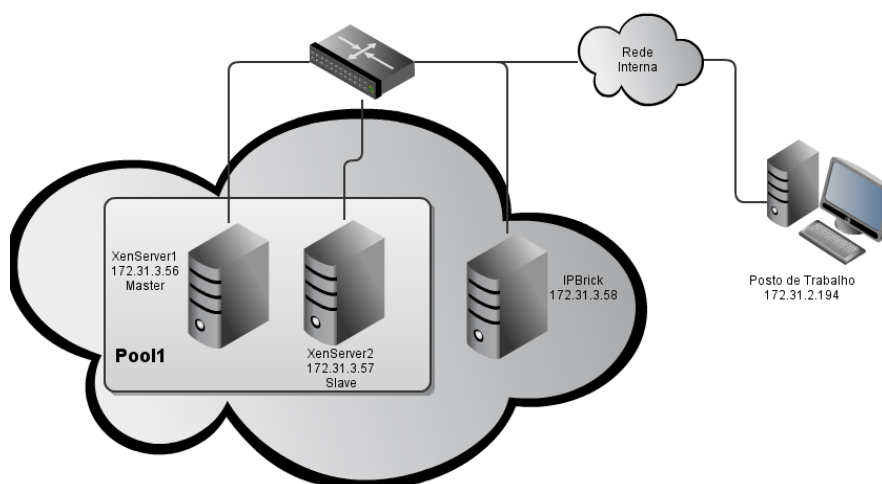


Figura 5.3 - Segundo Cenário

5.1.1 - Configurações

5.1.1.a - Configuração dos Servidores XenServer

Teve-se em atenção na escolha do hardware destes servidores, pois deveriam corresponder a certos requisitos[41], como é mostrado na tabela 5.1.

CPU	Um ou mais CPUs de 64-bit x86, mínimo 1,5 GHz. É recomendado um de 2 GHz ou um CPU multicore mais rápido
RAM	Mínimo 1 GB. Recomendado 2 GB ou mais.
DISCO	Mínimo 16 GB. Recomendado 60 GB.
REDE	No mínimo uma <i>Network Interface Card</i> (NIC) com 100 Mbits/s. É recomendado uma NIC gigabit.

Tabela 5.1 - Requisitos XenServer

A configuração destes servidores é efectuada durante a instalação que se encontra especificada em[42].

Para ser criado o repositório local de ISOs, é necessário já se encontrar criada a pasta onde estarão localizadas as imagens com extensão iso e essa pasta terá de se encontrar exactamente em /var/opt/xen/iso_import/ISOs como é dito em [40].

5.1.1.b - Configuração do Servidor IPBrick

Para o servidor IPBrick trabalhar como repositório foi necessário instalar o open-iscsi e o iscsitarget. Depois de instalados foi necessário alterar o ficheiro ietd.conf de maneira a criar as *targets* e as LUNs pretendidas. Deve-se ter em atenção que as partições que são associadas às LUNs não se devem encontrar montadas. Neste caso foi adicionado o seguinte código ao ficheiro ietd.conf:

```
Target iqn.2010-04.com.cloud:storage
    Lun 0 Path=/dev/sdb1,Type=fileio

Target iqn.2010-04.com.cloud:teste
    Lun 0 Path=/dev/sda8,Type=fileio
```

Para a IPBrick servir como alojamento da plataforma foi necessária a criação da interface *Web*, a partir da plataforma de gestão do servidor IPBrick.

5.2 - Análise dos Resultados

5.2.1 - Adição do primeiro servidor

Ao aceder à plataforma efectua-se o login (Anexo A 1) que permite um acesso seleccionado às *Pools*, pois só o administrador do sistema é que terá acesso a todas as *Pools*.

No diagrama seguinte (Figura 5.4), poderão ser verificadas as acções que são efectuadas quando o utilizador acede pela primeira vez à plataforma e adiciona o primeiro servidor.

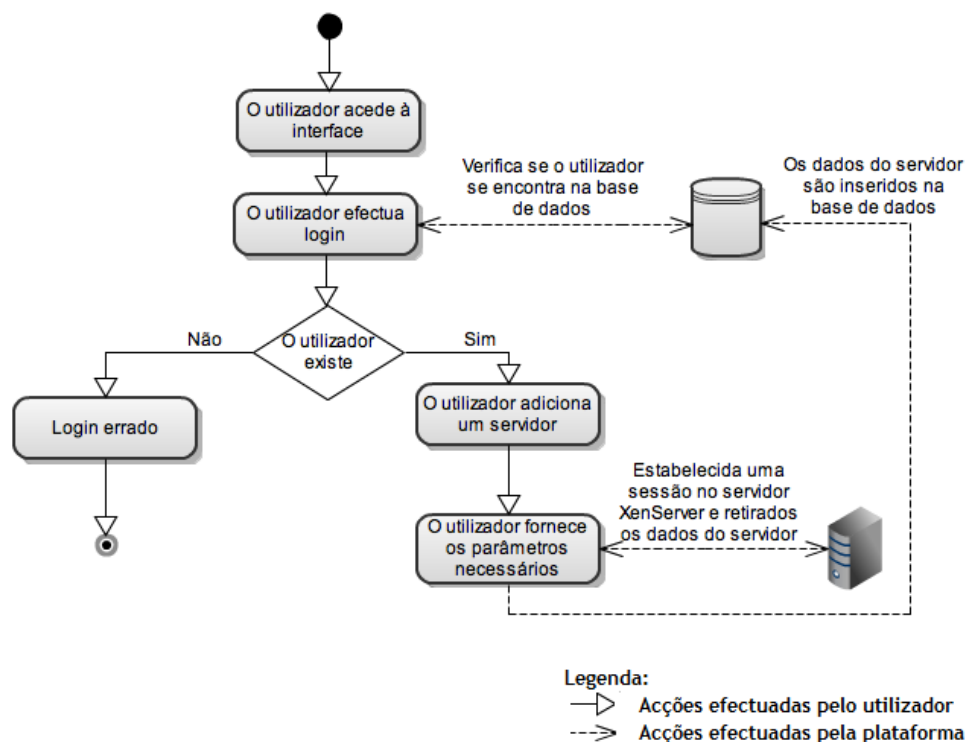


Figura 5.4 - Adição do primeiro servidor

Como se trata da primeira vez que o utilizador acede a esta plataforma, este não tem acesso a nenhum servidor, pois ainda não foi adicionado nenhum, com efeito são disponibilizadas apenas algumas opções (Anexo A 2). Para adicionar o servidor basta fornecer o *username*, a *password* e o IP do mesmo como é possível visualizar no diagrama na acção que remete ao fornecimento dos parâmetros necessários. (Anexo A 3). Só deverão ser adicionados os *masters* das *Pools*, pois a partir destes serão adicionados também os *slaves*. Quando é instalado o XenServer num servidor, este é *master* de si próprio, logo pode ser adicionado. No caso do servidor não estar associado a nenhuma *Pool*, é criada uma *Pool* com o nome “Pool”, automaticamente. Se por outro lado, o servidor adicionado já se encontrar associado a uma *Pool*, esta é adicionada com todas as suas definições, tal como o repositório de dados usado, isto no caso de estar definido, ou seja, herda todas as configurações já existentes, não tendo que ser novamente configurada. Durante o processo de adicionar o servidor, é criado, no mesmo, um repositório de ISOs que possibilitará mais tarde a criação de máquinas virtuais a partir das ISOs presentes nesse repositório. Aquando a inserção do servidor, são inseridos nas bases de dados todos os dados relevantes das *Pools*, dos *Hosts* e dos repositórios de dados, que se encontram nas mesmas, como se pode verificar na Figura 5.4.

A outra opção apresentada, no menu inicial da plataforma, é a opção de 'Configurações' (Anexo A 4), onde é possível alterar o *username* e a *password* de acesso à plataforma bem como o idioma pretendido.

Depois de adicionados um ou mais servidores (*masters*) são apresentadas mais opções como 'Hosts', 'Máquinas Virtuais', 'Repositórios de dados' e 'Alta disponibilidade' no menu principal.

