Gestão de Redes com Base em Políticas - Uma Abordagem Prática

Rui Pedro Lopes	José Luís Oliveira	Vitor Roque
rlopes@ipb.pt	jlo@det.ua.pt	vitor.roque@ipg.pt
ESTiG - IPB	DET - UA	ESTG - IPG
5301-854 Bragança	3810-193 Aveiro	6301-559 Guarda

Resumo

A gestão de redes é um processo essencial para manter uma rede de comunicação em bom estado de funcionamento. Todos os dias surgem novos dispositivos, novas formas de acesso e novos protocolos que permitem o acesso mais rápido e cómodo à informação. Como resultado, as redes crescem exponencialmente e diversificam-se em termos tecnológicos – a gestão torna-se mais complexa e sensível a erros.

Os modelos de gestão clássica, como o SNMP, CLI ou outros, assentam o seu funcionamento na monitorização e configuração individual de recursos. Mais recentemente, a introdução de técnicas e mecanismos de gestão de rede com base em políticas promete mudar a perspectiva para a globalidade da rede, dando ao utilizador uma visão e um controlo global. Além disso, permitem também associar as tarefas de monitorização e controlo aos objectivos e acordos de negócio, até então incompatíveis.

Palavras–Chave: PBNM, Gestão de Redes, Políticas.

1 Introdução

As redes de comunicação estão a tornar-se universais, em termos de localização, terminais e necessidades pessoais na nossa sociedade. A sua crescente importância é acompanhada pela necessidade de um sistema de gestão robusto, flexível e útil, que deve ajudar a manter a rede em bom estado de funcionamento.

No princípio dos anos noventa, o IETF procurou responder a esta necessidade com o modelo SNMP (*Simple Network Management Protocol*) [1], prevendo-se a sua substituição a curto prazo por uma arquitectura mais poderosa e flexível – o CMIP (*Common Management Information Protocol*) [2, 3]. Esta substituição nunca veio a suceder e, actualmente, ainda se usa o protocolo SNMP, neste momento na sua terceira geração. No entanto, o modelo SNMP sofre de vários inconvenientes, em particular relacionados com operações de configuração [4].

Como resultado, nota-se uma separação notável entre as operações de monitorização e de configuração. A primeira é tipicamente efectuada por SNMP enquanto que a segunda se baseia em intervenção directa no equipamento de rede, por CLI (*Command Line Interface*) ou ferramentas baseadas em tecnologia Web. Esta abordagem, como é óbvio, faz com que as tarefas de configuração

de equipamento de rede sejam repetitivas, sensíveis a erros de operador e, por vezes, cansativas. O problema é agravado com a crescente dimensão da rede.

Mais recentemente, temos vindo a assistir ao aparecimento de propostas e normas para a gestão de redes com base em políticas. Neste contexto, as políticas representam uma forma de agregar opções de configuração, no que é vulgarmente conhecido como *Policy-based Network Management* (PBNM) [5, 6].

Relativamente a PBNM e apesar de várias propostas se encontrarem actualmente disponíveis, resta saber até que ponto é que é possível e prático utilizar este tipo de tecnologia nas redes actuais. Váris questões se levantam, nomeadamente, como representar as políticas, como armazenar a informação do sistema, como traduzir as políticas de alto-nível para os comandos elementares de configuração, entre outras.

Este artigo pretende abordar estas questões com algum pormenor, apresentando um exemplo concreto de gestão com base em políticas, desde a sua especificação, armazenamento e algumas possíveis traduções para comandos de configuração.

A próxima secção apresenta alguns detalhes sobre PBNM. Segue-se um exemplo de aplicação, com a especificação da política todos os passos necessários para a sua aplicação prática. O artigo termina com algumas conclusões.

2 Gestão com Base em Políticas

No seu sentido mais lato, políticas são planos de uma organização para alcançar objectivos. Por outras palavras, uma política é uma especificação de objectivos ou de um conjunto de acções a desempenhar no futuro ou como resultado de uma actividade regular [5].

No contexto da gestão de redes, uma política é a relação entre os objectos de rede, tais como grupos de dispositivos de rede, os recursos, serviços e utilizadores. Por exemplo, uma política de gestão de largura de banda pode ser aplicada a todos os *routers* numa região particular ou de um certo tipo.

