

**UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**DESENVOLVIMENTO DE UMA
FERRAMENTA PARA
ADMINISTRAÇÃO DE
ARQUITETURAS VIRTUALIZADAS
BASEADAS EM XEN**

TRABALHO DE GRADUAÇÃO

Guilherme Piêgas Koslovski

Santa Maria, RS, Brasil

2007

**DESENVOLVIMENTO DE UMA FERRAMENTA
PARA ADMINISTRAÇÃO DE ARQUITETURAS
VIRTUALIZADAS BASEADAS EM XEN**

por

Guilherme Piêgas Koslovski

Trabalho de Graduação apresentado ao Curso de Ciência da Computação
da Universidade Federal de Santa Maria (UFSM, RS), como requisito
parcial para a obtenção do grau de
Bacharel em Ciência da Computação

Orientador: Prof^a Dr^a Andrea Schwertner Charão

**Trabalho de Graduação N^o 225
Santa Maria, RS, Brasil**

2007

**Universidade Federal de Santa Maria
Centro de Tecnologia
Curso de Ciência da Computação**

A Comissão Examinadora, abaixo assinada,
aprova o Trabalho de Graduação

**DESENVOLVIMENTO DE UMA FERRAMENTA PARA
ADMINISTRAÇÃO DE ARQUITETURAS VIRTUALIZADAS
BASEADAS EM XEN**

elaborado por
Guilherme Piêgas Koslovski

como requisito parcial para obtenção do grau de
Bacharel em Ciência da Computação

COMISSÃO EXAMINADORA:

Prof^a Dr^a Andrea Schwertner Charão
(Presidente/Orientador)

Prof. Dr. Marcelo Pasin (UFSM)

Prof. MSc. João Carlos Damasceno Lima (UFSM)

Santa Maria, 01 de Março de 2007.

AGRADECIMENTOS

Bom, vamos aos agradecimentos...

Após a conclusão deste trabalho, agradeço principalmente à Deus, e a todas as pessoas que de alguma forma auxiliaram na elaboração do mesmo.

Início os agradecimentos individuais lembrando de minha família em especial: pai, mãe e minha irmã, que embora distante sempre estiveram presente nos momentos difíceis. Além de agradecimento coloco este trabalho como uma forma de homenagem a minha segunda mãe, Dada (in memorian), que em vários momentos realmente conturbados me ofereceu apoio incondicional.

À minha orientadora Andrea Charão, que confiou em meu trabalho e me proporcionou diversas oportunidades de aprendizado, sempre apresentando suas sugestões construtivas, incentivando, criticando e principalmente auxiliando em todas as dificuldades que surgiram.

À todos os professores do curso pelos ensinamentos, e em especial aos professores Marcelo Pasin e João Carlos Damasceno (Caio), que participaram da banca avaliadora e contribuíram com a elaboração deste trabalho.

À minha namorada Lutielen, pelo apoio, paciência e compreensão nas horas em que passei digitando algumas coisas neste computador. Em especial, agradeço o grande incentivo nos momentos complicados e a sua participação nos momentos de felicidade ;).

À empresa Forza Informática, em especial ao Zeca, que forneceu um ótimo ambiente de trabalho durante toda a graduação, contribuindo para o meu aprendizado.

À todos os colegas de graduação, em especial ao Giovani e Eduardo, que além de colegas tornaram-se amigos, parceiros para longas cervejadas, que com certeza também contribuíram com o término de minha graduação.

Aos integrantes do LSC, em especial ao amigo Márcio pelo companheirismo na realização de diversos trabalhos e artigos.

Gracias a todos.

“Não sabendo que era impossível, ele foi lá e fez.” — JEAN COCTEAU

RESUMO

Trabalho de Graduação
Curso de Ciência da Computação
Universidade Federal de Santa Maria

DESENVOLVIMENTO DE UMA FERRAMENTA PARA ADMINISTRAÇÃO DE ARQUITETURAS VIRTUALIZADAS BASEADAS EM XEN

Autor: Guilherme Piêgas Koslovski

Orientador: Prof^ª Dr^ª Andrea Schwertner Charão

Local e data da defesa: Santa Maria, 01 de Março de 2007.

Este trabalho descreve o desenvolvimento de uma ferramenta para administração de arquiteturas virtualizadas baseadas no monitor de máquinas virtuais Xen. Utilizando Xen é possível executar diversos sistemas operacionais simultaneamente em um mesmo computador hospedeiro, permitindo um aproveitamento eficiente dos recursos disponíveis. Além disso, Xen permite realizar a migração de máquinas virtuais de um computador para outro sem interromper os sistemas em execução. O gerenciamento de ambientes baseados no monitor Xen exige a realização de diversas tarefas administrativas. Com este trabalho, pretende-se facilitar a manipulação de múltiplas máquinas virtuais nestes ambientes, uma vez que Xen originalmente não fornece ferramentas amigáveis para este fim. Desta forma, foram desenvolvidos diversos recursos administrativos, dentre eles o descobridor de máquinas virtuais e a funcionalidade de gerenciamento de grupos de máquinas virtuais. A avaliação da ferramenta desenvolvida ocorreu através de comparações na realização de diversas tarefas, inicialmente com a interface nativa de Xen e posteriormente com outras soluções existentes.

Palavras-chave: Virtualização, Xen, máquinas virtuais, administração de ambientes virtualizados.

ABSTRACT

Graduation Work
Graduate Program in Computer Science
Federal University of Santa Maria

DEVELOPMENT OF A TOOL FOR ADMINISTRATION OF VIRTUALIZED ARCHITECTURES BASED XEN

Author: Guilherme Piêgas Koslovski
Advisor: Prof^a Dr^a Andrea Schwertner Charão

The objective of this work is the development of a tool for administration of virtualized architectures that uses the virtual machine monitor Xen. With Xen is possible execute many simultaneous operating systems over the same computer, resulting in one efficient approve of the real resources. Particular, the monitor Xen allows make the virtual machine migration between two computers interconnected without stopping the applications in execution. In your native interface, Xen do not offers a friendly way for manipulate a lot of virtual machine. So, this work offers some graphical resources for manipulation of this kind of architecture.

Keywords: virtualization, Xen, virtual machines, virtualized enviromnet administration.

LISTA DE FIGURAS

Figura 2.1 – Arquitetura do monitor de máquinas virtuais Xen	15
Figura 2.2 – Exemplo da utilização da API do MMV Xen	21
Figura 3.1 – Visão geral do ambiente de execução da ferramenta	24
Figura 3.2 – Diagrama UML das classes de abstração da base de dados.....	25
Figura 3.3 – Diagrama UML simplificado dos formulários da ferramenta	26
Figura 3.4 – Diagrama UML simplificado do recurso de descoberta de hosts.....	26
Figura 3.5 – Diagrama de Seqüência do recurso de descoberta de hosts	27
Figura 3.6 – Diagrama de Seqüência da criação de uma nova máquina virtual.....	28
Figura 3.7 – Diagrama simplificado de Caso de Uso da ferramenta	29
Figura 3.8 – Representação da API XenClient.....	32
Figura 3.9 – Exemplo de abstração oferecida por XenClient	33
Figura 3.10 – Tecnologias utilizadas para armazenar os dados	34
Figura 3.11 – Interface principal da ferramenta	34
Figura 4.1 – Visualização de um grupo de máquinas virtuais	40

LISTA DE ABREVIATURAS E SIGLAS

MMV	Monitor de Máquinas Virtuais
SO	Sistema Operacional
NFS	Network File System
API	Application Programming Interface
GPL	General Public License
LDAP	Lightweight Directory Access Protocol
RPC	Remote Procedure Call
XML	Extensible Markup Language
SSH	Secure Shell
UML	Unified Modeling Language
GTK	Gimp Toolkit

SUMÁRIO

1	INTRODUÇÃO	11
2	VIRTUALIZAÇÃO DE RECURSOS COMPUTACIONAIS	13
2.1	Abordagens de virtualização	13
2.2	Monitor de máquinas virtuais Xen	14
2.2.1	Arquitetura	15
2.2.2	Migração de máquinas virtuais em Xen	16
2.3	Administração de ambientes virtualizados	17
2.3.1	Tarefas administrativas	17
2.3.2	Gerenciadores	18
2.4	API para gerenciamento de Xen	20
2.4.1	Estrutura	20
2.4.2	Implementação	21
3	FERRAMENTA DE ADMINISTRAÇÃO - <i>ADXEN</i>	22
3.1	Visão Geral	22
3.1.1	Funcionalidades	22
3.1.2	Ambiente de execução	23
3.2	Projeto	23
3.2.1	Arquitetura	23
3.3	Caso de Uso	28
3.4	Implementação	31
3.4.1	XenClient	31
3.4.2	Interface administrativa	33
3.4.3	Operações avançadas	35
4	AVALIAÇÃO	37
4.1	Operações Básicas	37
4.1.1	Comparação com a interface nativa de Xen	37
4.1.2	Comparação com ferramentas existentes	38
4.2	Operações Avançadas	39
5	CONCLUSÃO	42
	REFERÊNCIAS	44
	APÊNDICE A - PROCESSO DE INSTALAÇÃO	47
	APÊNDICE B - EXEMPLO DE UTILIZAÇÃO	49

1 INTRODUÇÃO

O conceito de virtualização de recursos computacionais, amplamente difundido nos *mainframes* IBM na década de 70, consiste basicamente na abstração do *hardware* e *software* subjacentes, permitindo a execução concorrente de diversas máquinas virtuais. Em computadores modernos, o alto desempenho e a utilização de plataformas interligadas motivou novamente a utilização desta abordagem, proporcionando o compartilhamento dos recursos através da execução simultânea de diferentes sistemas operacionais.

Em um ambiente que emprega virtualização em nível de sistema operacional, existe uma camada de *software* denominada Monitor de Máquinas Virtuais (MMV), que é responsável pelo compartilhamento, gerenciamento e alocação dos recursos existentes, garantindo independência e isolamento entre os sistemas em execução. Estas características impulsionam a utilização de virtualização em áreas como administração de recursos de redes, consolidação de servidores e balanceamento de carga entre computadores interligados (STAHL, 2005; HUANG et al., 2006).

Dentre os monitores existentes, Xen (BARHAM et al., 2003) destaca-se como um MMV de código aberto que atualmente oferece suporte para diversas famílias de processadores, como IA-32, AMD64 e EM64T, e para sistemas operacionais como Linux, FreeBSD, NetBSD e Windows XP. Além disso, Xen oferece um mecanismo de migração, denominado *live migration*, que permite a migração de máquinas virtuais entre computadores interligados, sem a necessidade da interrupção dos serviços em execução sobre o sistema operacional virtualizado.

Em ambientes virtualizados interligados, é comum realizar-se tarefas administrativas como criação, ativação, destruição, migração e monitoramento de execução de máquinas virtuais em diferentes computadores hospedeiros. Embora Xen ofereça recursos para administração local de máquinas virtuais, sua interface nativa é pouco amigável e pouco

adequada à manipulação de vários computadores hospedeiros e várias máquinas virtuais com configurações distintas.

Buscando preencher esta lacuna, alguns projetos recentes (INC, 2005; HERGERT, 2005; INC., 2006) oferecem uma interface que facilita a manipulação de máquinas virtuais. Muitas destas ferramentas, no entanto, ainda estão em um processo inicial de desenvolvimento, apresentando problemas de instalação e configuração, ou então são disponibilizadas sob licença comercial, o que dificulta sua ampla utilização em ambientes acadêmicos.

Em sua última atualização, Xen (3.0.3) inclui uma primeira versão de uma API com procedimentos que podem ser executados remotamente (MELLOR et al., 2006). Esta API foi desenvolvida para tornar-se um padrão para a execução de tarefas administrativas, sendo que sua especificação apresenta pontos fundamentais que não eram tratados em versões anteriores, como por exemplo a possibilidade de execução segura através de SSH (*Secure SHell*).

