

Medioambiente de evaluación de soportes de comunicación grupal para agentes móviles

Guillermo Rigotti

UNICEN – Fac. de Ciencias Exactas-ISISTAN

Pje. Arroyo Seco, (7000) Tandil, Bs. As. Argentina

TE: +54-2293-440363 FAX: +54-2293-440362 Email: grigotti@exa.unicen.edu.ar

Resumen

En los sistemas de agentes móviles, la capacidad de comunicación entre los agentes y con la aplicación que les da origen resulta indispensable, principalmente para la interacción entre los agentes, a efectos de mejorar la performance de la aplicación, y para actividades de managment sobre los mismos. En estos casos, se hace necesaria la capacidad de comunicación grupal o multicast, que permite la transferencia de información de un agente o de la aplicación a varios agentes identificados como integrantes de un grupo. Si bien la mayoría de los sistemas de agentes móviles ofrece comunicación grupal a través de facilidades de conformación de grupos y de envío de información a grupos, ésta se limita generalmente a entornos de red local o a aplicaciones que cubriendo mayores áreas, involucran a una pequeña cantidad de agentes. En casos de aplicaciones con grandes cantidades de agentes cubriendo áreas extensas de la Internet, los mecanismos adoptados (transmisiones individuales o uso de broadcast) no resultan aplicables o escalables. Se hace necesario contar entonces con soportes de comunicación grupal que sustenten esta capacidad ofrecida por los sistemas de agentes móviles.

Se presenta aquí un soporte que permite el desarrollo y evaluación en ambientes reales de diferentes heurísticas relacionadas con la comunicación grupal de agentes móviles. En particular, el servidor simple de agentes móviles implementado, soporta aplicaciones de agentes móviles simplificadas, focalizadas en aspectos de movilidad y comunicación grupal, y permite desarrollar funciones de soporte que se agregan en forma de módulos independientes al servidor de agentes. De esta manera es posible evaluar su comportamiento en ambientes reales. En particular, se encuentran en desarrollo un módulo multicast, para la implementación de diferentes heurísticas de armado y mantenimiento de árboles de distribución sobre la red integrada por los servidores de agentes móviles, y un módulo de chequeo de la red, que permite ajustar el mencionado árbol a la topología real y carga de la red con el objeto de mejorar la performance.

Palabras clave: agentes móviles, comunicación, multicast

1. Introducción y objetivos

En las aplicaciones de agentes móviles, un agente o conjunto de agentes generados inicialmente se traslada de un servidor a otro posiblemente generando copias (clones) de ellos mismos para llevar a cabo las funciones necesarias para cumplir con su objetivo.

Entre los escenarios de uso en los cuales estos sistemas son atractivos, se puede mencionar aquellos en los cuales al migrar los agentes a los lugares donde se encuentra la información a procesar, evitan la carga en la red y la demora resultantes de transferir volúmenes importantes de datos sobre los que deben operar; otro ámbito de aplicación lo constituye la comunicación de dispositivos sobre redes wireless, ya que reducen en gran medida la vulnerabilidad de la aplicación ante desconexiones temporarias de la red.

Pese a las indiscutibles ventajas que presenta esta tecnología, no existe aún una aceptación generalizada que les permita ser incluida en desarrollos comerciales. Esta situación se debe en parte a la carencia de soporte que permita su escalabilidad.

En este aspecto, es de destacar la situación referente a comunicaciones grupales: además de la comunicación necesaria para el proceso de migración, debe considerarse la producida por las interacciones entre la aplicación de origen y los agentes, y de estos entre sí. Por ejemplo para managment en el primer caso, y para la comunicación de un agente a los demás de resultados parciales obtenidos, a efectos de mejorar la performance de la aplicación en el segundo caso.

En ambos casos, los destinatarios de la información son la totalidad o grupos de agentes, característica que hace que adquiera importancia el concepto de comunicación grupal (uno a varios) en los sistemas de agentes móviles.