Resumindo, as políticas podem ser consideradas como regras que descrevem acções a serem tomadas quando condiçoes específicas acontecem:

$$\text{if } (\textit{policyCondition}) \text{ then } (\textit{policyAction}) \tag{1}$$

É possível que a aplicação de políticas dependa de acontecimentos, sob a forma de eventos. Neste caso, as regras poderão incluir uma parte associada [7]:

$$(\textit{policyEvent}) \text{ causes } (\textit{policyAction}) \text{ if } (\textit{policyCondition}) \tag{2}$$

A gestão com base em políticas (PBNM) apresenta cinco grandes vantagens:

- simplifica a gestão de dispositivos, redes e serviços,
- apresenta uma redução no esforço de configuração,
- define o comportamento da rede como um todo,
- escala melhor em termos de complexidade da rede,
- utiliza a visão e procedimentos de negócio como base para a configuração da rede.

No entanto, a PBNM aumenta, consideravelmente, a complexidade do sistema de gestão, o que pode agravar a sua aceitação no seio da organização.

2.1 Modelo de políticas

Do ponto de vista do utilizador, as políticas introduzem uma mudança de nível de abstracção, passando da configuração de dispositivos para a definição de operações de gestão em termos de metas ou objectivos, em oposição às sequências de *gets* ou *sets*.

Para que estas sejam interpretadas pelo sistema é necessário traduzir estas regras para um modelo de dados de mais baixo nível. Este passo segue o denominado *policy continuum* [8], onde vários níveis de abstracção definem como a política é representada desde o nível de negócio até ao de dispositivo (Figura 1).

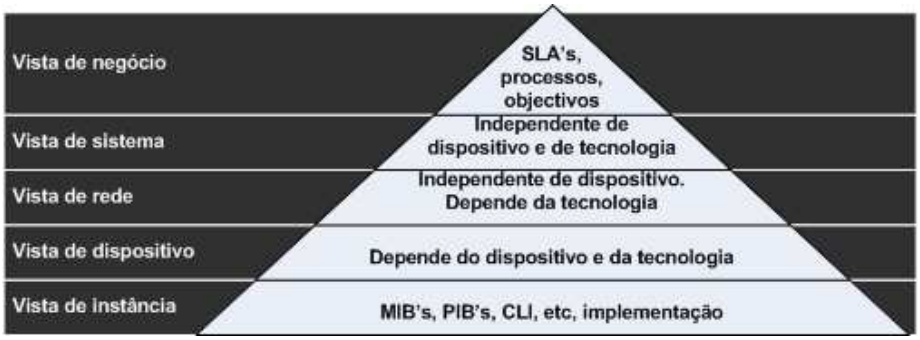

Figura 1: Policy continuum.

No seu nível mais elevado, a política tem a forma de uma regra que pode ser descrita em linguagem humana, do tipo: "o utilizador xpto pode transferir ficheiros". No nível de Sistema, a mesma política poderá ter de ser representada numa outra forma, ainda independente do dispositivo e da tecnologia (este formato foi adaptado de QPIM – `http://www.ietf.org/proceedings/00dec/slides/POLICY-3/index.html`):

```
Group A: Role=[transferênciaFicheiros] {
  if(user='xpto')
    autorizar transferência de ficheiros
}
```

O processo continua até ao nível mais baixo, onde a política é traduzida para informação num formato que depende da implementação do sistema. Poderá, por exemplo, ser traduzida numa sequência de comandos CLI para configurar *routers* ou comandos SNMP para alterar o estado de determinado dispositivo de rede.

Neste exemplo são introduzidos os conceitos de *Group* e de *Role*. De acordo com o IETF, as políticas podem ser agrupadas sem restrição para o número de níveis (grupos dentro de grupos dentro de grupos). Como exemplo, um grupo de políticas que se aplicam a "Alunos" pode ser constituído por outros grupos de "Alunos 1º Ano", "Alunos 2º Ano", ..., "Alunos 5º Ano".

O conceito de *Role* é utilizado para a selecção de políticas. Assim, é atribuído um *role* a cada recurso e depois utilizado como selector, do tipo: o *router* X é do tipo *access router*; aplicar a política Y a todos os *access router*.

