

Descrição e implementação de uma MIB para sistemas MASIF

Rui Pedro Lopes¹, José Luis Oliveira²

¹Instituto Politécnico de Bragança, ESTiG, 5300 Bragança (rlopes@ipb.pt)

²Universidade de Aveiro, DET-IEETA, 3810 Aveiro (jlo@det.ua.pt)

Palavras chave: SNMP, MASIF, Agentes Móveis.

Resumo

Este artigo descreve a arquitectura, o desenvolvimento e alguns resultados de um conversor SNMP para MASIF. O sistema permite gerir plataformas de agentes compatíveis com a especificação MASIF através de qualquer aplicação gestora SNMP, simplificando, desta forma, a integração de tecnologia de agentes móveis com plataformas de gestão clássicas. A estrutura de informação associada pode funcionar igualmente com uma interface de gestão para sistemas não compatíveis com MASIF.

I. Introdução

Os agentes de *software* são programas que desempenham determinada função em benefício de outro programa ou do utilizador [1]. São vários, os campos de aplicação desta tecnologia, desde a pesquisa de informação [2], comércio electrónico [3], mobilidade pessoal, telecomunicações e interfaces com o utilizador [4]. Um agente com capacidade de se movimentar ao longo da rede é designado por agente móvel.

Os agentes móveis podem, portanto, ser criados em qualquer ponto da rede, interromper temporariamente a execução, deslocar o código e o estado e retomar a execução num outro ponto da rede. Os agentes móveis baseiam-se em conceitos como: o de lugar (*place*), de agência (*agency* ou *agent system*) e de região (*region*). Um lugar, entre outras funções, estabelece uma abstracção (do ponto de vista do agente móvel) sobre a plataforma que o alberga. Esta função é importante, principalmente, do ponto de vista de segurança, uma vez que permite o acesso a recursos e serviços específicos apenas a agentes ali localizados. Uma agência cria um ambiente independente do sistema operativo, o que facilita tarefas como a migração de agentes, e alberga diversos lugares e agentes. Tipicamente, existe uma agência por computador. A maior parte das plataformas de agentes móveis criam estruturas semelhantes a esta. Há, no entanto, problemas de interoperabilidade se os agentes pretendem deslocar-se entre agências de diferentes fabricantes.

A interoperabilidade entre plataformas de agentes assenta em dois campos. Um deles, em desenvolvimento pela FIPA [5], visa normalizar a comunicação entre agentes bem como a linguagem utilizada. Além da tecnologia de comunicação, há também a preocupação de definir protocolos de negociação em áreas específicas, tais como multimédia, viagens, serviços de rede, produção, entre outros. O segundo campo assegura a interoperabilidade entre ambientes de execução e encontra-se em

desenvolvimento pelo OMG sob a forma de um documento formal – MASIF. Esta especificação tem como objectivo permitir que um agente móvel possa deslocar-se entre agências com perfil semelhante (partilhando a mesma linguagem, tipo de agência, tipo de autenticação e método de serialização) através de um conjunto de interfaces CORBA normalizadas [6].

II. A especificação MASIF

A especificação MASIF consiste no primeiro documento de normalização de operações em sistemas de agentes móveis visando, assim, a interoperabilidade entre sistemas de diversos construtores. Esta, define um conjunto de conceitos e interfaces (em IDL – *Interface Definition Language*), procurando a simplicidade de forma a facilitar o desenvolvimento futuro de sistemas de agentes móveis [6].

A. Acções comuns em sistemas de agentes móveis

Independentemente da tecnologia que o suporta, um agente móvel é constituído por duas partes: o código – que descreve o comportamento do agente – e o estado – o valor das variáveis internas. Ambos são mantidos durante a migração, o que significa que o agente não só sabe o que fazer como sabe também o que fez.

Os agentes móveis possuem um ciclo de vida composto por estados e transições bem definidas (Fig. 1).