Com base neste cenário, o principal objetivo deste trabalho é o desenvolvimento de uma ferramenta para administração de arquiteturas virtualizadas que utilizam Xen. Esta ferramenta é compatível com a mais nova versão da API de Xen e visa facilitar as tarefas administrativas em ambientes interligados, permitindo um controle local ou remoto de múltiplas máquinas virtuais e múltiplos computadores hospedeiros.

O restante deste texto está organizado da seguinte forma. O capítulo 2 apresenta uma revisão da literatura sobre virtualização de recursos computacionais, apresentando o MMV Xen e discutindo questões sobre a administração de ambientes virtualizados. No capítulo 3 descreve-se a ferramenta proposta, detalhando o ambiente a que se propõe sua execução juntamente com uma descrição de seu projeto e sua implementação. O capítulo 4 apresenta uma avaliação das funcionalidades da ferramenta. Por fim, no capítulo 5 apresenta-se a conclusão do trabalho e algumas sugestões de atividades futuras.

2 VIRTUALIZAÇÃO DE RECURSOS COMPUTACIONAIS

Neste capítulo apresenta-se uma introdução à virtualização de recursos computacionais. Posteriormente, uma revisão sobre o monitor de máquinas virtuais Xen, destacando-se pontos referentes à arquitetura e tarefas de gerenciamento. Por fim, é revisado a estrutura e o funcionamento da nova API do monitor.

2.1 Abordagens de virtualização

De modo geral, a virtualização de recursos computacionais consiste na utilização de máquinas virtuais para reproduzir o conjunto de *hardware* e *software* subjacente em uma arquitetura real. Desta forma, torna-se possível mapear recursos de uma mesma arquitetura às necessidades de diferentes sistemas e aplicações, auxiliando na obtenção de flexibilidade, portabilidade e gerenciabilidade de sistemas.

Historicamente, o conceito de virtualização de recursos foi inicialmente abordado entre a década de 60 e 70, permitindo o compartilhamento e melhor aproveitamento do poder computacional dos *mainframes* IBM (GOLDBERG, 1974). Atualmente, fatores como aumento do desempenho dos computadores e utilização de ambientes interligados, impulsionaram novamente a utilização de ambientes virtualizados. Neste cenário, além de permitir um melhor aproveitamento do conjunto de recursos dos computadores modernos, a utilização de máquinas virtuais surge também como uma alternativa para lidar com a incompatibilidade entre plataformas computacionais distintas.

A execução segura e independente de múltiplas máquinas virtuais sobre uma arquitetura real é responsabilidade de um Monitor de Máquinas Virtuais (MMV). Este monitor realiza tarefas de gerenciamento e alocação de recursos, oferecendo aos sistemas em execução uma camada de abstração que representa um sistema real, semelhante a arquitetura

original. Para realizar o gerenciamento de uma forma eficiente e transparente, MMV's devem possuir o mais alto nível de acesso aos recursos subjacentes, interceptando e manipulando as instruções solicitadas pelas máquinas virtuais, garantindo a isolabilidade e escalonamento correto dos recursos reais existentes.

Monitores de máquinas virtuais são classificados de acordo com sua abordagem de implementação, dando ênfase ao posicionamento do monitor na hierarquia de níveis de execução. A classificação distingue sistemas de virtualização hospedada e sistemas de virtualização clássica (SMITH; NAIR, 2005).

A primeira abordagem (virtualização hospedada) diz respeito aos monitores de máquinas virtuais que são executados sobre um sistema operacional hospedeiro, utilizando um nível mais baixo de privilégios de execução. Neste contexto, o MMV utiliza o artifício de tradução binária em tempo de execução para adaptar as instruções das máquinas virtuais que necessitam de um maior privilégio de execução, como por exemplo acesso ao *hardware*. Esta abordagem é utilizada para implementação do MMV VMWare (SUGERMAN; VENKITACHALAM; LIM, 2001).

Já em sistemas de virtualização clássica, o MMV é executado diretamente sobre o *hardware*, possuindo o mais alto nível de privilégios de execução. Esta abordagem, tipicamente utilizada em *mainframes*, permite que o monitor intercepte, avalie e direcione as instruções para execução direta, sem a necessidade de uma tradução.

Particularmente em arquiteturas semelhantes à IA-32, sistemas implementados com a abordagem de virtualização clássica apresentam limitações, na qual um sistema operacional, embora virtualizado, ainda consegue executar diretamente instruções que necessitam de um maior nível de privilégios. Sistemas como Denali (WHITAKER; SHAW; GRIBBLE, 2002) e Xen (BARHAM et al., 2003) contornam essa limitação utilizando uma técnica denominada *paravirtualização*, na qual o sistema operacional virtualizado necessita de pequenas adaptações, devido à ligeira diferença existente entre a interface apresentada para as máquinas virtuais e a arquitetura real.

2.2 Monitor de máquinas virtuais Xen

Xen (BARHAM et al., 2003) é um monitor de máquinas virtuais de código aberto desenvolvido pela Universidade de Cambridge, baseado na abordagem de implementação clássica. Atualmente, a versão 3.0.3 de Xen oferece suporte às famílias de processadores

IA-32, AMD64 e EM64T, permitindo inclusive execução de diversos sistemas operacionais virtualizados, como Linux, NetBSD, FreeBSD e Windows XP.

A figura 2.1 representa o monitor Xen sobre um *hardware* real. O nível de abstração posicionado acima do *hardware* representa o MMV Xen propriamente dito (*hypervisor*). As máquinas virtuais alocadas sobre este nível caracterizam os domínios Xen, onde sistemas operacionais virtualizados são executados. No topo da figura encontram-se as aplicações em execução sobre os sistemas operacionais.

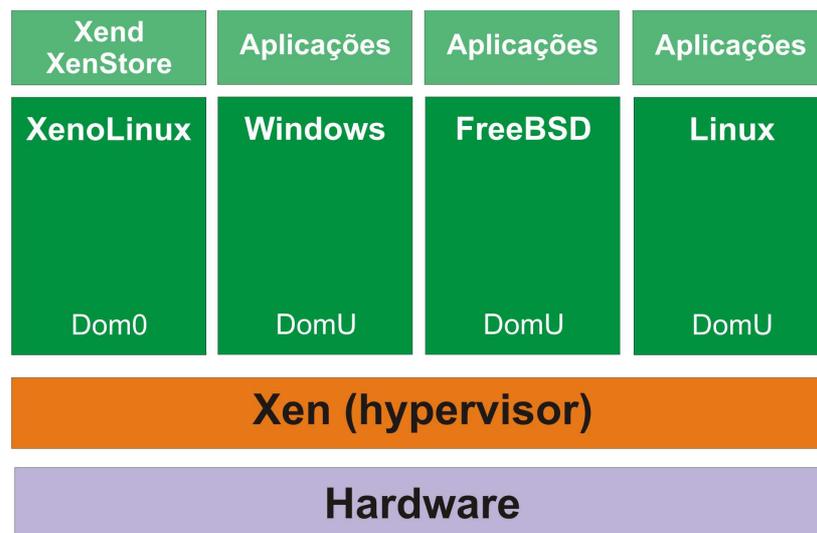


Figura 2.1: Arquitetura do monitor de máquinas virtuais Xen

2.2.1 Arquitetura

Conceitualmente, Xen apresenta uma divisão quanto a forma de implementação, na qual distingue-se especificamente o monitor de máquinas virtuais, denominado *hypervisor*, e as máquinas virtuais existentes, denominadas *domínios* Xen.

O *hypervisor* é uma camada de *software* que executa diretamente sob o *hardware*, possuindo mais alto nível de privilégio. Na abordagem de implementação utilizada em Xen, não existem mecanismos de acesso ao *hardware* diretamente no monitor, e sim um domínio administrativo, denominado *Domain0*, responsável por esta tarefa. Este domínio além de possuir o conjunto de *drivers* necessários para comunicação com o *hardware*, oferece algumas ferramentas administrativas que permitem o controle local das máquinas virtuais.

Utilizando a camada de abstração oferecida pelo *hypervisor*, os domínios Xen, que consistem em máquinas virtuais independentes, possibilitam a execução de sistemas ope-

racionais convidados de forma concorrente, sob uma mesma arquitetura. Cada máquina virtual comporta um sistema operacional virtualizado, permitindo que a execução ocorra de forma transparente e segura através do controle de execução imposto pelo *hypervisor*.

Quanto a implementação do controle de acesso à memória física, Xen utiliza um particionamento estático entre os domínios existentes. Na abordagem utilizada, cada sistema operacional, embora virtualizado, possui acesso direto ao seu gerenciador de memória para requisições de leitura. Já requisições de escritas são realizadas através de uma chamada para o *hypervisor*, que antes da sua propagação, executa verificações que garantem a isolabilidade das máquinas virtuais. Esta abordagem permite um ganho de desempenho considerável, já que a complexidade do acesso a memória física é diminuída.

2.2.2 Migração de máquinas virtuais em Xen

Xen permite o encapsulamento e migração de uma máquina virtual entre computadores interconectados, utilizando o mecanismo de *live migration*. Este mecanismo permite que um sistema operacional virtualizado migre entre computadores sem interromper a execução de seus serviços. Entretanto, para que o processo ocorra, o computador de destino da migração deve possuir no mínimo o mesmo conjunto de recursos disponíveis no computador de origem.

Live migration apresenta alguns pontos críticos de implementação, como migração do sistema de arquivos, conexões de rede e memória principal. Particularmente, o mecanismo pressupõe que os sistemas operacionais virtualizados empreguem um sistema de compartilhamento de arquivos e diretórios, tal como NFS (*Network File System*). Desta forma, não existe a necessidade de migração do sistema de arquivos, já que este é acessado remotamente.

Já a migração das conexões de rede é realizada através da propagação de um *ARP reply* (PLUMMER, 1982), informando que o endereço IP das interfaces de rede foi movido para uma nova localização, tornando inexistente a dependência residual ocasionada por redirecionamento de pacotes.

Durante o processo de migração da memória principal existe a necessidade de manutenção dos dados alterados enquanto a migração ocorre. Para possibilitar esta persistência, o mecanismo executa a migração através de cópias iterativas das páginas de memória. Em um primeiro momento, todo o conteúdo armazenado na memória principal é transferido

para o computador de destino, utilizando uma reserva de banda que permite a manutenção dos eventuais serviços em execução sobre o sistema operacional virtualizado. As demais iterações do mecanismo transferem somente o conjunto de dados alterados durante a iteração anterior.

2.3 Administração de ambientes virtualizados

Administrar um ambiente virtualizado consiste em executar procedimentos que manipulam máquinas virtuais localmente e remotamente. Esta seção descreve as tarefas que fazem parte destes procedimentos administrativos e apresenta ferramentas que buscam facilitar este processo.

2.3.1 Tarefas administrativas

Em ambientes virtualizados é comum a execução de diversas máquinas virtuais com diferentes configurações. A administração deste tipo de ambiente compreende as seguintes tarefas principais:

- instalação do monitor Xen
- configuração de *kernels* aptos à execução em um ambiente virtualizado
- criação, alocação, destruição, desativação, pause e migração de máquinas virtuais

O processo de instalação de um ambiente virtualizado baseado em Xen é trabalhoso, pois exige uma configuração correta dos parâmetros para compilação e instalação de *kernels*. Esta tarefa compreende a aplicação de *patches* para algumas funcionalidades, permitindo a comunicação entre sistema operacional hospedado e o MMV.

Ainda durante o processo de instalação é necessário configurar as imagens dos sistemas operacionais. Cada imagem deve conter o sistema de arquivos necessário para o funcionamento de um SO hospedado. Para isso, imagens são geradas utilizando uma representação através de arquivos (*loopback*) ou dispositivos de sistema.

A criação e alocação de cada máquina virtual corresponde a uma chamada para o monitor Xen solicitando a reserva de recursos e inicialização de uma imagem previamente existente. Este processo exige a existência de um arquivo de configurações referente aos recursos necessários e, um *kernel* apto a realizar chamadas para o MMV.