5.2.2 - Pools

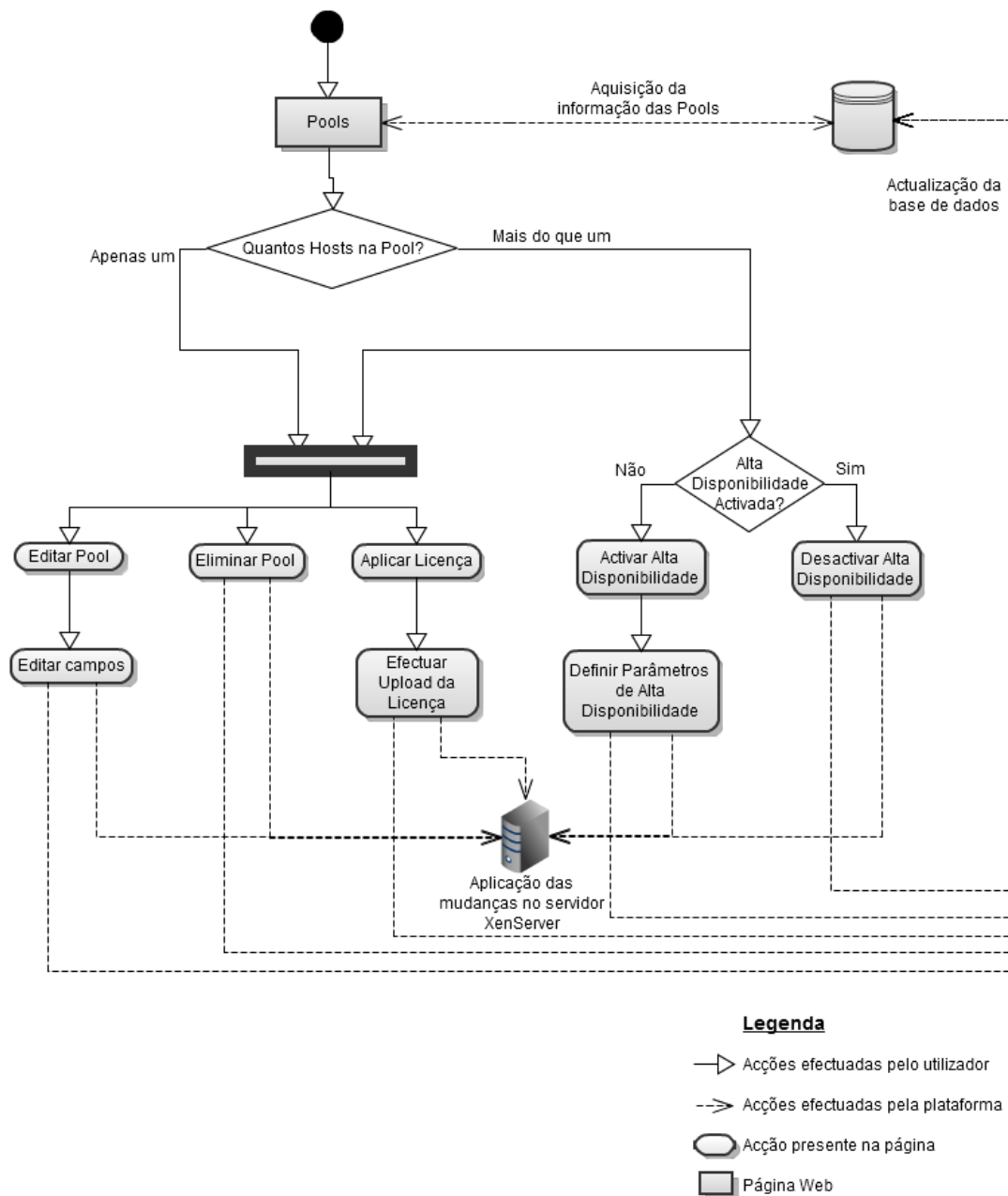


Figura 5.5 - Pools

Tal como está demonstrado na Figura 5.5, na página 'Pools' são mostradas as *Pools* que foram adicionadas, juntamente com alguns dados da mesma como o nome, o nome do servidor *master*, o nome dos servidores que se encontram na mesma e o nome do repositório de dados, por defeito, caso este já se encontre definido. Nas *Pools* adicionadas, é disponibilizado um submenu onde o utilizador tem a oportunidade de efectuar algumas acções sobre as mesmas (Anexo A 5). O Utilizador pode editar os campos da *Pool*, definindo o nome, escolhendo o servidor *master* e escolhendo o repositório de dados que deverá ser usado por defeito (Anexo A 6). Como se pode verificar na Figura 5.5, é disponibilizada uma opção de 'Aplicar Licença', pois a versão XenServer inicial é válida apenas para trinta dias e,

a partir desta opção, basta fazer o *upload* da licença adquirida no site da plataforma para poder usufruir da mesma durante um ano. Pode-se eliminar a *Pool*, eliminando esta da base de dados. Durante o processo de eliminação é eliminado também o repositório de ISOs adicionado durante a inserção da mesma, para evitar que, aquando uma nova inserção, não seja criado outro repositório de ISOs igual ao que já havia sido criado. A opção de activar alta disponibilidade só aparece como uma funcionalidade da *Pool*, se a mesma for constituída por dois ou mais servidores, ou seja, no primeiro cenário, pois só com um servidor na *Pool* não tinha sentido esta conter esta possibilidade de activar alta disponibilidade, pois esta opção tem o objectivo de se um servidor falhar as máquinas que se encontram nele serem migradas para outro servidor da *Pool*, e visto que a *Pool* só contém um servidor, a alta disponibilidade não se encaixa neste cenário.

Quando se trata do segundo cenário, com dois servidores na mesma *Pool*, é disponibilizada uma opção nova que permite activar a Alta disponibilidade na *Pool* (Anexo A 7). Ao seleccionar a opção 'Activar a Alta disponibilidade' tem de se especificar quais as máquinas virtuais da *Pool* que deverão reiniciar obrigatoriamente noutro servidor caso o servidor onde elas residem falhe, as que deverão reiniciar noutro servidor se este apresentar recursos suficientes, tais como a memória para as suportar e aquelas que não serão reiniciadas caso o servidor falhe. Quando é activada a opção de alta disponibilidade é alterada a opção de 'Activar a Alta disponibilidade' para 'Desactivar a Alta disponibilidade' (Anexo A 8).

5.2.3 - Hosts

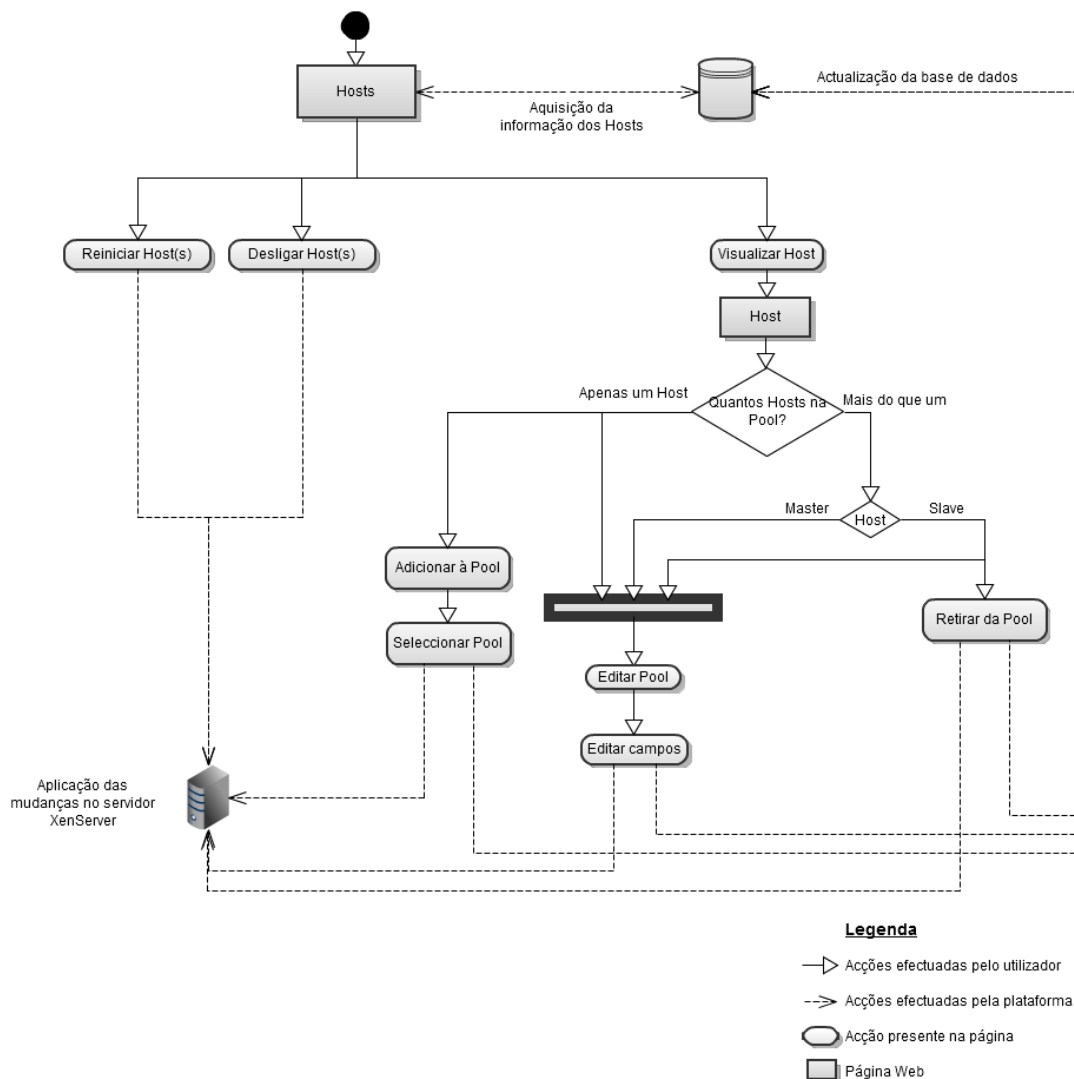


Figura 5.6 - Hosts

Como se pode verificar na Figura 5.6, na página 'Hosts' são mostrados os vários servidores físicos presentes em todas as *Pools*, sejam eles *masters* ou *slaves*. É apresentada uma lista com os nomes dos mesmos e com o nome da *Pool* a que pertencem. São disponibilizadas as acções como Reiniciar ou Desligar o(s) servidor(s), ou seja, basta seleccionar o servidor ou servidores, escolher a acção e submeter (Anexo A 9). Ao seleccionar o nome do servidor, o utilizador é reencaminhado para uma página que contém informações sobre o servidor escolhido (Anexo A 10). Como se trata de um único servidor na *Pool* (primeiro cenário), encontra-se disponível a opção de 'Adicionar à Pool', onde o utilizador pode adicionar este servidor a uma *Pool* existente. Ao seleccionar esta opção tem de se especificar a que *Pool* é que este servidor deve ser adicionado. Logo depois de ser adicionado, todos os dados referentes à *Pool* a que este pertencia são eliminados da base de dados bem como todos os recursos associados à *Pool*, por exemplo os repositórios de dados.