Sin embargo, si bien estos sistemas ofrecen posibilidad de comunicación grupal a alto nivel a través de facilidades para la conformación de grupos y el envío de mensajes a estos grupos, existe un vacío entre esta facilidad ofrecida y la comunicación real a llevarse a cabo en la red: las alternativas empleadas para concretar la comunicación grupal son en general, utilizar transmisión broadcast en la red local, que no es escalable a grandes áreas de la Internet, o realizar transmisiones individuales a cada miembro del grupo, que no escala a aplicaciones con gran número de agentes.

Una tercera alternativa, que en principio resultaría la más natural y eficiente, es el uso de las facilidades multicast provistas por el nivel de red. Estas consisten en el soporte provisto por los routers al tráfico grupal, principalmente la construcción y mantenimiento de árboles de distribución que permiten que la información emitida por un nodo a un grupo, sea enviada a todos los integrantes del mismo. Sin embargo, para implementar este tipo de facilidades, los routers deben ser configurados explícitamente, requiriendo protocolos específicos y consumiendo recursos de proceso y ancho de banda. Pese a que la transmisión multicast nativa en la Internet data de 1989 [Deering, 1989], aun no se ha extendido a grandes áreas de la red, contándose con una red virtual de prueba (MBONE) [Savetz, 1998] que tampoco ha logrado extenderse a la totalidad de la Internet. Otro problema grave que presenta este soporte, es el uso de protocolo UDP [Postel, 1980], no confiable, lo que hace que sea costoso proveer transmisión confiable a las aplicaciones. En realidad, para proveer confiabilidad a las aplicaciones, requerimiento de suma importancia en el tipo de aplicación que nos ocupa, debe implementarse un nivel denominado multicast confiable [Kermode, 2002], que corre sobre UDP. Para una caracterización de los requerimientos de este tipo de aplicaciones, ver [Rigotti, 2003], [Rigotti, 2004].

Debido a los inconvenientes que presentan las alternativas mencionadas, se ha recurrido a otra que consiste en la implementación de multicast en un nivel superior a IP, residente en los hosts [Janotti, 2000] [Chu, 2000] [Castro, 2002]. Si bien esta alternativa resulta ventajosa desde el punto de vista de la flexibilidad en la provisión de servicios, su facilidad de implementación y su extensibilidad en la red, debe tenerse en cuenta la eficiencia de los árboles multicast construidos, ya que los equipos finales no disponen en principio de la información topológica necesaria.

En particular, en el caso de aplicaciones de agentes móviles, la aplicación de una solución de este tipo resulta atractiva debido a lo siguiente

1-No requiere soporte específico en la red: esta característica es de suma importancia, ya que independiza a la aplicación de agentes móviles del soporte multicast ofrecido en las subredes involucradas en la aplicación, que no son conocidas a priori debido a la movilidad de los agentes.

2-Los eventuales nodos de soporte multicast son los servidores de agentes móviles: esto elimina el problema de la baja confiabilidad de los nodos que componen el árbol de distribución multicast en redes overlay, ya que ellos son equipos de usuarios susceptibles de fallas y apagado, casos en los cuales se desencadenan procesos muy costosos de rearmado del árbol de distribución multicast. La confiabilidad de un servidor de agentes podría equipararse con la de un router

3-Es posible utilizar conexiones TCP para conformar el árbol de distribución, resultando más simple implementar el servicio de multicast confiable a niveles superiores. Este es el motivo por el cual se utilizan conexiones TCP en el desarrollo presentado en este paper.

4-Es posible distribuir implementaciones de diferentes heurísticas de la funcionalidad a proveer (en nuestro caso, en una primera etapa, soporte multicast) a los servidores en forma de agentes móviles especiales (por el momento esta característica no se halla implementada).

5-Una característica propia de las aplicaciones de agentes móviles respecto de la generación de nuevos nodos integrantes del árbol de distribución que puede ser explotada por las diferentes heurísticas de construcción del árbol es la siguiente: la generación de un nuevo miembro del grupo

¹ En realidad, para proveer confiabilidad a las aplicaciones, debe implementarse un nivel denominado multicast confiable, que corre sobre UDP

se dará sólo cuando se produce el traslado de un agente a un nuevo servidor, el agente podrá proporcionar al nodo, un conjunto de servidores posiblemente pertenecientes al árbol (incluido el raíz) de entre los cuales se podrá seleccionar el padre.