Nestes arquivos, parâmetros relativos ao funcionamento e representação do *hardware* são especificados, incluindo dados referentes a identificação da máquina virtual. Alguns dos parâmetros disponíveis para configuração são: ¹

- caminhos relativos para especificação da localização do *kernel*, *ramdisk* e *disk*
- endereço IP para identificação de servidores *NFS* e caminho relativo para acesso ao sistema de arquivos remotos *nfs_server* e *nfs_root*
- parâmetros para configurações de acesso à rede, como *vif*, *dhcp*, *netmask*, *gateway* e *hostname*
- configurações referentes à utilização de memória física (*memory*)

O gerenciamento da migração de máquinas virtuais é executado diretamente sobre o domínio administrativo do computador de origem. Para que este processo ocorra, o sistema arquivos deve ser compartilhado entre todos os *hosts* e o administrador deve conhecer previamente os recursos do computador de destino e verificar a compatibilidade antes da migração. Após a execução do processo, informações sobre a máquina virtual são disponibilizadas apenas localmente no computador de destino.

Em seu domínio administrativo, Xen oferece nativamente apenas mecanismos de administração local de máquinas virtuais. Em ambientes interligados a complexidade de realização destas tarefas aumenta, pois existe a necessidade de conexão via terminais remotos. Neste contexto, tarefas simples de administração como destruição, desativação e pausa de máquinas virtuais tornam-se trabalhosas, pois exigem que o administrador conheça o número de identificação da máquina virtual alvo.

2.3.2 Gerenciadores

Buscando facilitar as tarefas de administração de ambientes virtualizados, diversos projetos estão em desenvolvimento, oferecendo ferramentas distintas que permitam administrar recursos locais e remotos (XEN-TOOLS.ORG, 2005; HAZARD; JEDI, 2006; HERGERT, 2005; INC., 2006; INC, 2005; VIRT-MANAGER VIRTUAL MACHINE MANAGER, 2006). A seguir apresentam-se os projetos que possuem um estágio de desenvolvimento mais avançado.

¹Além destes parâmetros mínimos necessários, configurações específicas podem ser realizadas através da utilização completa das opções disponíveis no manual de utilização de Xen (BARHAM et al., 2003).

- XenManager

Este projeto (HERGERT, 2005) é mantido pelas empresas *Mosaix Communications* e *Medsphere* sendo distribuído sob licença GPL. Atualmente, XenManager oferece facilidades administrativas como criação, destruição e ativação de máquinas virtuais localmente. Por estar em seu estágio inicial de desenvolvimento, este projeto ainda não apresenta alguns recursos como manutenção do cadastro dos *hosts* virtualizados, sendo que em cada utilização se faz necessário uma reconfiguração do ambiente de trabalho.

- Xen Enterprise

Xen Enterprise (INC., 2006) trata-se de uma ferramenta desenvolvida pela XenSource Inc., principal empresa desenvolvedora de Xen. Esta ferramenta apresenta um conjunto completo de soluções para administração de ambientes virtualizados baseados em Xen, possuindo interfaces administrativas para plataformas Linux e Windows. Este pacote proprietário apresenta soluções inovadoras, como a possibilidade de conversão de uma máquina física em uma máquina virtual (*Physical to Virtual (P2V)*).

- Enomalism

Enomalism (INC, 2005) é uma solução distribuída livremente, que permite a administração de ambientes virtualizados. Desenvolvida pela *Enomaly Labs*, esta ferramenta possui uma interface *web* que permite a manipulação de máquinas virtuais, oferecendo as tarefas administrativas comuns, como criação, destruição, ativação e migração.

O processo de instalação de Enomalism é uma tarefa que exige conhecimento sobre o funcionamento do mecanismo de autenticação LDAP, já que toda a interface baseia-se em seu funcionamento. Atualmente, este projeto exige uma reestruturação devido às alterações ocorridas na interface de comunicação remota de Xen.

- Virt-Manager

Desenvolvida pela *RedHat*, Virt-Manager (*Virtual Machine Manager*) (VIRT-MANAGER VIRTUAL MACHINE MANAGER, 2006) é uma ferramenta para administração de máquinas virtuais que oferece uma interface gráfica na qual permite a administração e acompanhamento de informações estatísticas referentes às máquinas virtuais.

Diferente das outras ferramentas, Virt-Manager foi desenvolvida baseada na biblioteca de virtualização LibVirt (LibVirt - VIRTUALIZATION API, 2006), que é uma biblioteca desenvolvida com o propósito de abstrair o monitor de máquinas virtuais.

Dentre os projetos citados, fatores como complexidade de instalação, abrangência parcial das tarefas administrativas e alterações de projeto realizadas pelos desenvolvedores de Xen, limitam a ampla utilização destas ferramentas. Em sua versão atual (3.0.3), Xen apresentou uma nova especificação para uma API (MELLOR et al., 2006) que permite a comunicação remota com domínios administrativos. Esta nova API não oferece suporte para chamadas que não correspondam ao protocolo especificado, comprometendo assim o funcionamento das ferramentas existentes atualmente.

2.4 API para gerenciamento de Xen

A API para gerenciamento de Xen foi implementada utilizando chamadas de procedimentos remotos (RPC) (NELSON, 1981), e permite a manipulação de máquinas virtuais alocadas em computadores interligados. A seguir descreve a estrutura da API e posteriormente o modelo de implementação utilizado.

2.4.1 Estrutura

O funcionamento da API exige que todos os domínios administrativos de Xen (*Domain0*) executem um servidor de chamadas de procedimentos remotos, no qual previamente o *daemon* principal de Xen (*xend*) registra quais procedimentos podem ser acessados. Dentre estes procedimentos ocorre uma distinção quanto à origem das informações, agrupados em informações de computadores hospedeiros (*hosts*) e máquinas virtuais alocadas.

As informações sobre os *hosts* retornam valores referentes aos recursos disponíveis (incluindo sua utilização atual) e uma listagem das máquinas virtuais existentes. Já para as máquinas virtuais são disponibilizadas informações virtuais, relativas às configurações utilizadas na inicialização. Ainda na versão atual da API, estão disponíveis tarefas de criação, destruição, pausa, migração e reconfiguração de máquinas virtuais ativas.

Dentre as especificações, existe a possibilidade de conexão de forma segura, através da utilização de autenticação via SSH. Desta forma, as chamadas remotas somente serão executadas se o usuário possui o nível correto de permissões, de acordo com a configura-

ção do ambiente de trabalho. Outra opção disponível consiste na utilização do mecanismo de chamada sem verificações de segurança, onde qualquer processo torna-se habilitado a efetuar a comunicação através de uma determinada porta.

2.4.2 Implementação

Chamada de procedimentos remotos (RPC), é um recurso amplamente abordado em ambientes distribuídos, em que existe a necessidade de execução de procedimentos alocados em computadores distintos (BIRRELL; NELSON, 1984). Este paradigma consiste basicamente na realização de pedidos de execução para procedimentos disponíveis entre computadores interligados, sendo executado de forma síncrona ou assíncrona, de acordo com a implementação.

A implementação de RPC utilizando as especificações da linguagem XML, denominado XML-RPC (COM., 1999) baseia-se na necessidade de interligar plataformas distintas, independentes de sistemas operacionais, famílias de processadores e protocolos de comunicação. XML-RPC, utiliza marcadores para especificar parâmetros e respostas a procedimentos remotos, caracterizando-se por ser uma especificação de fácil implementação, já que utiliza um alto nível de abstração. Estas características justificam a escolha de XML-RPC para implementação da API do monitor Xen.

Uma exemplificação da utilização da API é demonstrada a seguir (utilizando a linguagem de programação Python).

```
...
self.server = ServerProxy("ssh://user@host/RPC2", None, None, 1, 1)
...
return self.server.xend.domains(0)
```

Figura 2.2: Exemplo da utilização da API do MMV Xen

A figura 2.2 exemplifica a conexão com um domínio administrativo de Xen utilizando autenticação através de SSH. Após a conexão, ocorre o pedido de execução de um procedimento remoto que retorna os domínios existentes neste *host*. Neste exemplo é apresentado uma utilização simples da API, já sua utilização completa requer um conhecimento dos parâmetros e tratamento de erros de cada procedimento.

3 FERRAMENTA DE ADMINISTRAÇÃO - *ADXEN*

Este capítulo descreve a ferramenta de administração desenvolvida neste trabalho, denominada *AdXen*. Inicialmente apresenta-se uma visão geral da ferramenta, descrevendo-se suas funcionalidades e seu ambiente de execução. Na seqüência, apresenta-se as principais etapas e decisões dos processos de projeto e implementação.

3.1 Visão Geral

3.1.1 Funcionalidades

Todas as funcionalidades previstas para a ferramenta visam facilitar tarefas administrativas em ambientes virtualizados baseados em Xen, isto é, ambientes compostos por diversos computadores interligados executando Xen. Tais funcionalidades são divididas em operações básicas e operações avançadas de administração, que estão disponíveis ao administrador através de uma interface gráfica.

Como operações básicas, entende-se as tarefas apresentadas no capítulo 2 (seção 2.3), que compreendem a criação, a destruição, a desativação, a pausa/retomada e a migração de máquinas virtuais.

Como operações avançadas, têm-se a administração de grupos de máquinas virtuais e descoberta de *hosts* virtualizados. A administração de grupos permite executar operações simultaneamente em diversas máquinas virtuais. A principal motivação para este recurso é permitir o gerenciamento integrado de máquinas virtuais que executam serviços e aplicações inter-relacionados. Tal recurso pode ser útil, por exemplo, ao se fazer experimentos com um *middleware* de computação em grade, que comumente requer a instalação e configuração de diversos serviços em cada *host* da grade. Com o uso de virtualização e do recurso de administração de grupos, pode-se criar e operar diversas máquinas virtuais a partir de imagens e configurações geradas previamente.

A descoberta de *hosts*, por sua vez, consiste na identificação dos computadores de uma rede que possuem o MMV Xen ativo. Esta funcionalidade facilita o processo de configuração inicial do ambiente de execução, auxiliando o administrador no controle dos *hosts* virtualizados.

3.1.2 Ambiente de execução

No ambiente de execução de *AdXen*, pressupõe-se a existência de um servidor que armazena imagens de máquinas virtuais e arquivos de configuração da ferramenta. Estes arquivos devem estar acessíveis a todos os *hosts* através de um serviço de compartilhamento de arquivos e diretórios.

Outro pré-requisito para o ambiente de execução é um sistema de autenticação via SSH previamente configurado em todos os *hosts* envolvidos. Além disso, na versão atual de Xen as tarefas administrativas somente são realizadas por usuários que possuem privilégios de super-usuário, conseqüentemente existe a necessidade da utilização de um ambiente confiável, onde o sistema de permissões deve estar configurado de forma correta e segura.

A figura 3.1 apresenta uma visão geral de *AdXen* em execução sobre um ambiente virtualizado. No topo está localizada a ferramenta e ao seu lado o banco de imagens acessível a todos os *hosts* virtualizados. Já na parte inferior existe a representação de computadores distintos interligados em rede executando um ou mais sistemas operacionais virtualizados.

Ainda na figura 3.1, as setas representam o fluxo de comunicação entre as API's. Um comando de gerenciamento do ambiente executado a partir da interface administrativa, é imediatamente repassado para a API XenClient que executa o pedido de execução remota no *host* correspondente utilizando a API do monitor Xen.

3.2 Projeto

Esta seção descreve as etapas de projeto de *AdXen*, destacando a arquitetura da ferramenta através de diagramas UML.