Existem várias diferenças quando uma *Pool* apresenta mais do que um servidor (segundo cenário). Uma delas é quando é visualizado um *Host* que seja *slave* na *Pool* onde se encontra,

em vez da opção ‘Adicionar à Pool’ é disponibilizada a opção ‘Retirar da Pool’ (Anexo A 11). Ao retirar um servidor, a base de dados actualiza-se, retirando o servidor da *Pool*, logo, se for necessário efectuar operações nesse servidor tem de se o adicionar novamente, porque depois de retirado o servidor encontra-se numa *Pool* na qual é *master*.

Esta página contém também a opção de editar o servidor, onde é possível mudar os campos nome, IP, gateway, máscara, DNS e a maneira como este obtém o seu IP, ou seja, se se trata de um IP estático ou de um IP adquirido por *Dynamic Host Configuration Protocol* (DHCP) (Anexo A 12). Ao submeter estas mudanças no servidor, estas são guardadas também na base de dados do servidor IPBrick, onde se encontra alojada a interface de gestão, como se pode comprovar na Figura 5.6. Esta operação é efectuada para que em caso de falhas dos servidores da rede, não se perca a configuração da rede, tal como, quem era o servidor *master* e os servidores *slaves*, a que *Pool* pertenciam e que repositórios de dados haviam.

5.2.4 - Máquinas Virtuais

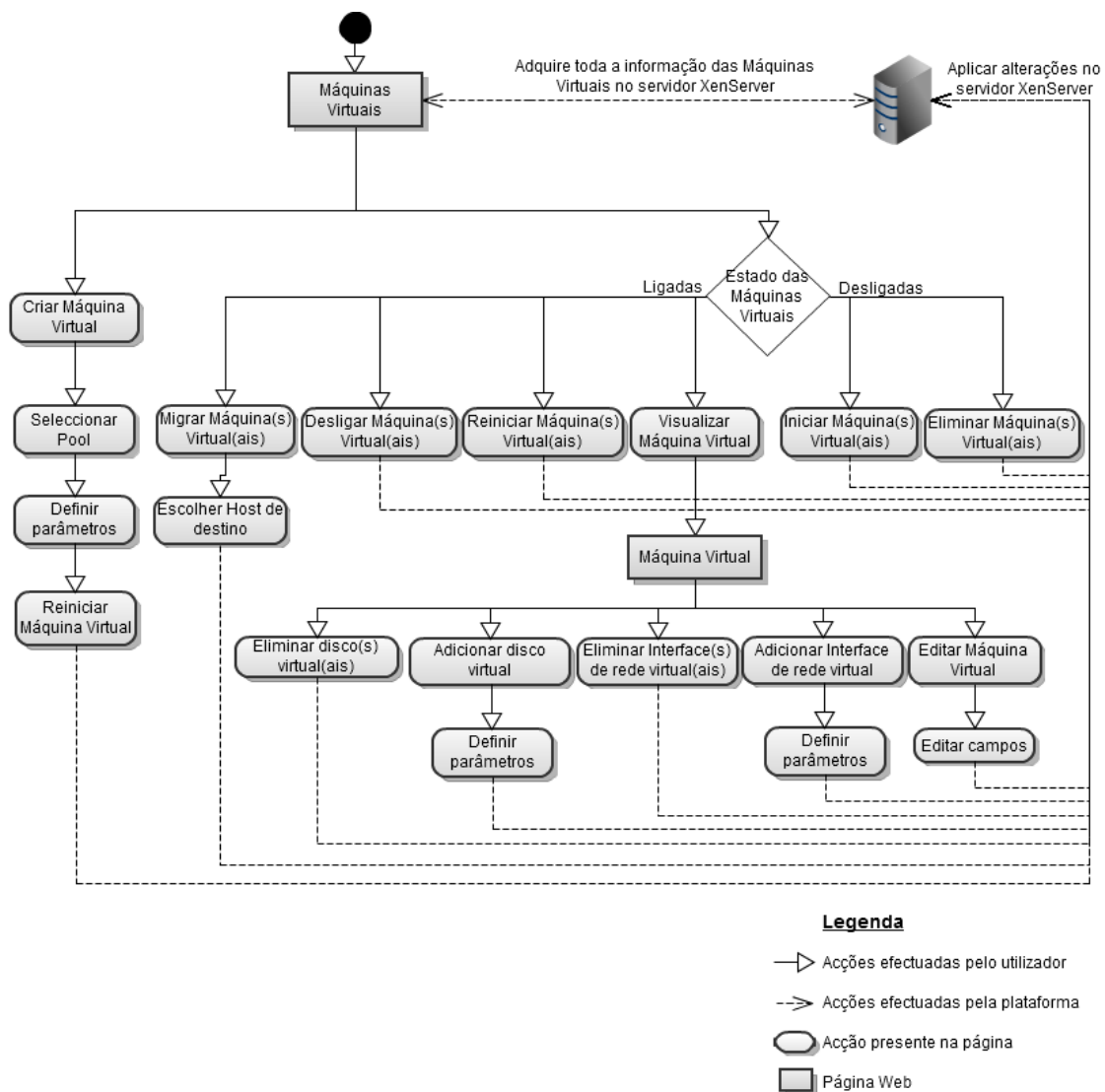


Figura 5.7 - Máquinas Virtuais

Com base na Figura 5.7, na página 'Máquinas Virtuais' são apresentadas duas listas: a primeira com as máquinas que se encontram ligadas e a segunda com as máquinas suspensas e desligadas. Esta lista contém o nome da máquina virtual e o nome do servidor onde esta se encontra. São disponibilizadas várias acções, tais como, Reiniciar, Desligar, Migrar a máquina para outro servidor da *Pool*, Reiniciar (Forçado) e o Desligar (Forçado) que podem ser aplicados tanto a uma como a várias máquinas virtuais (Anexo A 13). Tal como é possível visualizar na Figura 5.7, existe a opção de criar máquina virtual. Ao seleccionar esta opção é necessário seguir alguns passos que levam à criação da máquina virtual pretendida. No primeiro passo tem de se especificar em que *Pool* se pretende adicionar esta máquina (Anexo A 14). Depois de submetida a *Pool*, na qual se pretende criar a máquina virtual, é necessário especificar os parâmetros da máquina virtual: nome, descrição, memória, número de CPUs virtuais, o *Host* da *Pool* onde iniciar a máquina virtual, o repositório de dados onde serão guardados todos os dados das máquinas virtuais e por fim a imagem pretendida que se encontra guardada no repositório de ISOs (Anexo A 15). Depois de especificados os parâmetros, é disponibilizada a consola, alcançada através de uma applet java que suporta o protocolo VNC, com a instalação a decorrer para ser possível visualizar o final da instalação e aí reiniciar a máquina virtual. Para obter a consola é necessário criar um túnel *Secure Shell* (SSH) entre o servidor IPBrick, que tem a plataforma alojada, e o servidor XenServer, que dispõe de um servidor de VNC que possibilita a visualização não só da consola do servidor como também das máquinas virtuais, que se encontram nele instaladas. Esta máquina virtual é instalada em modo HVM, mas para obter total controlo sobre a máquina virtual é necessário passar para modo PV, logo, enquanto está a decorrer a instalação são mudados certos parâmetros para ocorrer esta alteração. Quando a instalação finaliza e a máquina é reiniciada, esta já se encontra a funcionar em modo PV (Anexo A 16).

Tal como nos 'Hosts', ao seleccionar o nome da máquina virtual obtém-se informações mais detalhadas sobre a mesma (Anexo A 17), onde se consegue visualizar também as interfaces de rede virtuais associadas à máquina virtual, podendo adicionar ou eliminar as existentes. Na opção 'Adicionar uma Interface de Rede Virtual' é onde se fornece também alguns parâmetros como a interface de rede física a que a virtual deverá se encontrar associada e especificar o MAC pretendido. Caso não seja pretendido especificar o MAC podemos seleccionar a opção 'Auto' e a este será atribuído um valor aleatório automaticamente (Anexo A 18). É apresentada uma lista de discos virtuais pertencentes à máquina, bem como as opções de adicionar novo disco e eliminar os que já estão criados. Na opção 'Adicionar um Disco Virtual' é necessário fornecer alguns parâmetros para a criação de um novo disco como nome, descrição, tamanho e o repositório de dados a usar para guardar as informações guardadas neste disco (Anexo A 19). É possível também editar os campos da máquina virtual como a memória que o servidor disponibiliza para a máquina virtual, o número de CPUs virtuais e o nome da máquina virtual (Anexo A 20).