2.2 Definições

Os sistemas tradicionais de gestão encontram-se fortemente associados aos procedimentos de instrumentação, tipicamente de baixo-nível. Contudo, as organizações de normalização encontram-se activamente a trabalhar no sentido de definir regras de gestão de mais alto-nível. Um exemplo é o COPS (*Common Open Policy Service*) [9] e *SNMP for Configuration* [10], no seio do IETF, e CIM [11] e PCIM [12], no âmbito do DMTF (na realidade, PCIM é o resultado de ambas as organizações).

De acordo com o IETF, a arquitectura de PBNM é constituída por quatro entidades (Figura 2): a *Policy Management Tool*, também conhecida por *Policy Console*, o *Policy Repository*, o *Policy Decision Point* (PDP) ou *Policy Server* e o *Policy Enforcement Point* (PEP). Tal como no caso do SNMP, o IETF,

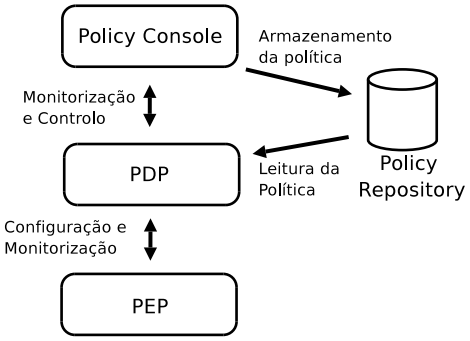

Figura 2: Arquitectura PBNM.

apesar de descrever os componentes chave, não tece qualquer consideração relativamente a detalhes de implementação. Como consequência, a *Policy Console* encontra-se relativamente esquecida e depende fortemente das opções adoptadas pelo programador.

O *Policy Repository* tem o papel de armazém de políticas. A informação aqui armazenada descreve utilizadores, aplicações, computadores, serviços e correspondentes associações.

O PDP é a entidade responsável por avaliar em que instante e de que forma as políticas podem ser aplicadas. O PEP é o ponto onde as decisões são aplicadas.

Relativamente à aplicação de políticas, foram definidos dois modelos: o *outsourcing* e o *provisioning* (também conhecido como configuração). No primeiro, o PEP envia pedidos de decisão (REQ), que podem resultar de eventos gerados na rede, ao PDP e aguarda que as decisões (DEC) sejam tomadas. O PDP toma a decisão e envia a mensagem correspondente (por exemplo, aceitar ou rejeitar) para o PEP. O modelo *outsourcing* é baseado no PEP e envolve um relacionamento de 1:1 entre os eventos do PEP e as decisões do PDP (Figura 3).

No modelo *provisioning*, o PDP gera decisões e depois informa o PEP relativamente às configurações que tem de aplicar. O PEP pode enviar notificações para o PDP no caso de acontecerem mudanças de estado. Não se assume um relacionamento de 1:1 entre os eventos gerados pelo PEP e as decisões do PDP.

A transferência de informação entre PDPs e PEPs é efectuada por intermédio de um Protocolo de Políticas (*Policy Protocol*). Uma das opções actualmente disponíveis é o COPS [9], especificado com o objectivo de transferência de

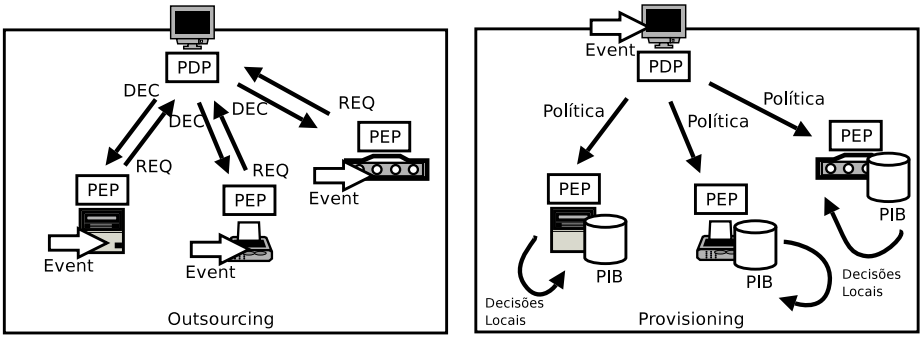

Figura 3: Modelos de distribuição de políticas.

políticas de âmbito intra-organizacional. Por outro lado, os PDPs envolvidos no processo de decisão podem estar localizados em diferentes organizações ou diferentes domínios (*Management Domains*), pelo que podemos distinguir a transferência intra-organizacional e inter-organizacional. No último caso, podem ser necessários acordos de serviço (SLA – *Service Level Agreements*) para que haja um entendimento comum.