3.2.1 Arquitetura

Esta etapa do projeto buscou identificar a arquitetura das funcionalidades administrativas previstas. Compreende-se como arquitetura itens como: a identificação de dados que devem ser armazenados, diagramas de classes principais, diagramas de seqüência de

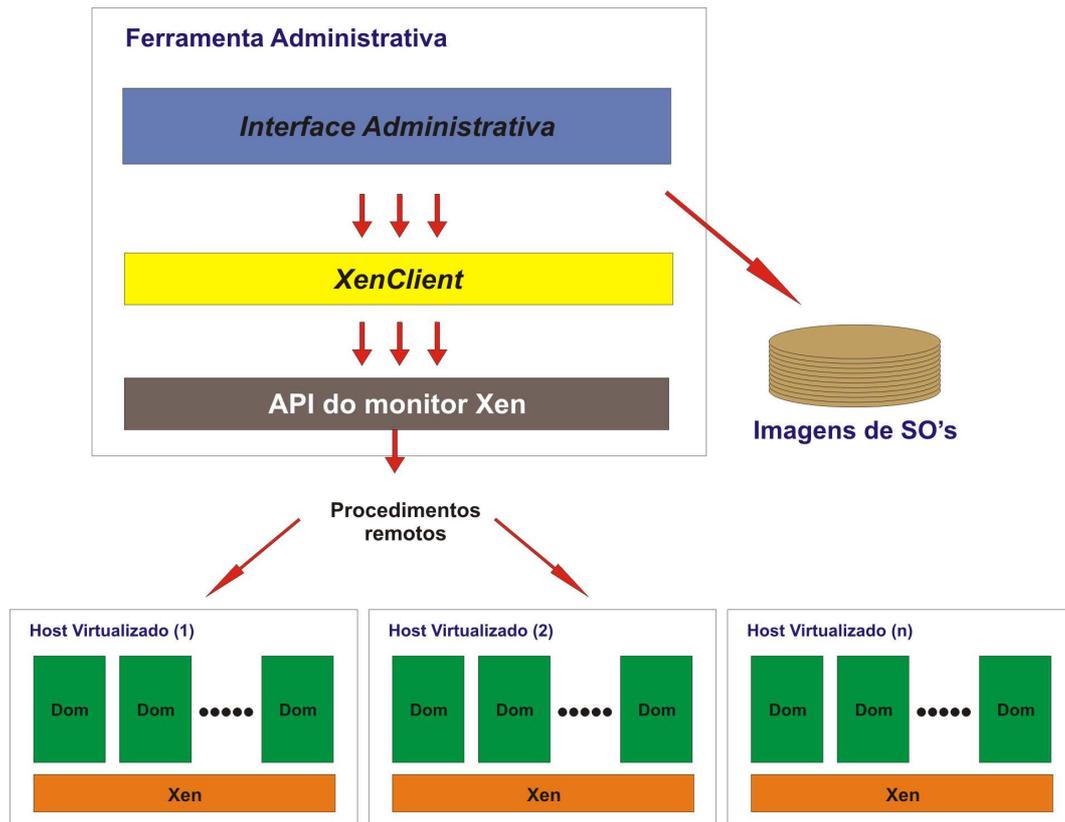


Figura 3.1: Visão geral do ambiente de execução da ferramenta

utilização de recursos e mecanismo de interação com o administrador do ambiente.

Inicialmente, descreve-se o conjunto de dados necessários para controlar o ambiente de execução da ferramenta.

A figura 3.2 apresenta as classes utilizadas para abstrair a base de dados. Nesta representação, as setas pontilhadas indicam uma dependência funcional, caracterizando a origem dos dados representados pelas classes. As setas contínuas representam as dependências relacionas, identificando os relacionamentos de dados existentes entre as classes. A seguir, resumem-se quais conjuntos de dados são representados em cada classe identificada.

- **Preferences:** armazena as informações de configuração da ferramenta, como tempo de atualização na visualização de *hosts* e domínios
- **Hosts:** utilizada para identificar quais computadores pertencem ao ambiente de execução.
- **MachineGroup:** armazena o nome e uma descrição sobre grupos de máquinas virtuais

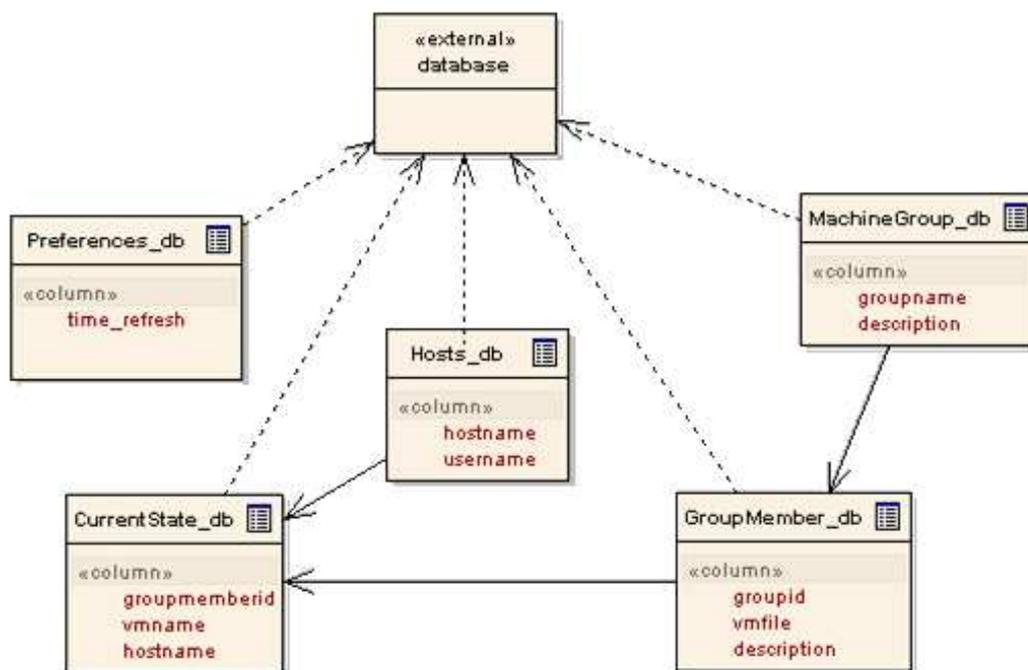


Figura 3.2: Diagrama UML das classes de abstração da base de dados

- GroupMember: identifica quais arquivos de configuração fazem parte de um determinado grupo
- CurrentState: classe responsável por armazenar o estado atual do ambiente de trabalho

Após esta etapa, buscou-se a identificação da forma como o administrador iria interagir com a ferramenta. Para isso, agrupou-se as funcionalidades propostas em formulários específicos, de acordo com a tarefa administrativa.

A figura 3.3 mostra o diagrama UML simplificado dos formulários que interagem com o administrador. As setas indicam as dependências entre os formulários, descrevendo a ordem em que o acesso é realizado. Inicialmente, acessa-se o formulário principal (*MainForm*) por onde pode-se acessar as demais funcionalidades da ferramenta.

Na etapa seguinte no projeto de *AdXen*, identificou-se as classes principais necessárias para o funcionamento da ferramenta. Neste ponto ocorreu uma atenção especial para a representação das funcionalidades avançadas propostas. Para isso, utilizou-se diagramas de classes e diagramas de seqüência.

A figura 3.4 apresenta o diagrama UML simplificado do recurso de descoberta de *hosts*. Neste diagrama observa-se a existência das classes necessárias para o funciona-

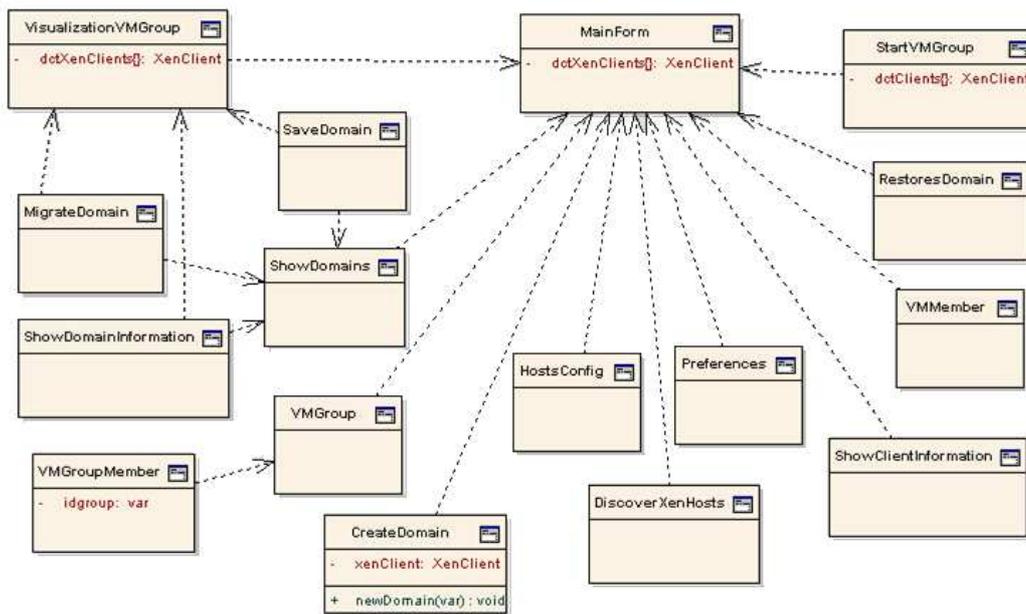


Figura 3.3: Diagrama UML simplificado dos formulários da ferramenta

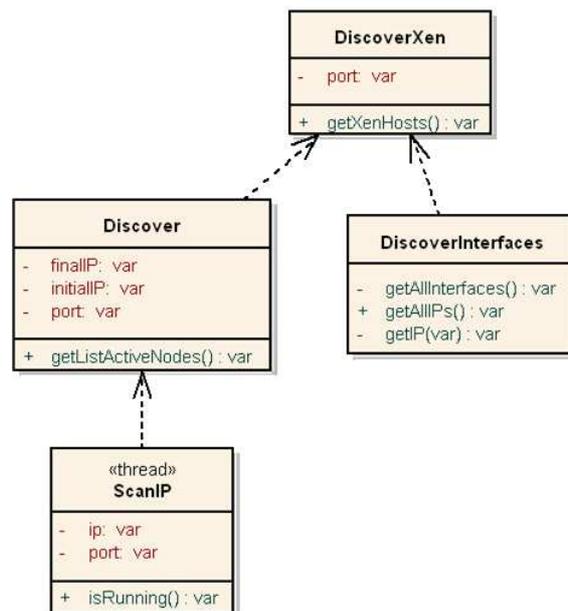


Figura 3.4: Diagrama UML simplificado do recurso de descoberta de hosts

mento do mecanismo (*ScanIP*, *Discover*, *DiscoverInterfaces* e *DiscoverXen*). As setas pontilhadas identificam a dependência funcional das classes.

Para exemplificar o funcionamento do mecanismo de descoberta de *hosts*, o diagrama de seqüência 3.5 apresenta a forma como o administrador pode acessar e utilizar este recurso. Inicialmente o administrador do ambiente acessa o formulário que oferece esta funcionalidade. Neste formulário, existe uma instância da classe *DiscoverXen*, onde realiza-

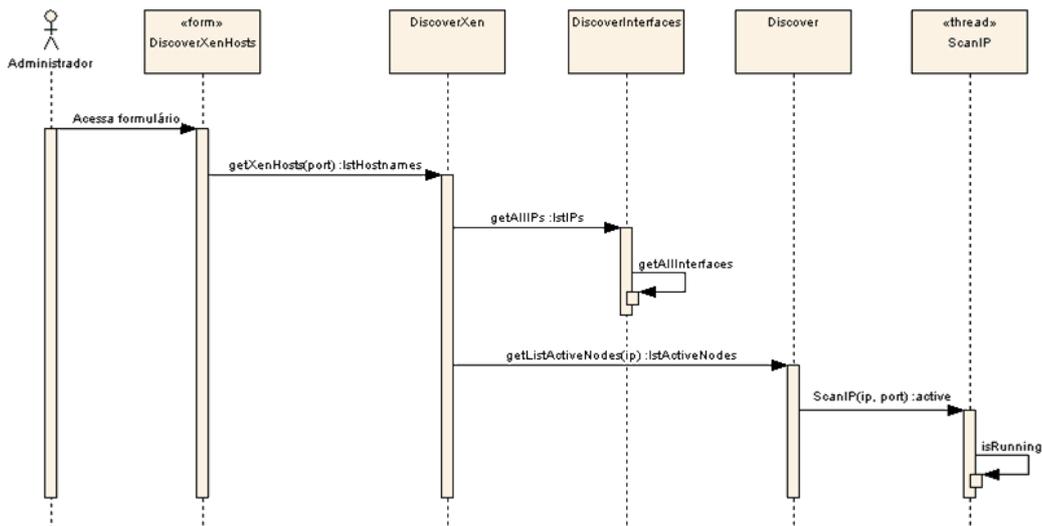


Figura 3.5: Diagrama de Seqüência do recurso de descoberta de hosts

se uma chamada para o método *getXenHosts(port)*.