5.2.5 - Repositório de dados

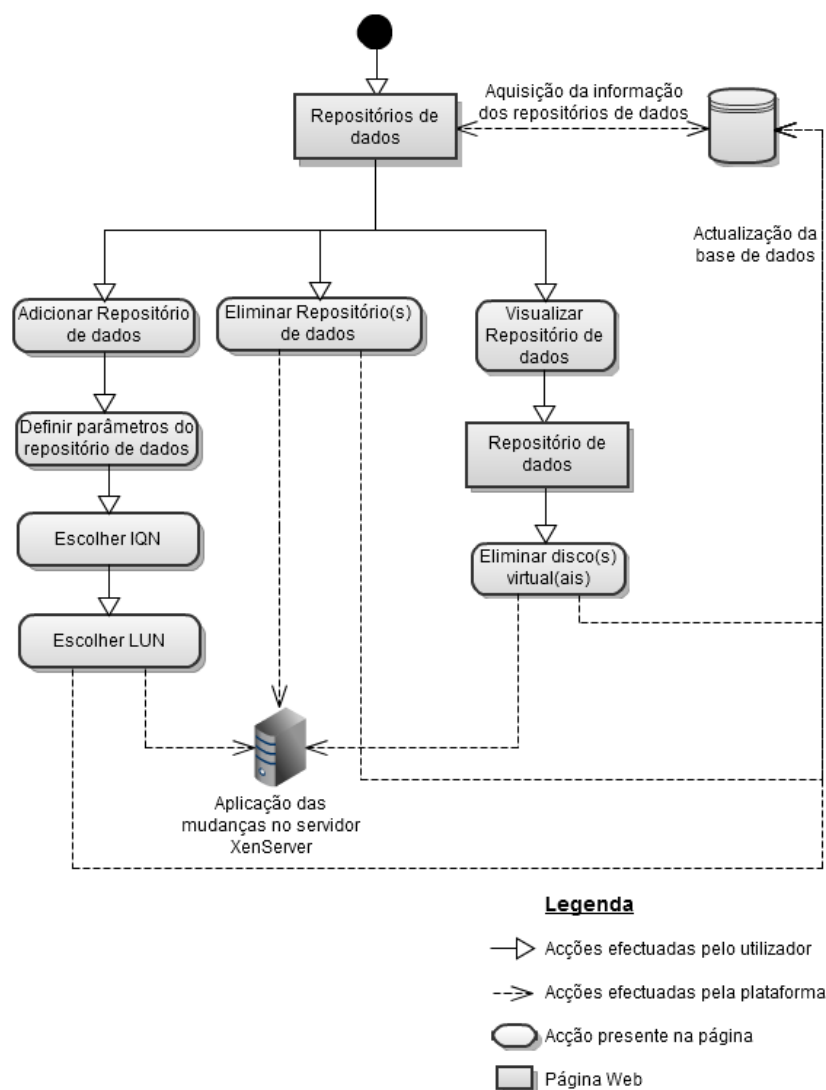


Figura 5.8 - Repositório de dados

Visualizando a Figura 5.8, na página ‘Repositório de dados’ pode-se visualizar os repositórios adicionados às *Pools*. É apresentada uma lista com o nome de vários repositórios e com o nome da *Pool* a que se encontram associados. São disponibilizadas as acções de Eliminar um ou mais repositórios e a de Adicionar um repositório (Anexo A 21).

Na opção ‘Adicionar Repositório de dados’, disponibiliza-se o campo para o fornecimento do nome que queremos que o repositório tenha, a descrição do repositório e o IP da máquina (servidor) que vai servir de repositório iSCSI (Anexo A 22). A partir destes parâmetros são apresentados os vários IQNs presentes no servidor a utilizar como repositório (Anexo A 23). Depois de escolhido o IQN pretendido são apresentadas as várias LUNs, que estão presentes na *target* seleccionada através da sua identificação IQN, que representam partições do disco do repositório (Anexo A 24).

Ao seleccionar o repositório de dados, visualiza-se o nome, a descrição, o espaço total, o espaço livre e uma lista de discos virtuais criados nesse repositório. Permite também a

possibilidade de eliminar um ou mais discos virtuais que se encontrem no repositório (Anexo A 25).

5.2.6 - Alta disponibilidade

Na opção ‘Alta disponibilidade’ pode-se verificar se alguma *Pool* da rede se encontra com esta opção activada. Caso esteja desactivada, é só fornecida a informação de que na *Pool* seleccionada não existe nada que garanta a alta disponibilidade das máquinas virtuais, logo se o servidor falhar as máquinas virtuais também falham (Anexo A 26).

Quando esta se encontra activa, é apresentada a situação actual do sistema, ou seja, com que definições de alta disponibilidade se encontram as máquinas virtuais pertencentes à *Pool* em questão. Neste menu pode-se também alterar estas definições, definindo o estado pretendido para as máquinas virtuais entre os disponíveis que são reiniciar, reiniciar se possível e não reiniciar (Anexo A 27).

5.3 - Compatibilidade

Neste tipo de sistema existe uma característica que é crucial para a sua aplicação à realidade, que é a compatibilidade do mesmo. Ou seja, o intuito deste sistema é vir a ser utilizado, logo, é pertinente que se insira com facilidade em diferentes ambientes. Visto que esta plataforma consiste num sistema que disponibiliza uma interface *Web* para a sua gestão e configuração, é necessário que esta seja compatível com vários browsers, de maneira a que o utilizador não tenha que utilizar um determinado browser para aceder à interface *Web* do sistema.

Este ponto serve assim, para validar que a plataforma funciona em diferentes browsers, nomeadamente nos seguintes: Firefox, Safari, Windows Internet Explorer e Google Chrome.

Primeiro foi utilizado o Firefox para visualizar a plataforma (figura 5.9).

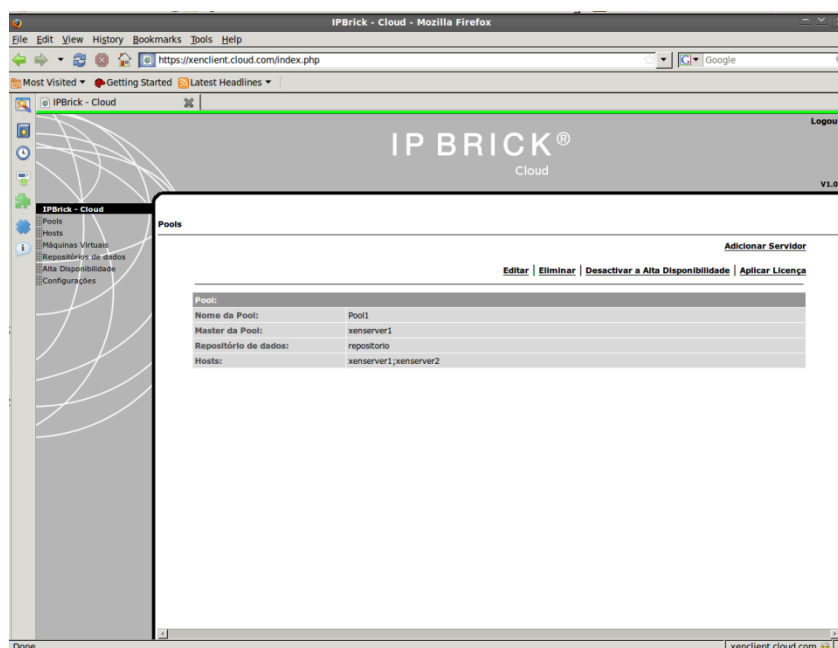


Figura 5.9 - Plataforma no browser Firefox

54 Compatibilidade

De seguida foi utilizado o Windows Internet Explorer para verificar o bom funcionamento da plataforma no mesmo (figura 5.10).

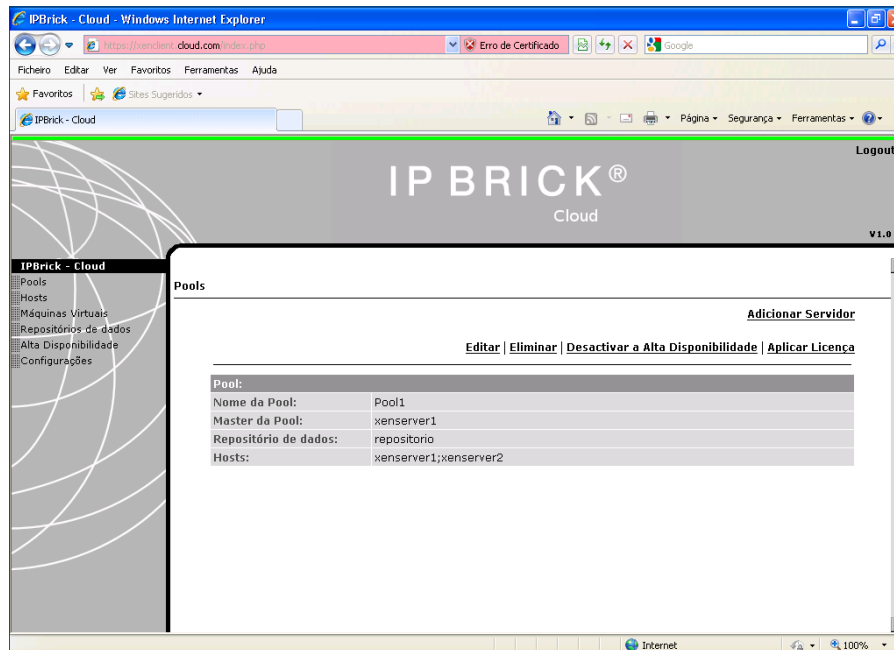


Figura 5.10 - Plataforma no browser Internet Explorer

Após o Windows Internet Explorer foi utilizado o Google Chrome (figura 5.11).