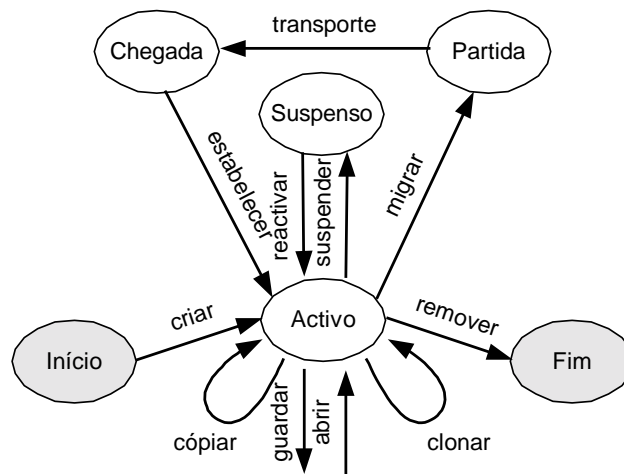


Fig. 1 – Ciclo de vida de um agente móvel.

A gestão de agentes móveis actua sobre diversas fases do ciclo de vida:

- Criar – tarefas de inicialização tais como criar as estruturas de dados e iniciar a execução do agente. Cada agente é baseado num conjunto de classes que têm de ser enviadas à agência. Este conjunto de classes é localizado por uma cadeia de caracteres (*string*) – o *code base*.
- Remover – actividades de fim de execução.
- Suspender – interromper temporariamente a actividade do agente.
- Reactivar – reiniciar a execução do agente.

- Clonar – criar outra instância do agente no mesmo lugar.
- Copiar – criar outra instância do agente em outra localização.
- Migrar – transferir o agente. Esta acção requer o conhecimento do destino e qual o protocolo de transporte (*sockets*, *SSL*, *RPC* ou outros).
- Guardar – armazenar permanentemente a informação interna do agente. Esta informação permite reiniciar um agente que tenha sido destruído inadvertidamente por alguma falha involuntária.
- Abrir – recupera o agente móvel do dispositivo de armazenamento permanente.
- Invocação – permite invocar acções definidas pelo programador.

Apesar de não estar directamente relacionadas com o ciclo de vida do agente, um sistema de agentes móveis necessita de um mecanismo de registo – a região – de forma a tornar possível a pesquisa de agentes, lugares e agências. Este mecanismo assegura a identificação unívoca no interior do domínio definido.

Por razões de segurança, nomeadamente, autenticação e controlo de acesso, uma agência identifica a autoridade que enviou o agente. É com base nesta que recursos como o sistema operativo, discos, processador, memória ou outros são protegidos contra actos ilícitos de agentes maliciosos.

B. Interfaces MASIF

A norma MASIF especifica duas interfaces em IDL que constituem a base para todas as operações sobre a agência e a região (Fig. 2).

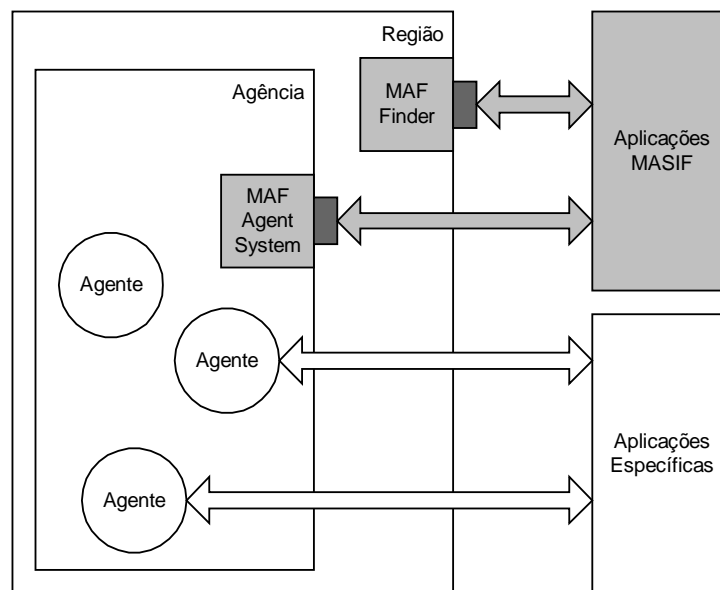


Fig. 2 – Arquitectura MASIF.

A interface `MAFFinder`, ligada à região, consiste num ponto de acesso ao serviço de directoria para agências, lugares e agentes. Os seus métodos permitem funções de pesquisa e catalogação:

Funcionalidade	Métodos
Registo	void register_agent(...) void register_agent_system(...) void register_place(...) void unregister_agent(...) void unregister_agent_system(...) void unregister_place(...)
Pesquisa	Locations lookup_agent(...) Locations lookup_agent_system(...) Locations lookup_place(...)