Esta invocação inicial aciona o mecanismo de procura que conceitualmente divide a pesquisa em duas etapas. Em um primeiro momento, uma chamada ao método *getAllIPs()* disponível na classe *DiscoverInterfaces* retorna os endereços IP existentes no computador hospedeiro da ferramenta. A segunda etapa consiste na pesquisa por *hosts* ativos nas interfaces de rede identificadas na etapa anterior. Neste ponto, ocorre a inicialização de um objeto *Discover* e a realização de chamadas para o método *getListActiveNodes(ip)* informando as interfaces identificadas.

A classe *Discover* inicializa pesquisas individuais simultâneas para todos os endereços da interface. Este procedimento ocorre através da criação de *threads* contendo o objeto *ScanIP*, que através do método *isRunning()* identificam se existe um MMV Xen ativo no endereço especificado. Para isso, realiza-se uma tentativa de conexão através de *sockets* na porta de destino informada na criação do objeto.

Ao término deste processo, o administrador recebe na interface do formulário uma lista contendo os *hosts* identificados.

Além de analisar a forma de representação dos recursos avançados, o projeto de *AdXen* compreendeu uma análise dos principais recursos básicos, como por exemplo a criação e a migração de uma máquina virtual. A figura 3.6 apresenta um dos diagramas de seqüência resultante deste processo.

Inicialmente o administrador seleciona no formulário principal de ferramenta o *host* de destino para a criação do novo domínio. Neste ponto, ocorre a verificação por parte do

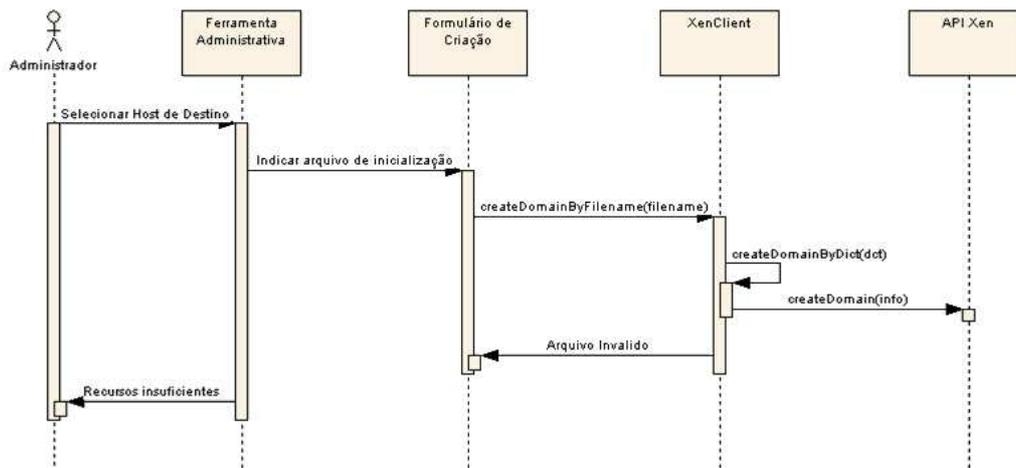


Figura 3.6: Diagrama de Seqüência da criação de uma nova máquina virtual

administrador, da disponibilidade de recursos no *host* selecionado. Considerando a existência de recursos disponíveis, inicializa-se o formulário de criação de novas máquinas virtuais, onde o administrador deve informar o arquivo de configuração para inicialização do respectivo domínio.

Neste momento, é realizado uma chamada para o método de criação de domínios da classe *XenClient* (capítulo 3 seção 3.2.1) informando o arquivo selecionado. *XenClient* verifica o formato do arquivo, transformando seu conteúdo em um dicionário de dados que posteriormente é passado como parâmetro para a API do monitor Xen, solicitando a inicialização da nova máquina virtual.

3.3 Caso de Uso

Para exemplificar a utilização das operações básicas e avançadas em um ambiente, esta seção apresenta um diagrama UML representativo de um caso de uso, descrevendo uma seqüência de atividades capazes de auxiliar as tarefas de administração. Consideramos um ambiente de experimentação onde diversos testes são realizados sobre sistemas que possuem configurações distintas, mas utilizam os mesmos recursos de *hardware* existentes.

Um exemplo de ambiente com estas características é observado em laboratórios de computação, onde muitas vezes os *softwares* são totalmente ou parcialmente reconfigurados para a realização de testes. A troca de distribuições de sistemas operacionais e reconfiguração de versões de programas como máquinas virtuais Java ou *middlewares*

específicos ocorre frequentemente, tornando-se uma tarefa trabalhosa.

Considerando que a opção selecionada para facilitar a reconfiguração dos equipamentos foi a adoção de uma plataforma virtualizada, utilizando o MMV Xen, descreve-se a seguir as tarefas onde a utilização da ferramenta pode auxiliar o administrador do ambiente.

O ponto inicial de partida é a existência de um ambiente no qual diversos computadores executam o MMV Xen e o compartilhamento de informações é realizado através de SSH, com uma configuração segura que permita acesso remoto de usuários com alto privilégio administrativo.

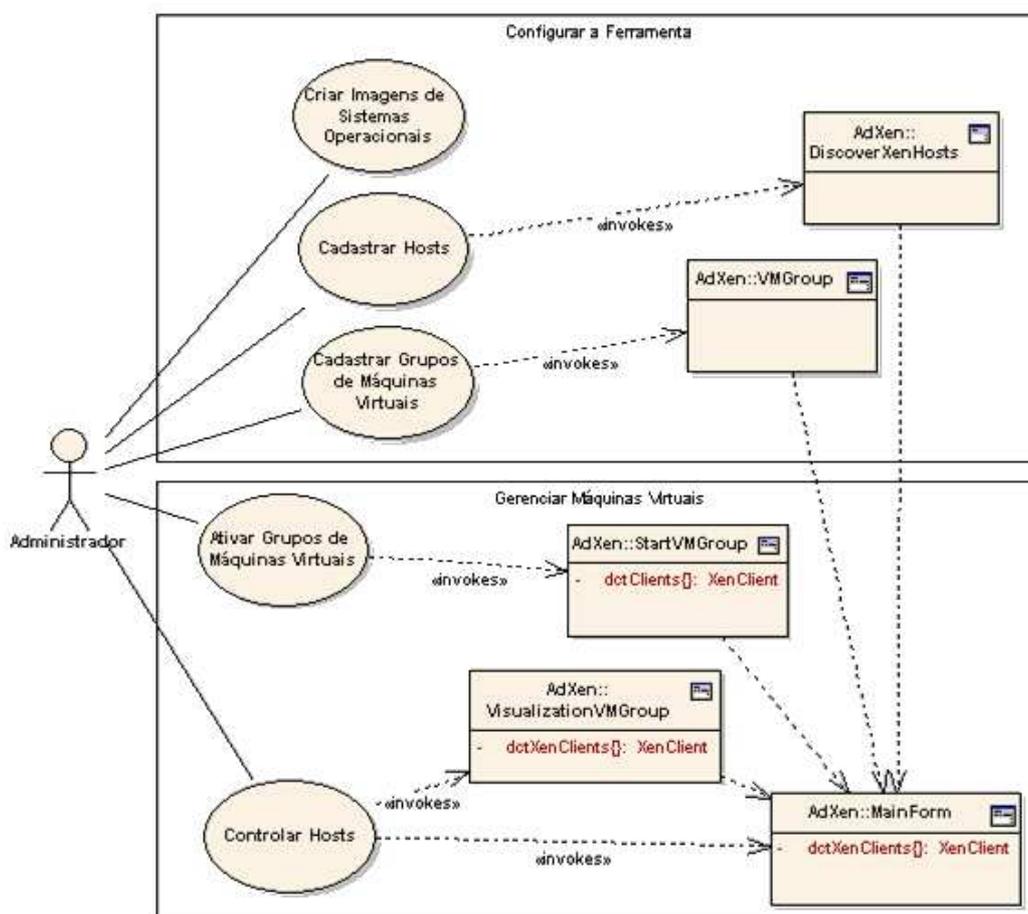


Figura 3.7: Diagrama simplificado de Caso de Uso da ferramenta

A figura 3.7 apresenta o diagrama UML de caso de uso de alguns recursos da ferramenta *AdXen*. As atividades selecionadas para exemplificação neste caso de uso representam os pontos diferenciais da ferramenta quando comparada as demais. Neste diagrama as elipses representam as atividades realizadas pelo autor, ou seja, o administrador do sistema. As setas com a etiqueta *invokes* representam os formulários que devem ser utili-

zados para realização da atividade e as demais setas representam a seqüência necessária para acesso aos formulários do sistema, conforme descrito na figura 3.3 apresentada no capítulo 3 seção 3.4.2.

Os passos seguintes representam a descrição conceitual do diagrama apresentando:

- Configuração da ferramenta
 1. Administrador configura imagens de sistemas operacionais que serão hospedados
 2. Administrador efetua o cadastro dos *hosts* que participam do ambiente
 3. Administrador cadastra as informações sobre os grupos de máquinas virtuais
- Gerenciamento de máquinas virtuais
 1. Administrador torna um grupo de máquinas virtuais ativo
 2. Administrador monitora grupos de máquinas virtuais previamente ativado, verificando *hosts* individualmente ou através da visualização de grupos

Para exemplificar de forma mais precisa, descreve-se a seguir o processo de realização das etapas, apresentando pontos onde facilidades são oferecidas para o administrador do sistema.

Primeiramente, o administrador deve criar imagens de sistemas operacionais com as suas configurações necessárias e disponibilizar para todos os computadores através de um sistema de compartilhamento de arquivos e diretórios como NFS. Além disso, *kernels* aptos a execução sobre uma arquitetura virtualizada devem ser preparados e disponibilizados para os *hosts* ativos.

Conforme a especificação da API de Xen, a inicialização de novos domínios requer arquivos de configuração, onde informações como configuração de rede, memória, forma de acesso ao sistema de arquivos, entre outras, devem ser informadas. Estes arquivos devem ficar acessíveis para o computador que executará a ferramenta administrativa.

A primeira execução de *AdXen* sobre o ambiente em questão, requer o cadastro dos *hosts* que serão monitorados. Para isso, o administrador pode utilizar o recurso *Xen Discover* para descobrir e cadastrar todos os hospedeiros encontrados. Neste ponto do processo, a interface principal da ferramenta já permite ao administrador acompanhar a ativação e desativação de qualquer hospedeiro, e obter informação sobre seus recursos disponíveis.

Para facilitar a inicialização de uma configuração específica, o administrador pode criar os grupos de máquinas virtuais e informar para a ferramenta a localização dos arquivos de configuração. Esta tarefa é opcional ao administrador já que máquinas virtuais podem ser criadas e manipuladas individualmente.

Terminado o processo inicial de configuração, um ou mais grupos de máquinas virtuais podem ser inicializados, considerando os recursos de *hardware* existentes no ambiente. O administrador pode optar entre uma visualização global dos *hosts* ativos ou uma visualização individual de um grupo específico, onde as operações de manipulação de domínios são disponibilizadas.

3.4 Implementação

Atualmente, a API de Xen está disponível nas linguagens C (RITCHIE; KERNIGHAN; LESK, 1975) e Python (The Python Language, 1990). Para implementação da ferramenta, optou-se pela utilização de Python em sua versão 2.5, somada a bibliotecas adicionais distribuídas sob licença GPL.