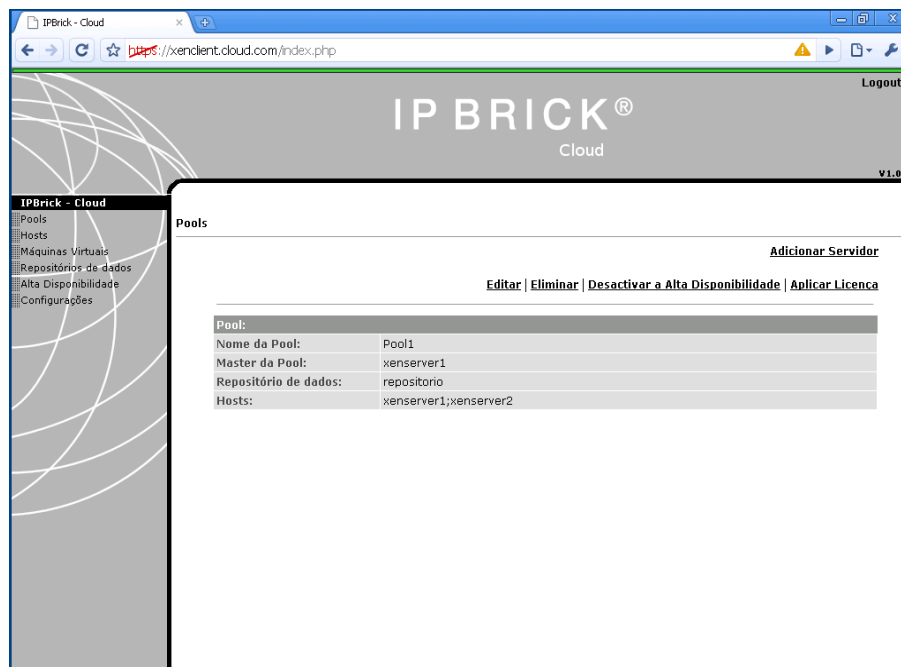


Figura 5.11 - Plataforma no browser Google Chrome

Por último foi testado no Safari (figura 5.12).

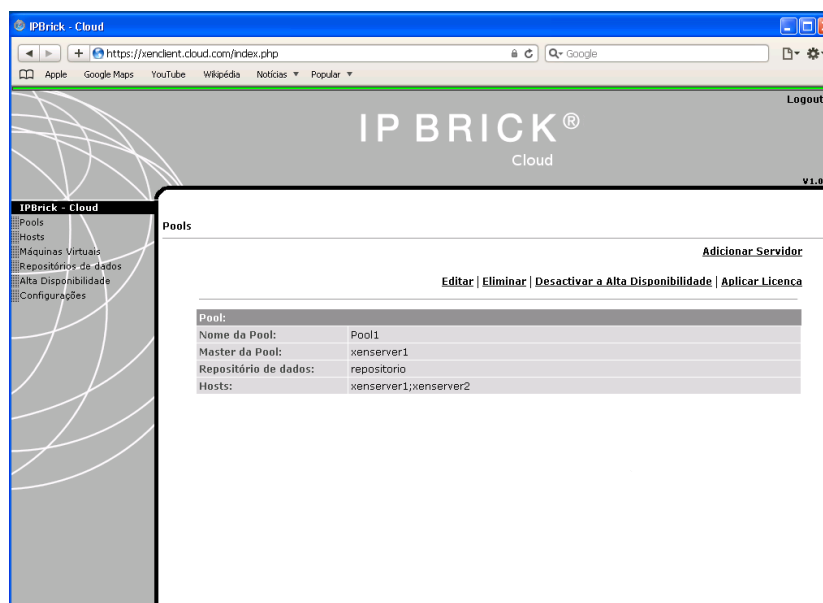


Figura 5.12 - Plataforma no browser Safari

Embora a plataforma não tenha sido testada em todos os *browsers Web* existentes, é possível concluir que não apresenta problemas de compatibilidade, pois como se pode visualizar nos testes efectuados a plataforma não apresenta qualquer tipo de incompatibilidade e isto deve-se à utilização de um design totalmente feito com CSS, disponibilizado pela empresa.

5.4 - Resultados Obtidos

Com base naquilo que foi apresentado ao longo deste capítulo, verifica-se que a plataforma apresenta todas as funcionalidades que correspondem aos requisitos estabelecidos para a mesma. A Plataforma apresenta uma interface *Webuser-friendly* para efectuar a gestão da rede, que foi programada em PHP, de forma a facilitar a configuração e gestão do sistema.

A nível de funcionalidades de gestão, foram cumpridos os objectivos relativamente à possibilidade de criar e gerir *Pool*, de adicionar servidores a essas mesmas *Pools* e ainda a criação e configuração de máquinas virtuais nos servidores adicionados.

Ainda relacionado com os serviços disponibilizados na plataforma, foi alcançada a possibilidade de adicionar repositórios às *Pools*, permitindo ao utilizador usufruir da capacidade do sistema de para além de disponibilizar repositórios para as máquinas virtuais, permite ainda a migração das máquinas virtuais, pelo facto desses mesmos repositórios serem partilhados por toda a *Pool*. Outro dos objectivos atingidos foi a nível da compatibilidade do sistema, ou seja, a possibilidade de ser acedido pelo utilizador independentemente do *browser* utilizado. E por último, foi alcançada a meta mais ambicionada, a disponibilização da funcionalidade de Alta disponibilidade.

No que toca à Alta disponibilidade, o sistema permite a configuração da mesma, ou seja, a possibilidade de especificar quais as máquinas que deverão ser reiniciadas obrigatoriamente, as que deverão reiniciar apenas se houver recursos e as que não deverão reiniciar, em caso de falha do servidor que as suporta.

A única alteração que não foi efectuada, foi ao CD-ROM de instalação, pois a instalação já era bastante simples.

5.5 - Conclusão

Neste capítulo foi analisada a solução que foi desenvolvida no âmbito desta dissertação. Foram abordados os cenários que foram considerados para efectuar os testes das funcionalidades da plataforma bem como as configurações efectuadas aos servidores para obter os serviços pretendidos. De seguida procedeu-se à análise da plataforma complementada com a apresentação das funcionalidades conseguidas. A compatibilidade da plataforma foi abordada posteriormente à apresentação dos cenários utilizados, para demonstrar que a plataforma não apresenta problemas de compatibilidade. Este Capítulo - Resultados - termina com um ponto dedicado aos resultados obtidos, ou seja, foi verificado se a plataforma corresponde ou não aos requisitos pré-definidos, que por sinal, correspondeu na perfeição.

Capítulo 6

Conclusões

6.1 - Conclusão

O aparecimento da internet veio ajudar ao processo contínuo que se vem a desenvolver ao longo do tempo da desmaterialização da informação e dos processos organizacionais para o formato digital. A consequente adesão ao formato digital e a softwares que façam a gestão desses mesmos processos vem provocar uma verdadeira reestruturação no que toca às redes empresariais. Sem dúvida que as aplicações *Web-based* e os servidores dedicados a várias áreas vieram complementar a necessidade das organizações de agregarem os seus recursos e de os ter sempre disponíveis. Contudo, esta desmaterialização de processos, faz com que aumente o número de máquinas e servidores dentro das empresas. O que por um lado foi poupar e dinamizar processos, por outro passa a ser gastar em manutenção e aquisição de hardware para suportar os seus recursos. Emerge novamente uma necessidade de desmaterialização, desta vez não é ligada aos processos, mas sim aos recursos que o fazem, ou seja, chega a hora de o espaço, condições e máquinas necessárias para tornar possível a disponibilidade destes serviços serem igualmente desmaterializados. É neste meio que surge o *Cloud Computing*.

O *Cloud Computing* é um conceito que traz às empresas a possibilidade de diminuir o número de servidores físicos. Desta forma, foi pensada e estruturada a possibilidade de desenvolver uma plataforma que assenta no conceito de *Cloud*, utilizando as ferramentas de virtualização adequando a uma solução já existente no mercado de servidores - a IPBrick.

A IPBrick Cloud Environment é um projecto que nasce no âmbito desta dissertação e que tem como objectivo a disponibilização de uma solução às empresas de gestão de *Pools*, *Hosts*, máquinas virtuais e repositórios de dados. Os requisitos que estiveram na base deste projecto foram a facilidade de instalação, configuração, acesso e ainda a alta disponibilidade do sistema. Para isso, foi desenvolvida uma plataforma que dispõe de uma interface *Web user-friendly* para a administração do sistema, seja para a configuração como para a gestão da rede. Esta plataforma teve como ferramenta de base o XenServer, que como foi referido no capítulo 3 - Revisão de Conceitos e Tecnologias, foi aquele que reuniu todos os requisitos necessários para o objectivo pretendido com o desenvolvimento desta solução. O resultado

obtido deste projecto foi a criação de uma plataforma que como foi mencionado no capítulo 5 - Resultados, foi testada em dois cenários demonstrando todas as funcionalidades da mesma, passando pela adição de *Hosts*, configuração de *Pools*, criação e gestão de máquinas virtuais e ainda a possibilidade de ter Alta disponibilidade definindo que máquinas devem ser migradas e reiniciadas em caso de falha. Em suma, é certo dizer que os requisitos definidos no capítulo 4 - Desenvolvimento da plataforma nos pontos Requisitos do sistema e Casos de Uso foram alcançados.