2.3 Topologia

A arquitectura conceptual sugere que os PEPs estejam instalados nos dispositivos de rede, à semelhança do que acontece com os agentes SNMP (Figura 4).

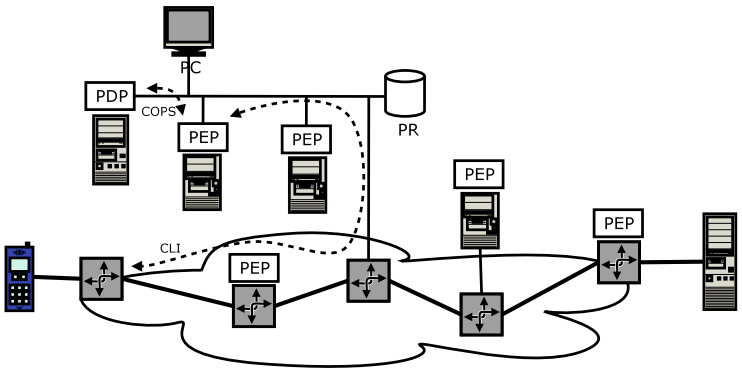

Figura 4: Topologia na prática.

Desta forma, eles usufruem das vantagens associadas à proximidade com o dispositivo que têm à sua responsabilidade. Na prática, esta possibilidade é, de momento, improvável. Há poucos *routers* e outros dispositivos que sejam capazes de albergar um PEP, pelo que esta terá de ser disponibilizada externamente. Neste caso, a funcionalidade de PEP será instalada em computadores que, por um lado, dialogam com o PDP e, por outro lado, com o dispositivo de rede. Esta abordagem requer que o administrador mantenha servidores dedicados para armazenar e executar PEPs. Geralmente, os dispositivos de rede possuem uma

aplicação ou "agente" responsável pela monitorização e controlo do seu funcionamento. Os *routers* Cisco, por exemplo, são geralmente configurados por CLI enquanto que outros dispositivos admitem configuração por SNMP.

O PEP deve ter o papel de converter os dados recebidos do PDP em parâmetros de configuração específicos, estabelecendo assim uma camada de abstracção sobre os parâmetros de configuração específica [13]. Esta abordagem permite uniformizar, ao nível do PDP, a especificação de políticas, recaíndo sobre o PEP a tarefa de as converter para a linguagem, protocolo e modelo de dados específicos (Figura 5).

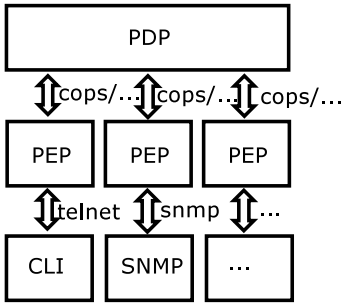

Figura 5: Camada de adaptação.

3 Caso de Aplicação Prática

Esta secção descreve um caso de aplicação prática de gestão de redes com base em políticas. Em PBNM, é praticamente impossível prever todas as regras e combinações possíveis, pelo que é impensável pretender implementar toda a estrutura logo à partida. Uma abordagem possível será considerar pequenos cenários e implementar cada um deles de forma independente, tendo em mente a futura expansão do sistema.

O caso de aplicação prática apresentado de seguida tem como cenário uma sala de alunos da Escola Superior de Tecnologia e de Gestão (ESTiG) do Instituto Politécnico de Bragança (IPB). A sala em causa, daqui para a frente identificada por "sala 103", encontra-se equipada com 45 estações de trabalho baseadas em arquitectura Intel e com os sistemas Windows 2000 e Debian GNU/Linux em *dual-boot*. A rede é 10/100BASE-T com ligações em *full-duplex* a um *switch* com capacidade de gestão SNMP (Figura 6).