A interface `MAFAgentSystem` define métodos e objectos que suportam tarefas como consultar o nome de uma agência ou receber um agente, entre outras:

Funcionalidade	Métodos
Ciclo de vida	Name create_agent(...) void receive_agent(...) void resume_agent(...) void suspend_agent(...) void terminate_agent(...) OctetStrings fetch_class(...) void terminate_agent_system(...)
Informação	Location find_nearby_agent_system_of_profile(...) AgentStatus get_agent_status(...) AgentSystemInfo get_agent_system_info(...) AuthInfo get_authinfo(...) MAFFinder get_MAFFinder(...) NameList list_all_agents() NameList list_all_agents_of_authority(...) Locations list_all_places()

Associadas a estas interfaces encontram-se definidas várias estruturas de dados:

- `Name` - com três atributos (`authority`, `identity` e `agent_system_type`) que asseguram a individualidade do agente.
- `ClassName` - define a sintaxe para nomes das classes que, quando instanciadas, dão origem ao agente. Esta informação é utilizada nos métodos `create_agent()` e `receive_agent()`, por exemplo.
- `Location` - consiste numa sequência de caracteres contendo: a) um URI (*Universal Resource Identifier*) com um identificador CORBA [8] ou b) um URL (*Universal Resource Locator*) contendo um endereço da Internet [9]. A vantagem do identificador CORBA é a independência do protocolo enquanto que os URLs são mais adequados à Internet.

III. Gestão de agentes móveis por SNMP

A arquitectura SNMP é, actualmente, a mais divulgada em termos de equipamento e *software*. A sua utilização ao longo de mais de uma dezena de anos criou um conhecimento técnico que outras arquitecturas não possuem. De forma que, qualquer nova ferramenta ou arquitectura de gestão, terá de lidar com as normas SNMP e não o contrário.

Para que a tecnologia de agentes móveis possa contribuir de forma válida terá de passar por uma fase de integração com o SNMP [10]. Esta integração poderá ser efectivada em duas vertentes. Por um lado, poder-se-á tirar partido de aplicações de gestão móveis no sentido em que os agentes móveis terão a capacidade de monitorar e controlar agentes SNMP clássicos. Por outras palavras, a solução consiste no enriquecimento de aplicações de gestão clássicas com mobilidade [11].

Esta solução tem como inconvenientes aumentar o tamanho e a complexidade do agente móvel resultante da adição do mecanismo de gestão. O código a transportar poderá, portanto, ser excessivo. De forma a atenuar este inconveniente, poder-se-ão agrupar os serviços comuns, como os existentes na arquitectura SNMP, na agência, o que pode não eliminar por completo as desvantagens apontadas anteriormente. Nestas situações pode ser desenvolvida uma interface que permita obter uma solução mais eficiente [12].

A segunda vertente tem a ver com a própria gestão de agentes móveis. A gestão de mobilidade não implica a integração com o SNMP, ou seja, a gestão do ambiente pode ser previsto pelo próprio sistema. Há, no entanto, vantagens em utilizar a arquitectura SNMP, uma vez que permite utilizar a mesma ferramenta para a monitorização e controlo do equipamento de rede e dos próprios agentes. Para resolver este problema, a solução óbvia será integrar a pilha protocolar SNMP no código do agente. Tal como apontado anteriormente, esta medida pode fazer com que o código seja excessivamente extenso. Por outro lado, a arquitectura de gestão pode ser providenciada pela agência, criando assim uma porta para todo o sistema sem sobrecarregar os agentes.

A. *Gestão de agentes móveis*

A utilização de agentes móveis para a gestão de redes poderá trazer vantagens relacionadas com a continuação de operação em situações sem conectividade, com o equilíbrio de carga, entre outras [13][14]. Do ponto de vista do utilizador, habituado a ferramentas de gestão baseadas em SNMP, é de todo vantajoso poder visualizar o estado de funcionamento dos agentes móveis bem como poder modificar o seu comportamento em conjunto com a gestão de aplicações e de equipamento de rede. Desta forma, os próprios agentes, lugares e agências passam a “fazer parte” da rede do ponto de vista de administração, facilitando assim a integração de tecnologia.