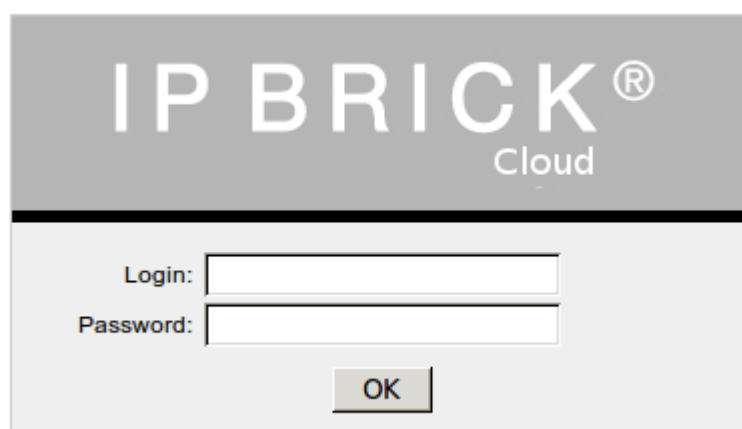
6.2 - Trabalho Futuro

No sector das tecnologias, é pertinente estar sempre um passo à frente na inovação, ou seja, uma solução nunca está na sua versão final, tendo sempre melhorias, e novas versões com novas funcionalidades e dinamização das que já existem.

O Projecto IPBrick Cloud Environment, embora tenha alcançado os objectivos principais definidos, tem com certeza passos que ainda estão por dar. Esses passos prendem-se com as seguintes funcionalidades: a possibilidade de tirar uma *Snapshot* de uma máquina virtual, e a partir dessa *Snapshot* criar uma máquina virtual; adaptar a solução paga da Citrix, tal como a Alta disponibilidade, *Workload Balacing* à plataforma; e por fim a automatização completa do CD-ROM de instalação para tornar ainda mais simples a instalação do sistema.

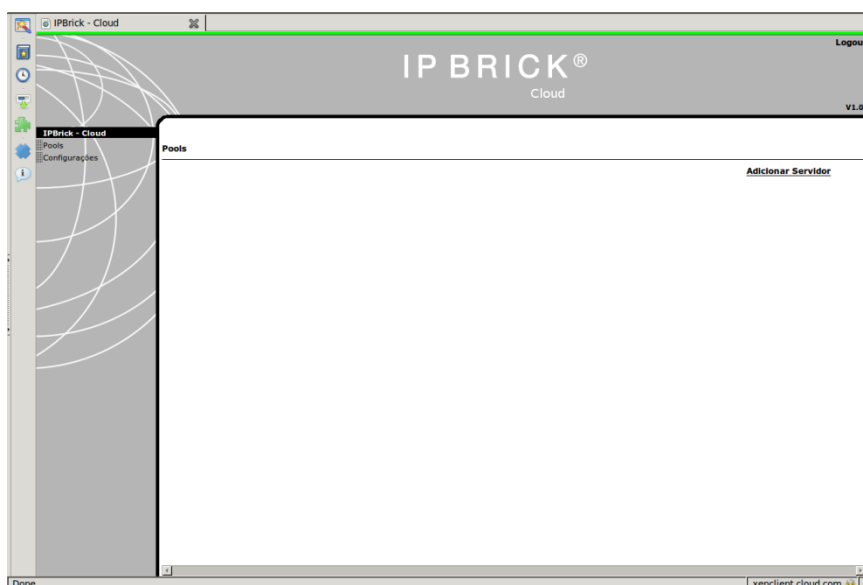
Anexo A

Neste anexo são apresentados os prints da plataforma desenvolvida para melhorar a compreensão do ponto 5.2 - Análise dos Resultados.



The image shows a login window for IP BRICK Cloud. It features a grey header with the text "IP BRICK® Cloud". Below the header, there are two input fields: "Login:" and "Password:". An "OK" button is positioned below the password field.

Anexo A 1 - Login



Anexo A 2 - Menu Inicial (Sem servidores adicionados)

Adicionar Servidor	
Login:	
Password:	
IP Address:	
OK	

Anexo A 3 - Formulário Adicionar Servidor

IP Brick - Cloud

IP BRICK® Cloud

Logout

V1.0

Configurações

Definições do acesso:

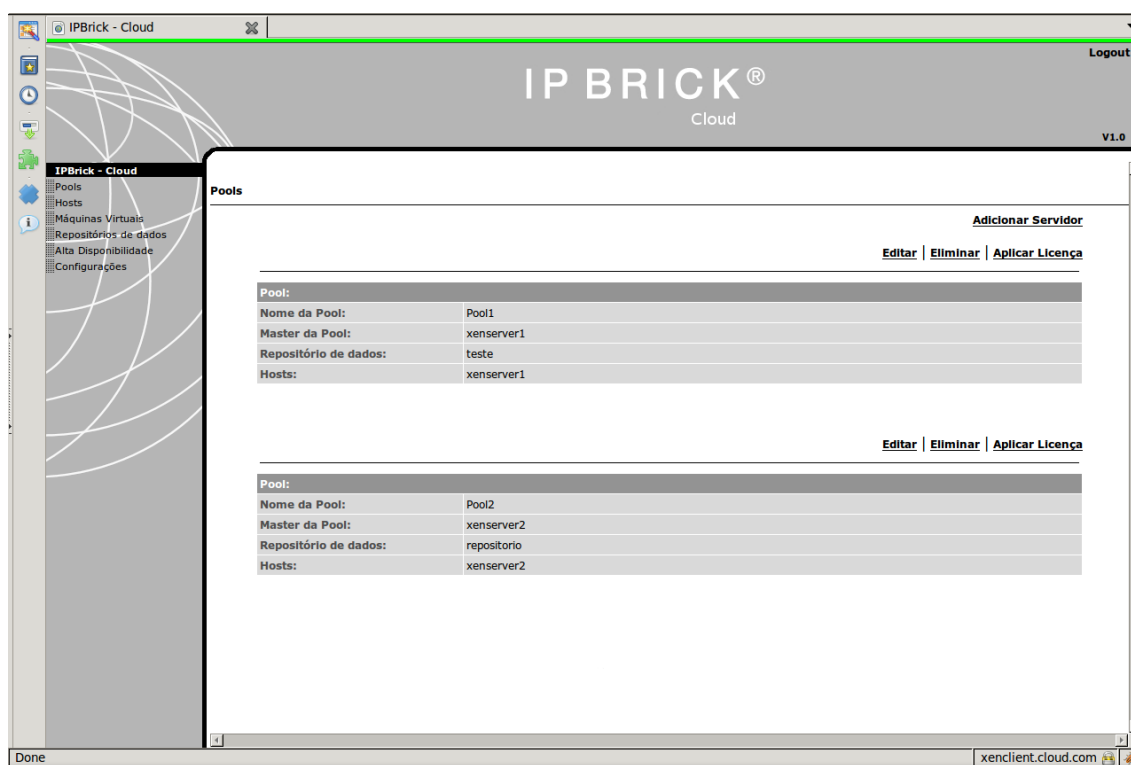
Login: administrator [Alterar](#)

Definições de utilização:

Língua: Português [Alterar](#)

Done xenclient.cloud.com

Anexo A 4- Configurações



Anexo A 5 - Pools (Com um Host em cada Pool)

Pool:	
Nome:	Pool1
Master da Pool:	xenserver1
Repositório de dados:	repositorio

OK

Anexo A 6 - Formulário Editar Pools

[Editar](#) | [Eliminar](#) | [Activar a Alta Disponibilidade](#) | [Aplicar Licença](#)