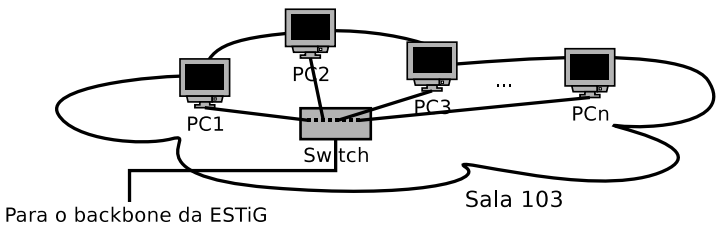

Figura 6: Topologia da sala 103.

Esta sala é de acesso livre para os alunos poderem realizar trabalhos, consultar a Internet ou outras tarefas extra-curriculares. Além disso, é também utilizada para a realização de exames práticos nas disciplinas de programação.

A sala tem um sistema PXE (*Preboot Execution Environment*) [14] que é utilizado para fazer o restauro dos ficheiros de sistema sempre que for necessário.

No estudo seguinte optámos por usar o modelo de PBNM definido pelo IETF em conjunto com a DMTF – PCIM [12]. Apesar de ainda não ser muito utilizado, estão já disponíveis algumas ferramentas de desenvolvimento, APIs e aplicações baseadas neste modelo. Além disso, encontram-se já especificados esquemas para LDAP [15], o que torna mais simples a utilização de um serviço de directoria como repositório de informação.

3.1 Conceptualização do sistema de gestão

A abordagem seguida para aplicar um modelo de PBNM à gestão desta sala é do tipo *top-down*, onde é(são) especificada(s), em primeiro lugar, a(s) política(s) a seguir a nível de negócio e, de seguida gradualmente detalhada até chegar ao nível mais baixo, ou seja, de configuração real do equipamento ou serviços.

A política de nível de negócio é, tipicamente, definida pelo Conselho Directivo ou pelo Coordenador dos Centro de Informática. Neste caso, vamos considerar a seguinte política: "Os alunos não podem comunicar nem aceder à Internet durante a realização de um exame."

Esta regra, depois de recebida pelo Centro de Informática, irá ser processada de forma a poder ser armazenada no sistema de gestão de políticas. Em primeiro lugar, será dado um formato de Sistema, ou seja, independente do dispositivo e da tecnologia (Figura 1):

```
Group Alunos: Role=[exameInformática] {
  if(exame_a_decorrer())
    desligar comunicação entre estações de trabalho
}
```

Ao descrever a regra desta forma, há várias questões que se levantam:

Como verificar as datas em que decorrem exames na sala 103? Esta condição é verdadeira no caso de a hora actual corresponder a um período de tempo em que está prevista a realização de um exame naquela sala. Por outras palavras, esta condição é verdadeira entre o início e o fim do exame.

Como fazer para interromper a comunicação entre as estações de trabalho? Dado que a rede é baseada em tecnologia Ethernet, em que o equipamento de ligação é um *switch*, é possível actuar de diversas formas sendo, talvez, a mais simples, desligar as portas em causa.

Como verificar se a regra foi correctamente aplicada? Após a política ser aplicada, é necessário verificar se a configuração teve sucesso. Neste caso, poder-se-á fazer um teste a cada porta do *switch* para verificar a sua configuração.

O que fazer no caso de a comunicação não ter sido interrompida? No caso de algo falhar, o administrador deverá ser notificado que algo não correu bem.

Estas questões fazem-nos aperceber que o sistema tem de estar ciente da topologia da rede, nomeadamente, das estações de trabalho e do equipamento de rede que se encontra instalado na sala 103. Além disso, é necessário modelar

a "forma como fazer". De notar que há várias abordagens para desligar uma porta de um *switch*, dependendo do fabricante e do modelo do dispositivo. É comum encontrar ferramentas de configuração que recorrem ao SNMP, CLI ou Web, pelo que é importante isolar estes detalhes do sistema de gestão. Assim, quando for enviada uma ordem para desligar determinada porta, o sistema saiba como agir.

De acordo com o modelo PCIM, o sistema é representado por um conjunto de classes que definem todos os intervenientes no processo de gestão com base em políticas. PCIM é uma extensão, ou seja, uma especialização, do CIM, pelo que podemos associar a hierarquia de classes ao *Policy Continuum* (Figura 7).