A gestão de agentes móveis através da agência vai de encontro à arquitectura MASIF, em que as próprias interfaces se encontram acessíveis nas agências e na região. Aproveitando este facto, o desenvolvimento de um conversor SNMP para MASIF permitirá gerir qualquer plataforma de agentes móveis compatível com MASIF sem a sobrecarregar com a adição de código, uma vez que o agente SNMP fica isolado da plataforma pela invocação remota de métodos da arquitectura CORBA (Fig. 3).

Nesta arquitectura, o conversor SNMP para MASIF, a partir da mensagem recebida, assegura a correcta identificação dos métodos a invocar nas interfaces MASIF. Em termos de segurança, a utilização da arquitectura SNMPv3 garante a autenticação, privacidade e controlo de acesso de acordo com o *User-based Security Model* [15] e o *View-based Access Control Model* [16].

Em resumo, o conversor realiza a invocação de métodos MASIF a partir de mensagens SNMP.

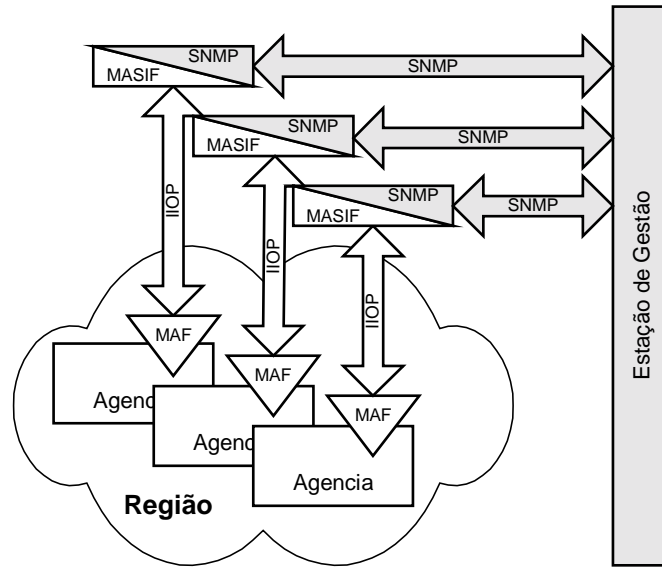


Fig. 3 – Conversor SNMP para MASIF.

B. MASIF-MIB

A arquitectura SNMP descreve um protocolo de transporte de informação. Esta encontra-se estruturada em forma de árvore e é denominada MIB (*Management Information Base*) [17]. A função da MIB é definir todos os objectos relevantes para a monitorização e controlo do agente SNMP.

No âmbito do conversor, a MIB é utilizada para identificar o método MASIF a invocar e os seus argumentos, de forma a conseguir uma gestão coerente da plataforma de agentes móveis. Por exemplo, quando a estação de gestão (NMS) envia um *set-request*, a mensagem transporta um identificador e um valor. O identificador refere um objecto particular da MIB que, por sua vez, se encontra associado ao método MASIF apropriado. O valor é utilizado como argumento ao método. Por exemplo, se a intenção do utilizador é criar um agente, terá de enviar um *set-request* com os argumentos necessários, ou seja, localização inicial do agente (*home*), nome da classe (*ClassName*), localização do código (*code base*), entre outros.

O funcionamento do conversor SNMP para MASIF assenta numa MIB desenvolvida para o efeito. De acordo com a especificação MASIF, a MASIF-MIB foi estruturada em dois grandes grupos: *mamObjects* e *mamLookup* (Fig. 4).

O grupo *mamObjects* está relacionado com as tarefas de gestão disponibilizadas pela interface *MAFAgentSystem* e possui objectos que permitem, em primeiro lugar, consultar parâmetros de funcionamento e obter características da plataforma, tais como a autoridade, identificação, tipo, descrição e versão da agência (*mamSysID*), linguagens suportadas (*mamLanguageTable*), listagem de lugares (*mamPlaceTable*) e agentes (*mamAgentTable*). Além de consulta, é possível actuar sobre os agentes, nomeadamente, suspender, terminar e reactivar a sua execução, por intermédio da *mamAgentTable*. A criação de agentes é efectuada no grupo *mamCreateAgent*.