O processo de desenvolvimento da ferramenta foi dividido em duas etapas. Primeiramente, implementou-se uma API denominada *XenClient* (figura 3.8) que encapsula e abstrai as chamadas remotas para Xen. Posteriormente, implementou-se uma interface gráfica que facilita a realização de configurações de domínios. As seções seguintes apresentam maiores detalhes sobre estas etapas de desenvolvimento.

3.4.1 *XenClient*

A API *XenClient* foi desenvolvida buscando dois objetivos principais: simplificar o acesso as chamadas da API do monitor Xen e oferecer uma camada de abstração entre a interface gráfica da ferramenta e o monitor de máquinas virtuais.

O uso da API originalmente oferecida por Xen exige muitas verificações de parâmetros e códigos de retorno. Buscando simplificar o uso desta API, desenvolveu-se a API *XenClient*, que abstrai detalhes da realização das chamadas para o MMV.

A figura 3.8 caracteriza o posicionamento de *XenClient*, localizada abaixo da interface gráfica, abstraindo as chamadas para a API do monitor Xen. Desta forma, o desenvolvimento da interface gráfica foi totalmente baseado sobre as chamadas de *XenClient*, tornando a ferramenta independente e facilmente adaptável as alterações que porventura

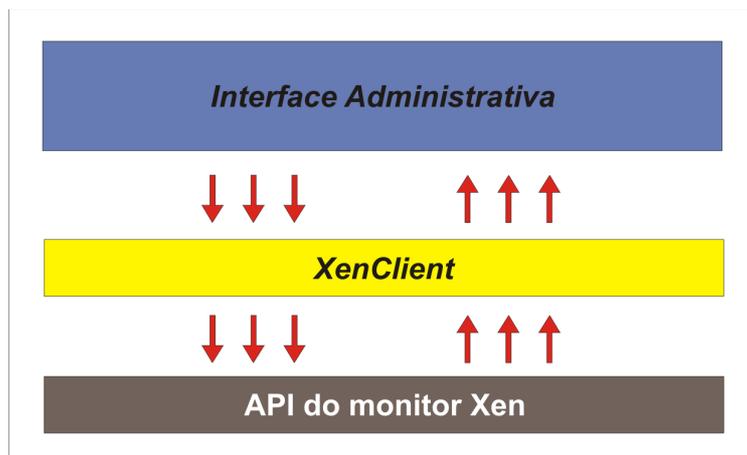


Figura 3.8: Representação da API XenClient

surgirem na API de Xen.

Dentre as funcionalidades oferecidas por XenClient, encontram-se chamadas que permitem a administração de domínios Xen como criação, destruição, desativação, migração, pausa e retomada da execução de máquinas virtuais, além de chamadas que retornam informações estatísticas sobre domínios e computadores virtualizados. As informações estatísticas atualmente oferecem dados quantitativos que representam a configuração atual dos domínios e do monitor.

A fim de ilustrar a necessidade desta camada de abstração, a figura 3.4.1 apresenta a implementação de duas funções de XenClient.

As funções na figura 3.4.1 encapsulam o processo de criação de máquinas virtuais utilizando a API do monitor Xen. XenClient permite a criação de máquinas virtuais configuradas através de arquivos ou de um dicionários de dados. Quando a criação utiliza arquivos de configuração, primeiramente converte-se os dados do arquivo para o formato do dicionário especificado pela API. Posteriormente, realiza-se um teste que verifica a consistência dos parâmetros fornecidos.

Somente após a conferência dos parâmetros realiza-se a chamada ao procedimento remoto para criação da máquina virtual, passando-se o dicionário com os dados como argumento. Por fim, aguarda-se a inicialização do sistema operacional virtualizado, concluindo sua ativação.

```

def createDomainByFile(self, filesxp):
    if filesxp is None:
        return False

    line = ['create', filesxp]
    (self.opts, config) = parseCommandLine(line)
    return self.createDomainByDictionary(config)

def createDomainByDictionary(self, config):
    if dict is None:
        return False

    if not create_security_check(config):
        return False

    try:
        dominfo = self.server.xend.domain.create(config)
        dom = sxp.child_value(dominfo, 'name')
        self.server.xend.domain.waitForDevices(dom)
        if not self.opts.vals.paused:
            self.server.xend.domain.unpause(dom)
    except Exception, ex:
        print sys.stderr, ex
        return False

    return True

```

Figura 3.9: Exemplo de abstração oferecida por XenClient

3.4.2 Interface administrativa

De posse da API XenClient, desenvolveu-se uma interface gráfica administrativa. Neste ponto do processo de desenvolvimento, analisou-se as bibliotecas gráficas disponíveis para a linguagem Python e optou-se pela utilização de PyGTK (PyGTK, GTK+ FOR PYTHON, 2000).

Durante a análise das funcionalidades necessárias para uma ferramenta de gerenciamento, observou-se a necessidade de salvar as configurações ativas em um perfil de trabalho. Para isso, utilizou-se SQLite (SQLite, 2003), uma biblioteca que oferece características básicas de um sistema gerenciador de banco de dados, e implementa grande parte das instruções SQL relacionais. Já para realizar o acesso à base de dados, utilizou-se SQLAlchemy (BICKING, 2005), uma biblioteca que provê uma interface orientada a objetos para diferentes bases de dados. Desta forma, pode-se armazenar informações sobre os computadores que estão sendo monitorados e os usuários com permissão de acesso, garantindo a persistência do perfil de trabalho em uma rede.

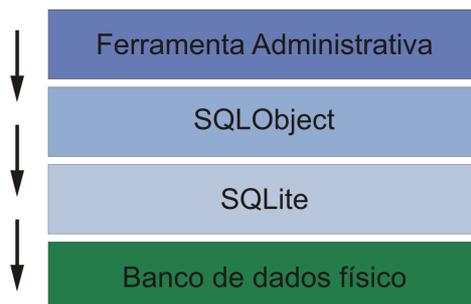


Figura 3.10: Tecnologias utilizadas para armazenar os dados

A figura 3.10 apresenta o posicionamento das bibliotecas utilizadas para armazenar os dados referentes a um perfil de trabalho. As setas representam o fluxo da realização das chamadas para recuperação dos dados. A interface gráfica solicita através de chamadas utilizando a biblioteca SQLAlchemy os dados armazenados e gerenciados por SQLite.

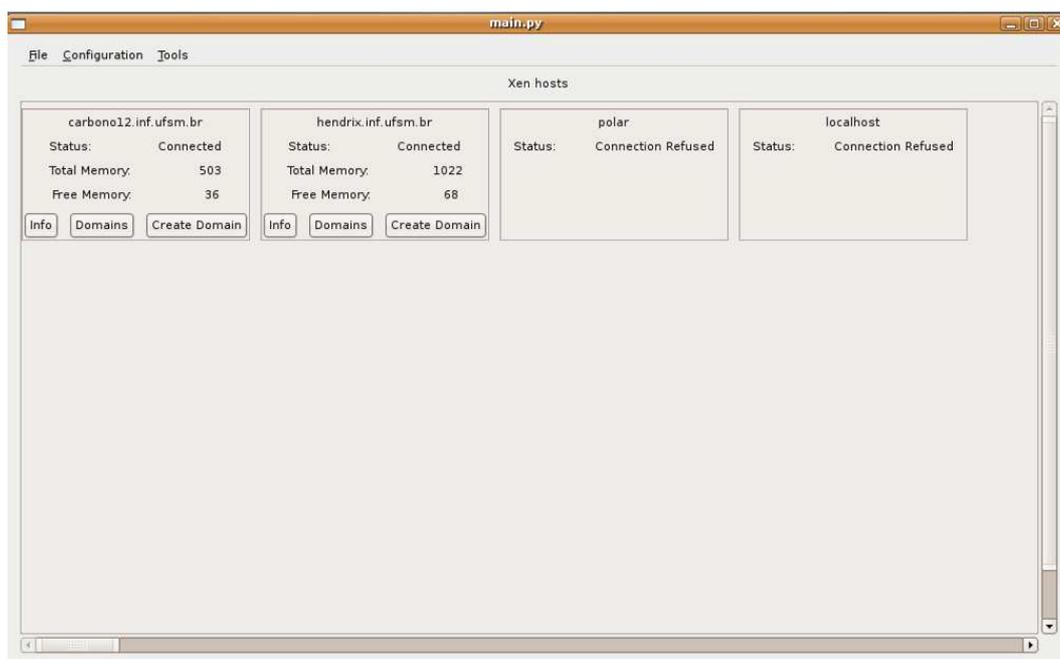


Figura 3.11: Interface principal da ferramenta

Já a figura 3.11 apresenta a interface gráfica de *AdXen*. Neste caso observa-se a existência de 4 *hosts* cadastrados e 2 *hosts* ativos. Para os *hosts* ativos são permitidos a execução de todas as funcionalidades da ferramenta, oferecendo acesso direto através da interface principal para criação de um novo domínio, visualização de informações do *host* e visualização das máquinas virtuais em execução. O menu de opções localizado no topo da interface permite acesso as demais funcionalidades oferecidas por *AdXen*.

A interface administrativa permite a visualização e acompanhamento dos *hosts* cadas-

trados, apresentando informações referentes à disponibilidade de conexão. Os computadores cadastrados podem estar ativos ou inativos. *Hosts* ativos podem ser usados em operações administrativas imediatamente, permitindo o acesso às informações de configuração e e realização de chamadas remotas. As informações são atualizadas temporariamente de acordo com uma opção de tempo informada no cadastro de preferências da ferramenta.

Para o sistema, cada computador cadastrado corresponde a uma instância de *XenClient*, que serve como classe base para a disponibilização das informações em tempo real. Desta forma, é possível identificar quando um computador altera seu estado de disponibilidade na rede, passando de ativo para inativo, ou vice-versa. Esta funcionalidade permite uma representação real dos recursos disponíveis a cada momento.

Durante a utilização de *AdXen*, a interface administrativa pode ser dividida em diversas janelas (formulários), agrupando informações de acordo com a sua utilização. Neste ponto, optou-se por utilizar apenas uma conexão com os *hosts* cadastrados, buscando não ocasionar uma sobrecarga nos computadores hospedeiros e no computador administrativo. Cada nova janela necessária utiliza o mesmo dicionário que armazena as referências de conexão.

3.4.3 Operações avançadas

Para facilitar o cadastro inicial e atualização dos computadores pertencentes a rede, desenvolveu-se uma funcionalidade denominada **Descobridor de hosts**. Basicamente, este recurso permite ao administrador identificar quais computadores possuem o MMV Xen ativo em uma rede. Para isso, foi implementado um conjunto de classes *python* independentes, utilizando recursos como *sockets* e *threads*.

No desenvolvimento da operação avançada **Gerenciamento de Grupos de Máquinas Virtuais**, realizou-se a representação de agrupamento de máquinas virtuais através de relacionamentos na base de dados. Desta forma, antes de inicializar um grupo o administrador deve informar quais máquinas virtuais fazem parte do mesmo. Este processo difere da criação básica de máquinas virtuais, onde o administrador informa somente o arquivo de configuração necessário.

Esta interação com o administrador ocorre através de formulários específicos, onde é possível cadastrar novos grupos e editar configurações existentes. Outros formulários

foram desenvolvidos para permitir a visualização de um grupo ativo, oferecendo um conjunto de operações básicas que podem ser efetuadas sobre todos os membros.

Para permitir este acompanhamento era necessário manter as informações do estado atual do ambiente monitorado. Assim, uma classe específica foi implementada para realizar um `log` administrativo de operações realizadas sobre máquinas virtuais participantes de grupos. Os métodos desta classe permitem um acompanhamento através do armazenamento das informações na base descrita anteriormente.