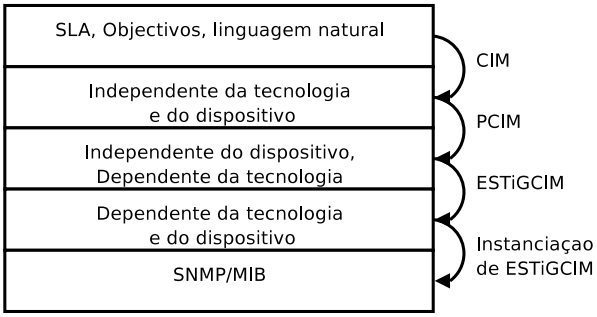

Figura 7: Extensão do modelo PCIM.

À medida que vamos especializando o modelo de dados vamos definindo mais detalhes que serão necessários para definir o procedimento de configuração. A estrutura ESTiGCIM será detalhada mais à frente.

3.2 Modelação OO

Relativamente ao processo de modelação Orientada ao Objecto, há vários conceitos que têm de ser modelados. PCIM permite modelar políticas usando condições (`PolicyCondition`), acções (`PolicyAction`), *roles* (`PolicyRoleCollection`) e grupos (`PolicyGroup`), entre outros. Nas figuras seguintes não é representada toda a hierarquia definida pelo DMTF no âmbito do CIM. São apresentadas apenas classes essenciais para perceber a forma como a extensão do modelo é feita.

A condição a modelar é `exame_a_decorrer()`. Esta trata-se de uma condição do tipo `PolicyTimePeriodCondition`, que representa um período de tempo no qual a condição é verdadeira (Figura 8).

É claro que é necessário modelar o Calendário de Exames (`ExameCalendar`) que considerámos ser uma subclasse de `Calendar` e, por sua vez, de `ManagementElement` (CIM). Dado que os exames decorrem em Sala de Aula, estas são modeladas pela classe `ClassRoom`, uma especialização de `Room`, sendo, no caso de existir material informático, consideradas como `ComputerClassRoom`. Estas classes fazem parte também dos elementos geríveis (`ManagedElement`) do modelo CIM.

Este conjunto de classes permite definir a condição necessária à implementação da política acima mencionada. Resta fazer o mesmo à acção.

A acção é "`desligar comunicação entre estações de trabalho`". As estações de trabalho estão ligadas por intermédio de um *switch*, pelo que é

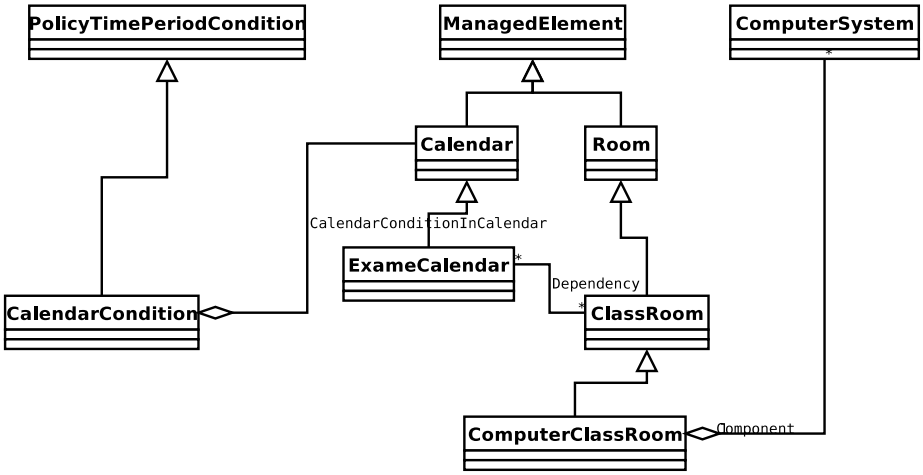

Figura 8: Condição de calendário.

possível interromper a comunicação se forem desligadas as portas associadas a cada estação de trabalho.

Há dois tipos de acções possíveis, neste cenário:

- Activar porta associada a uma estação de trabalho.
- Desactivar porta associada a uma estação de trabalho.

Além disso, é necessário iterar a acção por um conjunto de portas. Para representar a acção, definimos a classe `SwitchPolicyAction` (Figura 9).