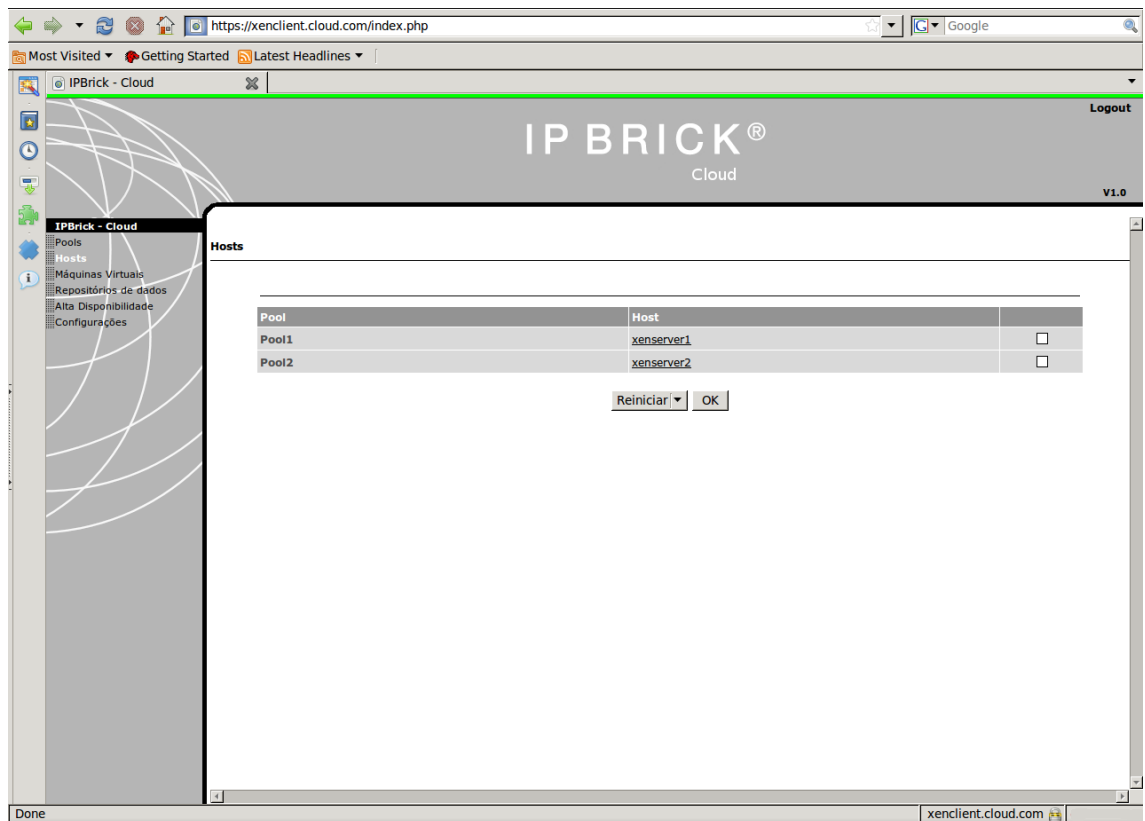
Pool:	
Nome da Pool:	Pool1
Master da Pool:	xenserver1
Repositório de dados:	teste
Hosts:	xenserver1;xenserver2

Anexo A 7 - Pool (Alta disponibilidade desactivada)

[Editar](#) | [Eliminar](#) | [Desactivar a Alta Disponibilidade](#) | [Aplicar Licença](#)

Pool:	
Nome da Pool:	Pool1
Master da Pool:	xenserver1
Repositório de dados:	repositorio
Hosts:	xenserver1;xenserver2

Anexo A 8 - Pool (Alta disponibilidade activada)



Anexo A 9 - Hosts

[Editar](#) | [Adicionar à Pool](#)

Hosts:	
Pool:	Pool1
Nome:	xenserver1
IP:	172.31.3.56
Gateway:	172.31.3.254
MAC:	00:26:18:2b:77:8b
DNS:	192.168.69.2,192.168.69.3
Máscara:	255.255.255.0
Tipo de IP:	Static
Licença:	30/03/2011
Memória Disponível:	1,59 GB (Memória Utilizada: 406,60 MB de 1,99 GB)

Anexo A 10 - Informações sobre o Host (Master e único na Pool)

[Editar](#) | [Retirar da Pool](#)

Hosts:	
Pool:	Pool1
Nome:	xenserver2
IP:	172.31.3.57
Gateway:	172.31.3.254
MAC:	90:e6:ba:61:d3:2e
DNS:	192.168.69.2,192.168.69.3
Máscara:	255.255.255.0
Tipo de IP:	Static
Licença:	30/03/2011
Memória Disponível:	1,59 GB (Memória Utilizada: 406,60 MB de 1,99 GB)

Anexo A 11 - Informações sobre o Host (Slave)

Host:	
Nome:	xenserver1
IP:	172.31.3.56
Gateway:	172.31.3.254
DNS:	192.168.69.2,192.168.69.3
Máscara:	255.255.255.0
Tipo de IP:	Estático

Anexo A 12 - Formulário de Editar Host

IP BRICK® Cloud V1.0

Logout

IPBrick - Cloud

- Pools
- Hosts
- Máquinas Virtuais
- Repositórios de dados
- Alta Disponibilidade
- Configurações

Máquinas Virtuais

[Nova Máquina Virtual](#)

Pool1

Host:	Nome:	Estado:
xenserver1	IPBrick1	● ☐

Máquinas Virtuais:

Nome:	XenServer SDK 5.5.0-24648p Import	● ☐
Nome:	IPBrick	● ☐

Pool2

Host:	Nome:	Estado:
xenserver2	IPBrick3	● ☐

Done

xenclient.cloud.com

Anexo A 13 - Máquinas Virtuais

Máquinas Virtuais:	
Pool:	Pool1 ▾
OK	

Anexo A 14-Formulário de Criar Máquina Virtual (Seleccionar Pool)

Máquinas Virtuais:	
Nome:	
Descrição:	
Memória:	Mb
Número de VCPUs:	
Host:	xenserver1 ▾
Repositório de dados:	teste ▾
Imagem:	IPBrick 5.2 ▾
Criar	

Anexo A 15-Formulário de Criar Máquina Virtual (Especificar Parâmetros)

Disconnect Options Clipboard Send Ctrl-Alt-Del

```

MOUNTING NEW FILESYSTEM...
INSTALLING...
This step can take some time please wait...
INSTALLING DRIVE /cdrom/data/drive1.dat
INSTALLING DRIVE /cdrom/data/drive2.dat

INSTALLING DRIVE /cdrom/data/drive3.dat
INSTALLING DRIVE /cdrom/data/drive4.dat
INSTALLING DRIVE /cdrom/data/drive5.dat
INSTALLING DRIVE /cdrom/data/drive6.dat
CHECKING physical address extension support...
INSTALLING boot loader...
XEN Kernel has been selected to boot by default.
Boot loader installed.
CREATING initrd for this system...
done
System is now installed.
UNMOUNTING CDROM
REMOVE CDROM BEFORE REBOOTING YOUR SYSTEM
DO NOT RESTART SYSTEM WITH IPBRICK RECOVER CD
UNMOUNTING ALL DEVICES
Installation scripts ENDED.
..

```

Reiniciar

Anexo A 16 - Criar Máquina Virtual (Instalação)

[Editar](#)

Máquina Virtual:	
Nome:	IPBrick1
Memória Total:	386,00 MB
CPUs Virtuais:	2
Tempo de funcionamento:	38 minutes

[Adicionar uma Interface de rede Virtual](#)

Interfaces de rede virtuais:				
Dispositivo:	MAC:	IP:	Activa:	
0	c6:df:83:5c:4b:50	192.168.69.199	Yes	☐
1	8e:78:1b:96:58:4e	10.0.0.253	Yes	☐

Delete

[Adicionar um Disco Virtual](#)

Discos Virtuais:				
Nome:	Descrição:	Repositório de dados:	Tamanho:	
0		teste	20,00 GB	☐

Eliminar

Anexo A 17 - Informações sobre a Máquina Virtual

Interfaces de rede virtuais:	
Rede:	Interface eth0 ▼
MAC:	☒ Auto ☐ Manual:

Adicionar

Anexo A 18 - Formulário de Adicionar uma Interface de Rede Virtual

Discos Virtuais:	
Nome do Disco Virtual:	
Descrição do Disco Virtual:	
Tamanho do Disco Virtual:	Gb
Repositório de dados:	teste ▼

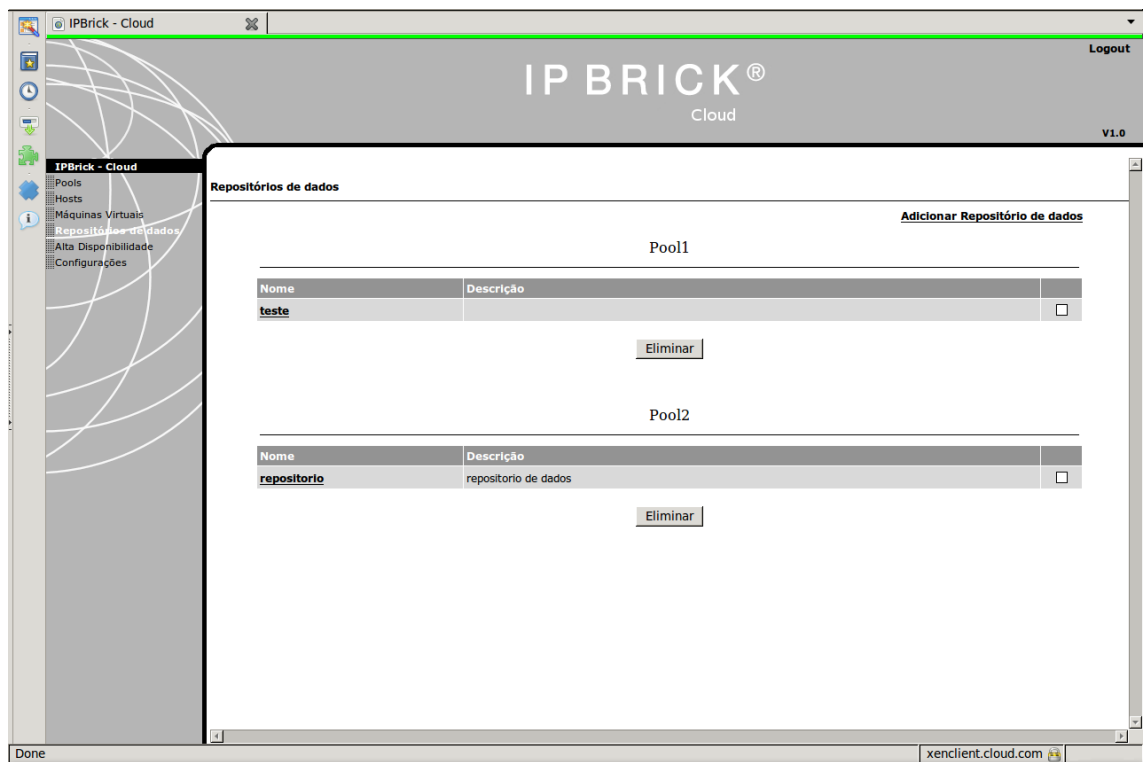
Adicionar

Anexo A 19 - Formulário de Adicionar um Disco Virtual

Máquina Virtual:	
Nome:	IPBrick1
Memória Total:	386,00 MB
CPUs Virtuais:	2

OK

Anexo A 20 - Formulário de Editar Máquina Virtual



Anexo A 21 - Repositórios de dados

Repositório de dados:	
Pool:	Pool1 ▾
Nome do Repositório:	repositorio
Descrição do Repositório:	repositorio de dados
IP do Repositório:	172.31.3.58
OK	