Basicamente, este acompanhamento permite manter as informações sobre grupos de máquinas virtuais mesmo quando a ferramenta é fechada e iniciada novamente. A forma de controle e identificação das máquinas virtuais na interface de Xen difere da identificação utilizada em *AdXen*, já que o MMV não disponibiliza um código único e identificador permanente para os domínios, dificultando sua identificação quando este já encontra-se em execução. Desta forma, o acompanhamento identifica e salva um histórico das operações realizadas, identificando em quais hospedeiros encontram-se os domínios participantes de um grupo de máquinas virtuais previamente cadastrado.

4 AVALIAÇÃO

Este capítulo apresenta uma avaliação das funcionalidades oferecidas pela ferramenta *AdXen*. São realizadas comparações entre operações administrativas efetuadas com e sem a utilização da ferramenta, destacando a forma de realização das operações avançadas.

4.1 Operações Básicas

Inicialmente, a comparação entre a realização de operações básicas com e sem a utilização de *AdXen*, permite observar as facilidades obtidas na manipulação das máquinas virtuais alocadas remotamente. Com este objetivo, descreve-se primeiramente a realização das atividades utilizando a interface nativa de Xen comparando realização do processo com a utilização da ferramenta. Posteriormente, apresenta-se alguns pontos que diferem na forma de realização destas atividades entre as ferramentas existentes.

4.1.1 Comparação com a interface nativa de Xen

Dentre as operações básicas descritas no capítulo 2 (seção 2.3), **criação, destruição, pausa/retomada, migração, desativação e obtenção de informações**, necessitam de um processo inicial de conexão, onde o administrador deve conectar-se ao computador remoto, utilizando um usuário que possua o mais alto nível de acesso.

Para efetuar a conexão, é necessário conhecer o IP ou o nome do *host* remoto. Após a conexão, algumas operações (**como criação e migração**) requerem que o administrador verifique manualmente a disponibilidade de recursos atuais local e remotamente. A etapa seguinte consiste na realização de uma chamada para o MMV Xen através da utilização do utilitário `xm`, passando os parâmetros adequados (particularmente a tarefa de **criação** requer a utilização de um arquivo para passagem de parâmetros específicos).

Após concluir a operação, qualquer alteração ou obtenção de informações administra-

tivas, como disponibilidade de recursos do *host* e das máquinas virtuais, necessita de uma nova conexão ao computador hospedeiro.

Utilizando *AdXen*, o administrador possui uma visão geral do ambiente monitorado diretamente na interface inicial (conforme a figura 3.11, apresentada no capítulo 3), possuindo acesso direto as tarefas de criação de máquinas virtuais e obtenção de informações sobre o *host* hospedeiro. Ainda nesta interface existe um botão *Domains* que permite a visualização dos domínios existentes no *host* selecionado.

Neste ponto, o administrador possui uma visualização dos domínios ativos, inclusive do domínio administrativo *Domain0*. As operações básicas são disponibilizadas individualmente para os domínios através da seleção de uma opção em uma lista. Desta forma, uma tarefa complexa como migração de uma máquina virtual entre computadores, é realizada facilmente, auxiliando o administrador do ambiente.

Ao término da operação, o administrador pode voltar a interface inicial e obter uma visualização atualizada do ambiente, já que os dados são atualizados seguindo um intervalo especificado pelo próprio administrador.

4.1.2 Comparação com ferramentas existentes

A comparação de operações básicas entre as ferramentas existentes difere basicamente na forma de apresentação da interface para o administrador e na disponibilidade ou não de algumas operações.

Sistemas como XenManager e Enomalism oferecem um conjunto reduzido de operações, compreendendo basicamente criação, destruição, ativação, desativação e migração. Particularmente a ferramenta Enomalism possui uma interface *web* que permite a manipulação das máquinas virtuais remotas utilizando autenticação unicamente através de LDAP. Desta forma, a instalação desta ferramenta exige a utilização deste mecanismo de autenticação, não utilizando o sistema disponível pela API de Xen, baseado em SSH.

Já XenManager permite o controle de máquinas virtuais apresentando uma interface *desktop* para o administrador, onde as informações são agrupadas através de abas de manipulação. Desta forma, não é possível ao administrador obter uma visão global do ambiente.

As ferramentas Virt-Manager e Xen Enterprise encontram-se em um estágio de desenvolvimento avançado e oferecem todas as operações básicas. Particularmente, Xen

Enterprise possui um conjunto de operações que permite a conversão de máquinas reais para máquinas virtuais, denominado *Physical to Virtual (P2V)*, mas é distribuído sob licença comercial, o que dificulta sua utilização em ambientes acadêmicos.

Desta forma, o desenvolvimento da ferramenta voltou-se para encontrar uma interface onde as operações básicas pudessem ser realizadas pelo administrador de uma forma simples e rápida, permitindo ainda o acompanhamento e visualização global do sistema. Assim, a representação utilizada permite a realização de qualquer tarefa administrativa através da seleção de opções em listas e/ou através da utilização de barras de botões.

4.2 Operações Avançadas

Este ponto da avaliação descreve a realização de operações avançadas disponibilizadas por *AdXen*, caracterizando o principal diferencial perante outras soluções existentes. As operações avançadas, descoberta de *hosts* com o MMV Xen em execução e gerenciamento de grupos de máquinas virtuais, auxiliam o administrador na implantação do sistema no seu ambiente de trabalho e no controle do conjunto de domínios em execução.

O administrador do sistema pode optar por realizar o cadastro manual dos *hosts* existentes na rede ou utilizar o recurso denominado *Xen Discover*, que percorre todos os computadores identificando quais possuem o MMV Xen em execução. Este processo é realizado em todas as interfaces de rede existentes do computador utilizado para administrar o ambiente. Uma lista com os *hostnames* encontrados é apresentada para o administrador, que pode efetuar ou não o seu cadastro.

Em ambientes que possuem um alto número de computadores utilizando o MMV Xen a utilização deste recurso torna-se fundamental para a configuração inicial do sistema, evitando assim a necessidade de digitação e cadastro manual dos *hosts*. Por outro lado, o administrador pode efetuar a descoberta de *hosts* a qualquer momento, buscando identificar alterações em sua rede, já que somente serão cadastrados os novos computadores identificados.

Gerenciar grupos de máquinas virtuais consiste basicamente em agrupar domínios que possuem particularidades em comum. Desta forma, o administrador pode identificar domínios que executem programas inter-relacionados e agrupá-los através de uma identificação única. Inicialmente deve-se cadastrar um grupo de máquinas virtuais informando seu nome e uma breve descrição sobre as aplicações executadas e posteriormente informar

quais arquivos de configuração serão utilizados para inicializar o grupo.

No momento da ativação de um grupo, os *hosts* ativos são apresentados para o administrador, que pode selecionar quais deles receberão as novas máquinas virtuais. A distribuição dos domínios sobre os *hosts* selecionados é feita de forma igualitária, dividindo o número de domínios pelo número de hospedeiros selecionados.

Grupos ativos podem ser visualizados separadamente dos demais *hosts* do ambiente, permitindo assim uma visão específica dos computadores hospedeiros e suas máquinas virtuais. Neste ponto, o administrador possui acesso direto a todas as operações básicas, podendo executar qualquer tarefa sobre um domínio específico. Considerando a necessidade de reconfiguração do ambiente de trabalho, o administrador possui operações que podem ser aplicadas sobre todas as máquinas virtuais pertencentes ao grupo em questão, como desativação e destruição.

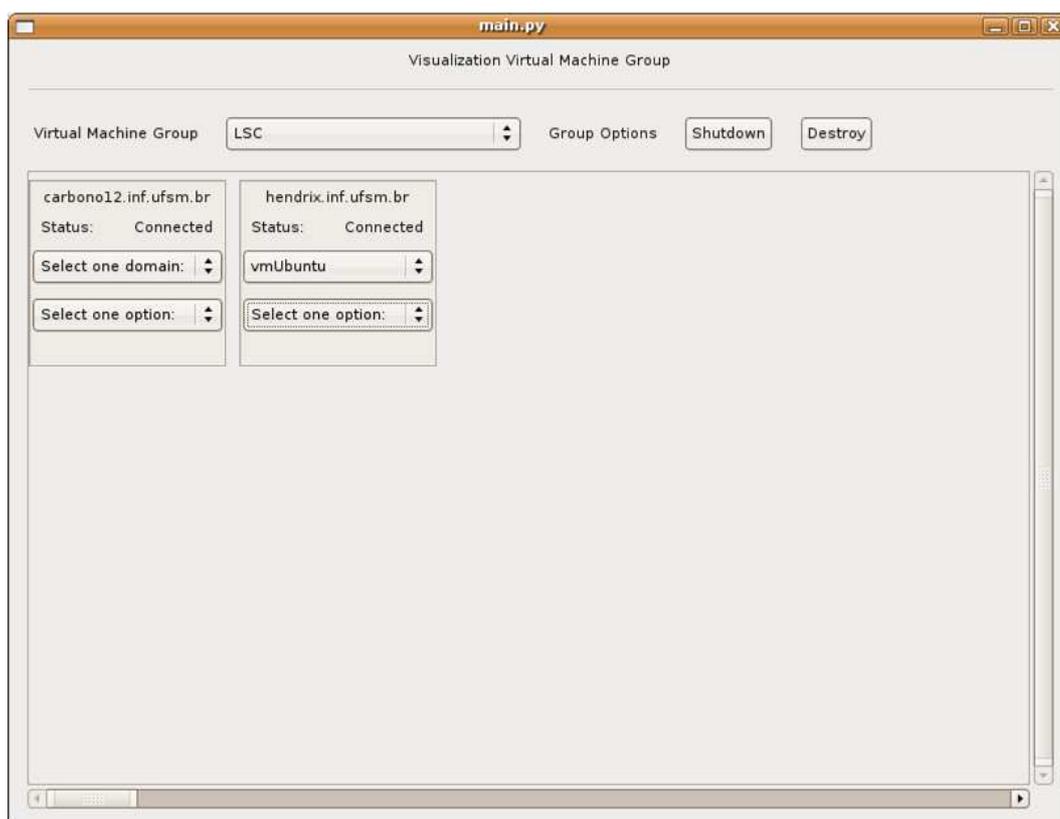


Figura 4.1: Visualização de um grupo de máquinas virtuais

A figura 4.1 apresenta a interface utilizada para visualizar um grupo de máquinas virtuais. Após a seleção do grupo para visualização, ocorre um agrupamento inicial dos domínios alocados sobre um mesmo *host*. Internamente em cada *host*, é possível acessar os domínios e as operações através da seleção de opções em listas. Na parte superior da

interface, botões permitem ao administrador efetuar as operações sobre o grupo selecionado, propagando a opção para todas as máquinas virtuais existentes.

5 CONCLUSÃO

Através da virtualização de recursos computacionais é possível obter um melhor aproveitamento do *hardware* e *software* existentes em uma máquina real. Atualmente este conceito surge como uma alternativa para otimização do poder de processamento de computadores modernos. Monitores de máquinas virtuais permitem a execução simultânea e concorrente de diversos sistemas operacionais sobre uma mesma arquitetura real, compartilhando o *hardware* de forma segura e transparente. Particularmente, o monitor de máquinas virtuais Xen oferece o recurso de migração de uma máquina virtual entre computadores interconectados, permitindo sua utilização para fins como balanceamento carga e administração de recursos de redes.

Arquiteturas virtualizadas requerem uma manipulação das máquinas virtuais e sistemas operacionais que serão alocados sobre o MMV. Tarefas de gerenciamento de máquinas virtuais, sejam tarefas básicas ou tarefas avançadas, tornam-se trabalhosas devido a inexistência por parte de Xen de uma interface amigável para este fim.

Este trabalho descreveu o processo de implementação de *AdXen*, uma ferramenta para administração de arquiteturas virtualizadas baseadas em Xen, descrevendo seu ambiente de execução e apresentando uma avaliação de sua utilização. O processo de desenvolvimento desta ferramenta guiou-se pelas especificações da API de gerenciamento de Xen, oferecendo para o usuário uma interface para administração de *hosts* virtualizados local e remotamente.