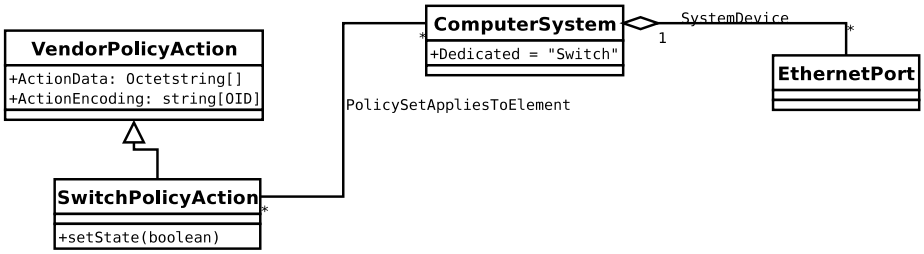

Figura 9: Especialização de acções.

A classe `VendorPolicyAction` encontra-se especificada no PCIM, enquanto que as classes `ComputerSystem` e `EthernetPort` encontram-se especificada no CIM e permitem representar um *Switch*.

As classes definidas encontram-se agrupadas no contexto ESTiGCIM, consistindo este contexto na especialização do PCIM. O DMTF definiu um mecanismo para armazenar as classes, atributos, métodos e associações num serviço de directoria, pelo que o contexto ESTiGCIM será também armazenado num servidor LDAP.

3.3 Implementação do sistema

O sistema está a ser implementado em Java, usando JNDI [16] para contactar um servidor OpenLDAP. A comunicação entre os módulos do sistema é feita com RMI e com os dispositivos em SNMP (Figura 10).

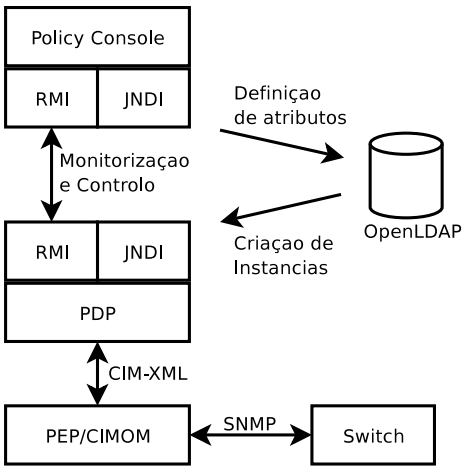

Figura 10: Arquitectura da implementação.

O PDP interage com o PEP usando CIM-XML, protocolo definido pelo DMTF para WBEM [17]. O PEP funciona como um CIMOM, responsável por manter objectos que representam os dispositivos. Estes objectos, denominados fornecedores na terminologia WBEM, implementam mecanismos para comunicar com os dispositivos, neste caso SNMP. A API de comunicação, tal como o CIMOM, assentam numa implementação *Open Source* de serviços WBEM [18]

O *switch* é configurado por intermédio de mensagens SNMP dirigidas à tabela `ifTable` [19], implementado directamente na classe `SwitchPolicyAction`.

3.4 Outras políticas

A principal característica do modelo de gestão com base em políticas é que, depois de o sistema estar modelado, é relativamente simples de definir novas políticas e de as implementar na prática. O processo de modelação requer que todos os recursos, incluíndo serviços, dispositivos, estações de trabalho ou mesmo utilizadores, estejam representados no sistema. Este processo é de suma importância e está, de alguma forma, associado ao desenvolvimento. É necessário programar como é que determinado recurso se configura e, tipicamente, associar um mecanismo de supervisão. Estes passos, apesar de não estarem modelados, encontram-se implementados nas classes que encapsulam o seu funcionamento.

A emissão de uma nova política, como por exemplo, "As salas de computadores não funcionam ao fim de semana", pode ser descrita da seguinte forma:

```
Group Alunos: Role=[fimDeSemana] {
  if(fim_de_semana())
    desligar comunicação entre estações de trabalho
}
```

O modelo de acção e de condição é semelhante ao caso anterior, pelo que não é necessário qualquer tarefa de implementação. Basta adicionar a política ao *Policy Repository* que o sistema vai, automaticamente, efectuar as configurações adequadas.

Se a política incluir novos conceitos, como por exemplo, "Os alunos com dívidas de impressão não podem utilizar a sala de computadores", já há um novo dado a acrescentar à regra:

```
Group Alunos: Role=[dívidasImpressão] {
  if(quotaPrn<0)
    desligar comunicação entre estações de trabalho
}
```

Neste último exemplo, é necessário acrescentar novos conceitos ao modelo, pelo que terão de ser criadas novas classes e os atributos associados.