Fig. 4 – Estrutura global da MASIF-MIB.

Como exemplo de utilização da MIB, suponha-se que o utilizador necessita avaliar o estado de conectividade de vários troços da rede. Para o efeito, actua sobre o grupo `mamCreateAgent` no sentido de fornecer a informação necessária para que o agente seja criado, nomeadamente:

- `ClassName` – “`nms.mobile.ConnectivityEvaluation`”
- `Code base` – “`file:/usr/local/nms/classes`”
- Nome do lugar inicial – “`pingPlace`”
- Activar agente – 1.

Quando criado, o agente inicia o teste de conectividade, armazenando o tempo de resposta nos diversos troços. De seguida, por intermédio da região, o agente percorre as agências disponíveis continuando a operação até ter visitado todos os pontos disponíveis.

Supondo que, em determinado momento, o utilizador decide que o teste de conectividade não é relevante e, portanto, decide terminar o agente. Para o efeito, envia um `set-request` à `mamAgentTable` com o estado pretendido de `terminate`. Esta mensagem será traduzida para uma invocação MASIF e o agente será eliminado.

O grupo `mamLookup` assenta nos serviços de pesquisa providenciadas pela interface `MAFFinder`, nomeadamente, pesquisa de agentes, lugares e agências. Antes de poder actuar sobre uma determinada agência ou agente, o utilizador terá de a/o localizar. A chave de pesquisa depende do tipo de objecto a procurar. Por exemplo, a localização de lugares é efectuada unicamente por intermédio do nome. Não há outras características relevantes para este objecto. Para agentes e agências, outras chaves de pesquisa podem ser importantes, como, por exemplo, a linguagem suportada ou o método de serialização.

Por estes motivos, a chave de pesquisa é baseada num filtro de forma a permitir pesquisas mais complexas:

```
Filtro = item (|,&) filtro /* ou, e */
item = chave comparador valor
comparador = (=,^,$,~) /* igual, começa com, termina com, contém */
chave = string /* um entre: Name, Codebase, Language, Serialization, ...*/
valor = string
```

O filtro “`Name=pingAgent&Codebase~myHost`” é um exemplo de pesquisa de um agente com o nome “`pingAgent`” e que possui as classes em “`myHost`”. Do ponto de vista de funcionamento, esta chave é enviada ao conversor, em particular ao grupo `mamLookupAgents`, e os resultados são consultados numa tabela no mesmo grupo.

C. O conversor SNMP para MASIF

A MIB apresentada é associada a outros módulos de forma a implementar o agente SNMP (Fig. 5):

1. O módulo de comunicações, normalmente uma pilha SNMP.
2. O modulo de processamento de mensagens, responsável pela ligação entre a MIB e o módulo de comunicações.
3. A estrutura de dados virtual, ou seja, a MIB, responsável por invocar os métodos MASIF adequados.

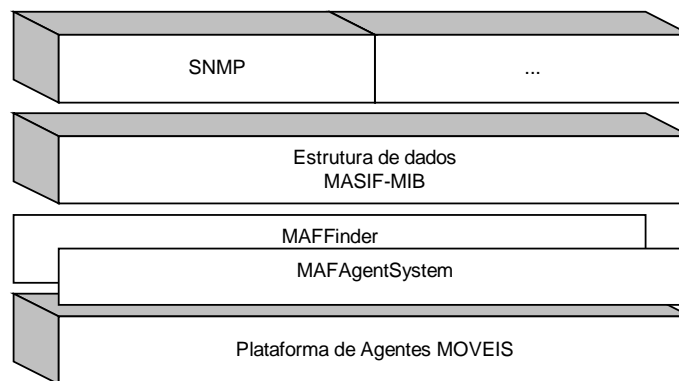


Fig. 5 – Estrutura do conversor.

O estabelecimento de uma interface apropriada entre o módulos de comunicações e o de processamento de mensagens possibilita a utilização de vários protocolos, mantendo a estrutura do agente inalterada. Esta capacidade permite utilizar SNMP, CORBA, RMI ou outros, tais como dotar o agente de mobilidade.

Na implementação usámos a plataforma de agentes móveis Grasshopper com o módulo de compatibilidade MASIF [6]. Usámos, para simplicidade, o ORB do JDK1.3 bem como o serviço de registo CORBA do JDK (*tnameserv*). Para a pilha SNMPv3 usámos a da AdventNet (<http://www.adventnet.com/>).