Anexo A 22 - Formulário de Adicionar Repositório de Dados (Especificar parâmetros)

Repositório de dados:	
Pool:	Pool1
Nome do Repositório:	repositorio
Descrição do Repositório:	repositorio de dados
IP do Repositório:	172.31.3.58
IQN do Repositório:	iqn.2010-04.com.cloud:storage ▾
OK	

Anexo A 23 - Formulário de Adicionar Repositório de Dados (Escolha do IQN)

Repositório de dados:	
Pool:	Pool1
Nome do Repositório:	repositorio
Descrição do Repositório:	repositorio de dados
IP do Repositório:	172.31.3.58
IQN do Repositório:	iqn.2010-04.com.cloud:storage
LUN do Repositório:	LUN 0 : 149,05 GB (IET) ▼

OK

Anexo A 24- Formulário de Adicionar Repositório de Dados (Escolha da LUN)

Repositório de dados:	
Nome:	repositorio
Descrição:	repositorio de dados
Espaço total:	149,04 GB
Espaço livre:	108,94 GB

Disco Virtual:	Tamanho:	Máquina Virtual:	
0	20,00 GB	IPBrick2	☐
0	20,00 GB	IPBrick3	☐

Delete

Anexo A 25 - Informações sobre o Repositório de Dados

IPBrick - Cloud

Logout

IP BRICK®
Cloud

V1.0

IPBrick - Cloud

- Pools
- Hosts
- Máquinas Virtuais
- Repositórios de dados
- Alta Disponibilidade
- Configurações

Pools

Pool:

Nome da Pool: Pool1

Alta Disponibilidade: Off

Pool:

Nome da Pool: Pool2

Alta Disponibilidade: Off

Done

xenclient.cloud.com

Anexo A 26 - Alta disponibilidade (Desactivada)

Pool:		
Nome da Pool:	Pool1	
Alta Disponibilidade:	On	

Máquinas Virtuais:		
Máquina Virtual:	Estado:	
IPBrick1	Protected	☐
XenServer SDK 5.5.0-24648p import	Protected	☐

Proteger
OK

Anexo A 27 - Alta disponibilidade (Activada)

Referências

- [1] CARISSIMI, Alexandre. Virtualização: da teoria a soluções. Minicursos do Simpósio Brasileiro de Redes de Computadores - SBRC.2008.
- [2] POLLON, Vanderlei. Virtualização de Servidores em Ambientes Heterogêneos e Distribuídos - estudo de caso - Universidade Federal do Rio Grande do Sul. Outubro de 2008.
- [3] <http://hbueno.wordpress.com/2009/04/29/virtualizacao-um-pouco-de-historia/> Consultada em 26 de Janeiro de 2010.
- [4] KLEIN, Jandir. Análise de Desempenho de Rede de Servidores Virtuais - Centro Universitário Feevale. Junho de 2008.
- [5] VIRTUALIZATION TECHNOLOGY UNDER THE HOOD. National Instruments.2009.
- [6] http://blogs.oracle.com/galdino/2009/06/paravirtualizacao_e_outros_tip.html Consultada em 27 Janeiro de 2010.
- [7] http://www.gta.ufrj.br/grad/08_1/virtual/Virtualizaototalepara-virtualizacao.html Consultada em 27 Janeiro de 2010.
- [8] <http://blogs.technet.com/lpalma/archive/2008/10/01/paravirtualiza-o-emula-o-bare-metal.aspx> Consultada em 27 de Janeiro de 2010.
- [9] SANTOS, R. C. M.; CHARÃO, A. S. Análise Comparativa de Desempenho do Hipervisor Xen: Paravirtualização versus Virtualização Total.Universidade Federal de Santa Maria.Brasil.
- [10] BADGER, Lee. Cloud Computing: Some Implications for Key Management - National Institute of Standards and Technology.8 Junho de 2009.
- [11] ABREU, Paulo; CASTRO, Vitor. Servidor de mail elástico em Cloud Computing - Universidade do Minho. 31 de Janeiro de 2009.
- [12] HAYES,Brian .News Technology /Cloud Computing - Communications of the ACM.Vol.53 No. 7. Junho de 2008.
- [13] <http://www.wikipedia.org/> Consultada em 25 Novembro 2009.
- [14] <http://www.infowester.com/cloudcomputing.php> Consultada em 23 Novembro 2009.
- [15] KIM, Won. Cloud Computing: Today and Tomorrow - Journal Of Object Technology Vol.8 No.1. Janeiro-Fevereiro de 2009.
- [16] BUYYA, Rajkumar; YEO Chee Shin; VENUGOPAL, Srikumar .Market-Oriented Cloud Computing:Vision, Hype, and Reality for Delivering IT Services as Computing Utilities-The University of Melbourne. Australia.2009.
- [17] Informação disponibilizada pela iPortalMais.
- [18] <http://br-linux.org/linux/mais-sobre-o-kvm> Consultada em 31 de Janeiro de 2010.
- [19] http://www.linux-kvm.org/page/Main_Page Consultada em 5 de Janeiro de 2010.
- [20] <http://www.vmware.com/> Consultada em 20 de Janeiro de 2010.
- [21] <http://xen.org/> Consultada em 19 de Janeiro de 2010.
- [22] <http://www.citrix.com/lang/English/home.asp> Consultada em 19 de Janeiro de 2010.
- [23] <http://www.microsoft.com/brasil/servidores/hyper-v-server/default.msp> Consultada em 10 de Junho de 2010.
- [24] Technical and commercial comparison of Citrix XenServer and VMware. Disponível na WWW. Em www.citrix.com. Consultada em 31 de Janeiro de 2010.

- [25] VMware Inc.(2009) VMware vSphere 4 Features and Benefits Compared to Microsoft Hyper-V and Citrix XenServer: VMware vSphere: The best platform for building cloud infrastructures.Palo Alto.
- [26] MATTOS, Diogo Menezes Ferrazani.Virtualização: VMWare e Xen.GTA/POLI/UFRJ.2008
- [27] http://php.net/manual/pt_BR/ Consultada em 3 de Fevereiro de 2010.
- [28] <http://www.truquesedicas.com/tutoriais/javascript/00001a.htm> Consultada em 3 de Fevereiro de 2010.
- [29] <http://pt-br.html.net/tutorials/HTML/lesson2.asp> Consultada em 3 de Fevereiro de 2010.
- [30] <http://phpxmlrpc.sourceforge.net/> Consultada em 10 de Junho de 2010.
- [31] SILVEIRA, Leonardo Salgado; PERES, André. Alta Disponibilidade em ambientes Virtualizados.ULBRA Guaíba.2008.
- [32] http://ipinfo.info/html/vnc_remote_control.php Consultada em 14 de Junho de 2010.
- [33] <http://www.realvnc.com/> Consultada em 14 de Junho de 2010.
- [34] <http://www.postgresql.org/> Consultada em 14 de Junho de 2010.
- [35] <http://www.php.net/manual/en/intro.pgsql.php> Consultada em 14 de Junho de 2010.
- [36] <http://www.xvpsource.org/> Consultada em 9 de Junho de 2010.
- [37] <http://community.citrix.com/display/xs/XenCenter> Consultada em 9 de Junho de 2010.
- [38] <http://www.xenserver5.com/xencenter.php> Consultada em 9 de Junho de 2010.
- [39] Citrix Systems, Inc. (2009). XenServer Software Development Kit Guide: 5.5.0. (1ª Edição).Flórida.
- [40] Citrix Systems, Inc. (2009). XenServer Virtual Machine Installation Guide: 5.5.0. (1ª Edição).Flórida.
- [41] Citrix Systems, Inc. (2009). XenServer Installation Guide: 5.5.0 Update 1. (1ª Edição).Flórida.
- [42] <http://forum.zwame.pt/archive/index.php/t-421012.html> Consultada em 25 de Fevereiro de 2010 e em 21 de Junho de 2010.
- [43] CHE, Jianhua [et al]. Performance Measuring and Comparing of Virtual Machine Monitors - IEEE/IFIP International Conference on Embedded and Ubiquitous Computing.2008.