Durante o processo de desenvolvimento foram descritas as operações básicas e avançadas necessárias para administrar um ambiente virtualizado. O processo de avaliação comparou a utilização de *AdXen* com a interface nativa de Xen e com outras ferramentas, apontando as facilidades obtidas pelo administrador na execução de operações básicas.

Quanto as operações avançadas, apresentou-se uma descrição de suas funcionalidades,

apresentando através de um estudo de caso os pontos onde esta pode auxiliar o administrador do ambiente.

Além do objetivo principal do trabalho, foi desenvolvida uma API denominada *XenClient*, que abstrai as chamadas realizadas pela interface administrativa para o MMV Xen. Desta forma, a interface torna-se independente da API do MMV Xen, permitindo assim sua integração futura com outras tecnologias de virtualização, como por exemplo a biblioteca *LibVirt* (LibVirt - VIRTUALIZATION API, 2006).

O processo de desenvolvimento da ferramenta proposta permitiu a percepção de outros pontos que podem ser abordados para aumentar sua funcionalidade. Dentre eles a instalação e preparação de *kernels* e imagens de sistemas operacionais, requer uma atenção especial, pois caracteriza-se por ser uma tarefa trabalhosa e complexa. Projetos como *Xen-Get* (Xen-Get, 2006) oferecem uma interface que permite a instalação de sistemas operacionais previamente configurados, diminuindo a complexidade desta tarefa. Desta forma, a adaptação destas funcionalidades na ferramenta desenvolvida aumentará seu escopo de atuação.

Considerando ainda o gerenciamento de imagens de sistemas operacionais, outro ponto que pode ser abordado é a possibilidade de *clonagem* de uma máquina virtual. Esta funcionalidade auxiliará na administração de ambientes onde deseja-se salvar o estado atual de uma máquina virtual, considerando as diversas alterações realizadas desde sua criação. Este processo de clonagem pode ser estendido para atender ambientes onde deseja-se manter réplicas semelhantes em execução. Esta duplicação permitiria a criação de uma nova máquina virtual considerando como estado inicial o ponto onde efetuou-se a clonagem.

Outra possibilidade seria a alteração da API *XenClient* para a realização de chamadas para a biblioteca *LibVirt* (LibVirt - VIRTUALIZATION API, 2006). Desta forma, a ferramenta administrativa tornaria-se independente dos monitores de máquinas virtuais utilizados no ambiente, já que o desenvolvimento de *LibVirt* baseia-se nessa funcionalidade.

REFERÊNCIAS

BARHAM, P.; DRAGOVIC, B.; FRASER, K.; HAND, S.; HARRIS, T.; HO, A.; NEUGEBAUER, R.; PRATT, I.; WARFIELD, A. Xen and the art of virtualization. In: ACM SYMPOSIUM ON OPERATING SYSTEMS PRINCIPLES (SOSP '03), 19., 2003, Bolton Landing, USA. **Proceedings...** ACM, 2003. p.164–177.

BICKING, I. **SQLObject - Object Relational Manager**. Disponível em: <http://www.sqlobject.org>.

BIRRELL, A. D.; NELSON, B. J. Implementing Remote Procedure Calls. **ACM TOCS**, [S.l.], v.2, n.1, p.39–59, Feb. 1984.

COM., U. **XML Remote Procedure Call Specification**. Disponível em: <http://www.xmlrpc.com/>.

GOLDBERG, R. Survey of virtual machine research. **IEEE Computer**, [S.l.], v.7, n.6, p.34–45, 1974.

HAZARD, H.; JEDI, J. **XenMan - Xen management tool**. Disponível em: <https://sourceforge.net/projects/xenman/>.

HERGERT, C. **Xen Manager**. Disponível em: <http://xenmanager.dronestudios.com/cgi-bin/trac.cgi>.

HUANG, W.; LIU, J.; ABALI, B.; PANDA, D. A Case for High Performance Computing with Virtual Machines. **The 20th ACM International Conference on Supercomputing**, [S.l.], 2006.

INC, E. **Enomalism - Virtualized Management Console**. Disponível em: <http://www.enomalism.com>.

INC., X. S. **Xen Enterprise Data Sheet V11022006, version 3.1.** [S.l.]: Xen Source Inc., 2006. (02-11-2006).

LibVirt - Virtualization API. Disponível em: <http://libvirt.org/index.html>.

MELLOR, E.; SHARP, R.; SCOTT, D.; HARROP, J. **Xen Management API Draft - Revision 0.4.3.** [S.l.]: XenSource, Inc., 2006. (25-08-06).

NELSON, B. **Remote Procedure Call.** 1981. Tese (Doutorado) — Carnegie-Mellon U. published as CMU technical report CMU-CS-81-119.

PLUMMER, D. C. **Ethernet Address Resolution Protocol: or converting network protocol addresses to 48 bit ethernet address for transmission on ethernet hardware.** Disponível em: <http://www.ietf.org/rfc/rfc826.txt>, RFC 826 (Standard).

PyGTK, GTK+ for Python. Disponível em: <http://www.pygtk.org>.

RITCHIE, D. M.; KERNIGHAN, B. W.; LESK, M. E. **The C Programming Language.** Murray Hill, New Jersey: Bell Laboratories, 1975. Comp. Sci. Tech. Rep. No., 31 Superseded by B. W. Kernighan and D. M. Ritchie, *The C Programming Language*, Prentice-Hall, Englewood Cliffs, N.J., 1988. (31).

SMITH, J. E.; NAIR, R. The Architecture of Virtual Machines. **IEEE Computer**, [S.l.], v.38, n.5, p.32–38, 2005. Disponível em: <http://doi.ieeecomputersociety.org/10.1109/MC.2005.173>.

SQLite. Disponível em: <http://www.sqlite.org>.

STAHL, E. **Virtualization Security and Integrity in the IBM @server POWER5 Environment.** [S.l.]: IBM Systems and Technology Group, 2005. (090105).

SUGERMAN, J.; VENKITACHALAM, G.; LIM, B.-H. Virtualizing I/O Devices on VMware Workstation's Hosted Virtual Machine Monitor. In: **USENIX ANNUAL TECHNICAL CONFERENCE, 2001.**, 2001. **Proceedings...** Usenix Assoc., 2001. p.1–14.

The Python Language. Disponível em: <http://www.python.org>.

VIRT-MANAGER Virtual Machine Manager. Disponível em: <http://virt-manager.et.redhat.com/>.

WHITAKER, A.; SHAW, M.; GRIBBLE, S. D. **Denali**: lightweight virtual machines for distributed and networked applications. [S.l.]: University of Washington, 2002. (02-02-01).

Xen-Get. Disponível em: <http://www.xen-get.org/>.

XEN-TOOLS.ORG. **Argo - Xen Control Panel**. Disponível em: <http://xen-tools.org/software/argo/>.

APÊNDICE A - PROCESSO DE INSTALAÇÃO

Este apêndice descreve o processo necessário para instalação da ferramenta *AdXen*, apresentando os pré-requisitos para seu funcionamento e alguns erros comuns.

Pré-requisitos

- Sistema Operacional Linux
- Bibliotecas Python fornecidas pelo MMV Xen (API do MMV Xen)
- Permissão para execução com direitos de super-usuário
- Interpretador/Compilador Python com versão mínima 2.4
- Bibliotecas SQLAlchemy com suporte para linguagem Python
- Bibliotecas SQLite com suporte para linguagem Python
- Ambiente seguro utilizando conexão através de SSH

Instalação

O processo de instalação é composto pelas seguintes etapas:

1. Download de *AdXen* e seus arquivos fontes no endereço:

<http://www.inf.ufsm.br/~guilherm/tg/AdXen.tar.gz>

2. Descompacte os arquivos dentro de uma pasta não temporária (ex: `$ tar -xvzf AdXen.tar.gz`)
3. Execute o arquivo *createDataBase.py* (ex: `$ python createDataBase.py`) para criar a base de dados necessária para utilizar o sistema. Observação: caso ocorra algum

erro neste processo, revise a instalação das bibliotecas SQLAlchemy e SQLite de acordo com a distribuição Linux utilizada

4. Execute a ferramenta com permissões de super-usuário (ex: `$ sudo python main.py`)

Erros comuns

Alguns erros comuns que podem ocorrer no processo de instalação e suas prováveis soluções são especificados abaixo.

- Erro ao criar a base de dados: Deve-se principalmente a utilização de versões incorretas das bibliotecas SQLAlchemy e/ou SQLite. Recomenda-se a utilização das últimas versões disponíveis em <http://www.sqlalchemy.org> e <http://www.sqlite.org> respectivamente. Algumas distribuições possuem um mecanismo auxiliar de instalação que pode facilitar o processo.
- Erro ao efetuar conexão utilizando SSH: O usuário que está executando a ferramenta deve possuir direitos de super-usuário. De acordo com as especificações da API do MMV Xen, somente super-usuários podem manipular as máquinas virtuais.
- Erro na conexão aos computadores hospedeiros: De acordo com o guia do usuário do MMV Xen, deve-se configurar o arquivo `xend-config.sxp` habilitando a conexão remota. Maiores informações podem ser obtidas em <http://www.cl.cam.ac.uk/research/srg/netos/xen/documentation.html>.

APÊNDICE B - EXEMPLO DE UTILIZAÇÃO

O processo básico de utilização de *AdXen* consiste no cadastro dos *hosts* com o MMV Xen em execução, criação de grupos de máquinas virtuais e utilização das operações administrativas básicas, sejam sobre um único domínio ou sobre um conjunto de domínios.

Todos os formulários utilizam por padrão uma barra de botões que apresenta as seguintes opções (de acordo com a necessidade): **Ok**, **Delete** e **Cancel**, responsáveis pelas seguintes tarefas, respectivamente: confirmação da operação efetuado, exclusão de registro selecionado em uma lista e fechamento da janela ativa.

Cadastro dos computadores hospedeiros

O cadastro dos *hosts* pode ser efetuado manualmente ou utilizando a ferramenta *Xen Discover*.

Para efetuar o cadastro manual dos *hosts* acesse a opção **Configuration / Hosts** e informe o *hostname* seguido pelo nome de usuário para conexão.

A ferramenta *Xen Discover* pode ser acessada em **Tools / Discover Xen Hosts**, onde o usuário deve informar a porta utilizada para conexão com o MMV Xen caso seja diferente da porta padrão sugerida. Ao clicar em *Scan*, a ferramenta inicia o processo de busca, apresentando os *hosts* encontrados em uma lista, aguardando a confirmação do cadastro.

Grupos de máquinas virtuais

Inicialmente, cadastre os grupos e suas descrições utilizando o formulário disponível em **Configuration / VMs Groups**. Após este cadastro, selecione na lista o grupo desejado e clique sobre o botão *Members*.

O formulário apresentando neste ponto permite o cadastro dos arquivos de configurações das máquinas virtuais informando uma breve descrição sobre seu conteúdo (opcio-

nal).

Para ativar um grupo de máquinas virtuais, acesse **Tools / Start VM Group**. Inicialmente selecione o grupo que será ativado e posteriormente selecione quais dos *hosts* apresentados na lista no canto esquerdo serão utilizados. Os botões *Add* e *Sub* selecionam e removem *hosts* da lista de seleções, localizada no canto direito do formulário.

Grupos de máquinas virtuais já ativos podem ser visualizados separadamente através do formulário **Tools / Visualization VM Group**. Selecione na lista qual dos grupos será visualizado e automaticamente os *hosts* ativos e seus respectivos domínios serão apresentados. Neste ponto é possível efetuar operações sobre um domínio específico, através de sua seleção nas listas. Estão disponíveis também os botões *Shutdown* e *Destroy* que permitem a desativação e destruição de todos os domínios pertencentes ao grupo selecionado.

Execução de operações básicas

Em diversos formulários de *AdXen*, o administrador pode efetuar as operações básicas através da seleção da opção desejada em uma lista de operações. Observe que o domínio *Domain0* permite somente a obtenção de informações, pois trata-se do domínio administrativo do MMV Xen.