O módulo de comunicações faz a adaptação da mensagem recebida para invocação de métodos. Para o caso do SNMPv3, este módulo implementa os mecanismos de segurança e de processamento de mensagem definidos pelo IETF, incluindo USM (*User-based Security Model*) e VACM (*View Access Control Model*).

O módulo de processamento de mensagens, de acordo com o método invocado pelo módulo de comunicações, identifica e dirige a mensagem (neste caso a invocação do método) para o objecto adequado. Em termos de implementação, o módulo possui uma árvore binária (ordenada) onde todos os objectos implementados se registam (por exemplo, o *mamPlaceTable* ou o grupo *mamCreateAgent*).

Cada um destes objectos é responsável por acções definidas na MIB. Assim, o *mamLanguageType* é responsável por invocar o método MASIF que lhe permite averiguar quais são as linguagens suportadas pelo sistema de agentes. Outros objectos têm outras responsabilidades, como, por exemplo, averiguar a versão, pesquisar agentes, e assim sucessivamente.

Este cenário permite utilizar um simples MIB Browser para monitorar e controlar qualquer sistema de agentes compatível com MASIF (Fig. 6). Actividades como, por exemplo, parar, reiniciar e mover agentes passam a poder ser efectuadas utilizando a mesma metáfora de gestão da rede.

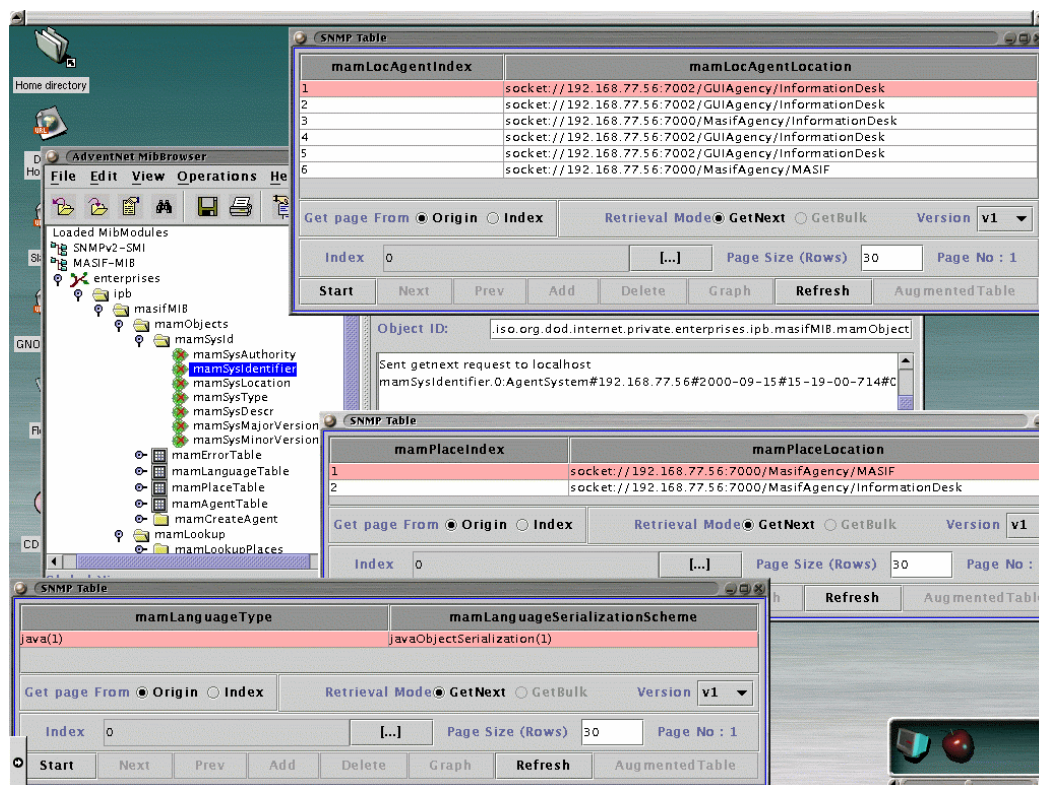


Fig. 6 – A captura de ecrã de uma sessão utilizando um MIB browser: mamAgentTable, mamPlaceTable, mamLanguageTable e a janela principal do MIB browser.