4 Conclusões

Este artigo apresenta uma abordagem prática para aplicação de uma arquitectura de gestão com base em políticas (PBNM) num cenário real. Uma política reflecte opções organizacionais, definidas pelas instâncias que gerem a instituição e que, tipicamente, se encontra numa linguagem e com uma semântica bastante afastadas das operações de configuração do equipamento e dos serviços de comunicação. A política tem de ser convertida e manipulada de forma a se poder aproximar da linguagem da rede e, se possível, em instruções de configuração, como CLI, SNMP ou Web.

Este artigo define uma política a aplicar a uma sala de computadores de uma instituição de ensino superior nacional e acompanha o processo de conversão e adaptação da política até ao nível de mensagens de configuração SNMP.

Referências

[1] J.D. Case, M. Fedor, M.L. Schoffstall, and J. Davin. Simple Network Management Protocol (SNMP). RFC 1157, IETF, May 1990.

[2] ISO. *Information Processing Systems—Open Systems Interconnection—System Management Overview.* ISO/IEC, 10040 edition, 1998. ITU-T Recomendation K.701.

[3] ISO. *Information Technology—Open Systems Interconnection—Common Management Information Protocol–Part1 Protocol Specification.* ISO/IEC, 9596-1 edition, 1998. ITU-T Recommendation K.711.

[4] A. Bierman. SNMP set: Can it be saved? *The Simple Times*, 9(1), December 2001. `http://www.simple-times.org/pub/simple-times/issues/9-1.html` last accessed on the 15/4/2004.

[5] M. Sloman. Policy driven management for distributed systems. *Journal of Network and Systems Management*, 2(4), December 1994. `http://www-dse.doc.ic.ac.uk/dse-papers/management/pdman.ps.Z` last accessed on the 15/4/2004.

[6] R. Wies. Policies in network and system management – formal definition and architecture. *Journal of Network and Systems Management*, 2(1), January 1994.

[7] S. Naqvi J. Chomicki, J. Lobo. Conflict resolution using logic programming. *IEEE Transactions on Knowledge and Data Engineering*, 15(2), March 2003.

[8] J. Strassner. *Policy Based Network Management – Solutions for the Next Generation*. Morgan Kaufman, 2003.

[9] D. Durham, Ed., J. Boyle, R. Cohen, S. Herzog, R. Rajan, and A. Sastry. The COPS (Common Open Policy Service) Protocol. RFC 2748, IETF, January 2000.

[10] M. MacFaden, D. Partain, J. Saperia, and W. Tackabury. Configuring Networks and Devices with Simple Network Management Protocol (SNMP). RFC 3512, IETF, April 2003.

[11] DMTF. *Common Information Model - Core Model, v2.6*. DMTF, cim-core edition, 2002.

[12] B. Moore, E. Ellesson, J. Strassner, and A. Westerinen. Policy Core Information Model – Version 1 Specification. RFC 3060, IETF, February 2001.

[13] A. Saxena K. Law. *Performance of a Multi-Tiered Policy-Based Management System*, pages 203–214. Network Control and Engineering for QoS, Security and Mobility. Kluwer Academic Press, 2003.

[14] Intel Corp. *Preboot Execution Environment (PXE) Specification version 2.1*, September 1999.

[15] DMTF. *CIM Core Model V2.5 - LDAP Mapping Specification*. DMTF, cim-ldap edition, 2002.

[16] Sun Microsystems. *Java Naming and Directory Interface (JNDI)*. Sun Microsystems. `http://java.sun.com/products/jndi/` last accessed on the 16/7/2004.

[17] DMTF. *Web-based Enterprise Management (WBEM) Initiative*. `http://www.dmtf.org/standards/wbem/` last accessed on the 16/7/2004.

[18] Sun Microsystems. *WBEM Services*. Sun Microsystems. `http://wbemservices.sourceforge.net/` last accessed on the 16/7/2004.

[19] K. McCloghrie and M.T. Rose. Management Information Base for Network Management of TCP/IP-based internets:MIB-II. RFC 1213, IETF, March 1991.