De acordo com a especificação MASIF, é necessário um conversor para cada agência, tal como para o caso do equipamento de rede em que, tipicamente, cada um está associado a um agente SNMP.

As interacções entre a estação de gestão e a plataforma de agentes móveis envolvem, tipicamente, três tipos de comunicação: 1) entre o NMS/MIB Browser e o conversor é utilizado o SNMP; 2) este último utiliza o IIOP (*Internet InterORB Protocol*) para contactar a agência; 3) no interior da plataforma de agentes móveis são usados protocolos específicos (geralmente, *sockets*, SSL, RMI ou outros).

IV. Conclusões

As vantagens advindas da utilização da tecnologia de agentes móveis em gestão de redes estão ainda por esclarecer, embora existam boas perspectivas para esta associação. Atendendo à importância que o SNMP tem nesta área, a coexistência entre as duas tecnologias parece ser um passo natural. Apesar disso, poucos trabalhos apontam nessa direcção.

Neste artigo foi proposta uma arquitectura que permite gerir plataformas MASIF através de qualquer solução de gestão baseada em SNMP. Para o efeito, é igualmente proposta uma MASIF-MIB e descrito o conversor SNMP para MASIF.

V. Referências

- [1] V. Pham, A. Karmouch, “Mobile Software Agents: An Overview”, *IEEE Communications, Vol. 36, No. 7* (1998) pp. 26-37.
- [2] I. Nekrestyanov, T. O’Meara, A. Patel, E. Romanova, “Building Topic-Specific Collections with Intelligent Agents”, *actas da 6th International Conference on Intelligence in Services and Networks, IS&N’99*, Barcelona, Espanha, Abril 1999.
- [3] M. Gleizes, A. Léger, E. Athanassiou, P. Glize, “Self-Organization and Learning in MultiAgent Based Brokerage Services”, *actas da 6th International Conference on Intelligence in Services and Networks, IS&N’99*, Barcelona, Espanha, Abril 1999.
- [4] S. Green, L. Hurst, B. Nangle, P. Cunningham, F. Somers, R. Evans, *Software Agents: A review*, Maio 1997.
- [5] Foundation for Intelligent Physical Agents, (<http://www.fipa.org/>).
- [6] Mobile Agent Facility Specification, Object Management Group, 00-01-02.pdf (<ftp://ftp.omg.org/pub/docs/formal/00-01-02.pdf>).
- [7] Grasshopper Agent Platform, (<http://www.grasshopper.de/>).
- [8] T. Berners-Lee, “Universal Resource Identifiers in WWW”, RFC1630, Junho 1994.
- [9] T. Berners-Lee, L. Masinter, “Uniform Resource Locators (URL)”, RFC1738, Dezembro 1994.
- [10] J. Oliveira, R. Lopes, “Agents in SNMP Framework”, *Actas do Spanish Symposium on Distributed Computing, SEID2000*, Ourense, Espanha, Setembro 2000.
- [11] J. Oliveira, R. Lopes, “Distributed Management based on Mobile Agents”, *Proc. of the 1st International Workshop on Mobile Agents for Telecommunications Applications – MATA’99*, Ottawa, Canada, Outubro 1999.
- [12] B. Pagurek, Y. Wang, T. White, “Integration of Mobile agents with SNMP: Why and How”, (<ftp://ftp.sce.carleton.ca/pub/netmanage/integration.ps.gz>).

- [13] G. Goldszmidt, Y. Yemini, "Delegated Agents for Network Management", *IEEE Communications Magazine*, Vol. 36 No. 3 (1998) pp. 66-71.
- [14] R. Lopes, J. Oliveira, "On the use of Mobility in Distributed Network Management", *actas da Hawaii International Conference on System Sciences – HICSS*, Maui, Hawaii, EUA, Janeiro 2000.
- [15] U. Blumenthal, B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", RFC2573, Abril 1999.
- [16] B. Wijnen, R. Presuhn, K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)", RFC2575, Abril 1999.
- [17] K. McCloghrie, D. Perkins, J. Schoenwaelder, J. Case, M. Rose, S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, Abril 1999.