Debido al vacío existente entre la provisión de facilidades multicast por parte de los sistemas de agentes móviles y la falta de una implementación escalable a más bajo nivel, que efectivamente posibilite el transporte de la información más allá de el ámbito local y a gran número de agentes, se intenta en este trabajo proveer un medioambiente que permita el desarrollo de heurísticas multicast para tal fin, y su comprobación en ambientes reales². Teniendo en cuenta que es difícil comprobar el comportamiento de aplicaciones prototipo en un gran número de nodos, se prevé en un futuro inmediato integrar el ambiente desarrollado con un soporte de simulación, que permita evaluar el comportamiento a gran escala de las heurísticas a desarrollar.

En resumen, el trabajo presentado tiene como objeto, en una primera etapa, el desarrollo de facilidades de comunicación multicast a ofrecer a los sistemas de agentes móviles.

Es decir, es posible desarrollar diferentes alternativas de soporte multicast a ser ofrecido a los agentes móviles, e incluirlas en el sistema de soporte, lo cual permite luego comprobar su funcionamiento en casos reales (acotados desde ya por la cantidad de equipos que es posible utilizar en una aplicación real –no simulada-). La comprobación a través de simulación, serviría para verificar los resultados obtenidos en el caso real, a escenarios con grandes cantidades de agentes dispersos en toda la Internet.

Para no tener que realizar dos implementaciones diferentes del soporte multicast y de la aplicación, uno para la implementación del sistema real y otro para la simulación, se contemplan alternativas como la posibilidad de modificar Java Network Simulator [JNS], para que sea posible un traslado relativamente sencillo de la aplicación real a la simulada, o implementar un nivel de emulación de red que reemplace los servicios provistos por el sistema operativo.

El presente trabajo está organizado de la siguiente manera. En la sección 2, se describen brevemente las características de un sistema de agentes móviles, tomado como referencia para el desarrollo realizado; en la sección 3 se describe la estructura del sistema y sus distintos componentes, completándola en la sección 4 con una descripción de la interacción entre los diferentes elementos del sistema. En la sección 5 se describe, a título de ejemplo, el módulo multicast en desarrollo, finalizando con las conclusiones y trabajos a realizar en la sección 6.

2. Sistema de agentes móviles utilizado como referencia

En los siguientes párrafos se describe de manera general la estructura de los sistemas de agentes móviles tomada como base para el desarrollo realizado.

Una aplicación cliente representa al usuario del sistema. Esta aplicación está situada en un equipo, posiblemente de propiedad del usuario, y para cumplir con sus objetivos, debe interactuar con diferentes equipos remotos (servidores) que le proveen los servicios necesarios. De acuerdo con el paradigma de agentes móviles, la aplicación cliente se basa en las funciones realizadas por entidades activas, autónomas (los agentes móviles) que se desplazan de servidor en servidor para ir cumpliendo parcialmente los objetivos. Estos agentes interactúan en cada servidor, con la parte de la aplicación que localmente les provee el servicio necesario.

Como puede verse, los agentes necesitan por un lado un soporte básico general que les permita desplazarse entre los servidores, y comunicarse entre sí y con la aplicación cliente que los origina, a efectos de poder realizar trabajo colaborativo y ser controlados, y por otro lado con un

² Si bien el desarrollo realizado fue motivado y tiene por objetivo la incorporación de funciones multicast al servidor de agentes, es posible incluir otros tipos de funcionalidad sin necesidad de realizar cambios al soporte.

soporte a nivel aplicación, que les permita solicitar los servicios de nivel aplicación en cada uno de los servidores donde se ejecutan.

El primero de los servicios es el que nos concierne, siendo provisto por los servidores de agentes móviles. Para el servicio a nivel aplicación, está prevista la utilización de agentes estáticos cuyo objetivo es proveer una interfaz genérica para la interacción de los agentes móviles con el servidor de aplicación. De esta manera, el sistema en general resulta adaptable a cambios en la aplicación

En la figura 1 se muestra la estructura general de un sistema de agentes móviles, tal como es considerado en este trabajo.

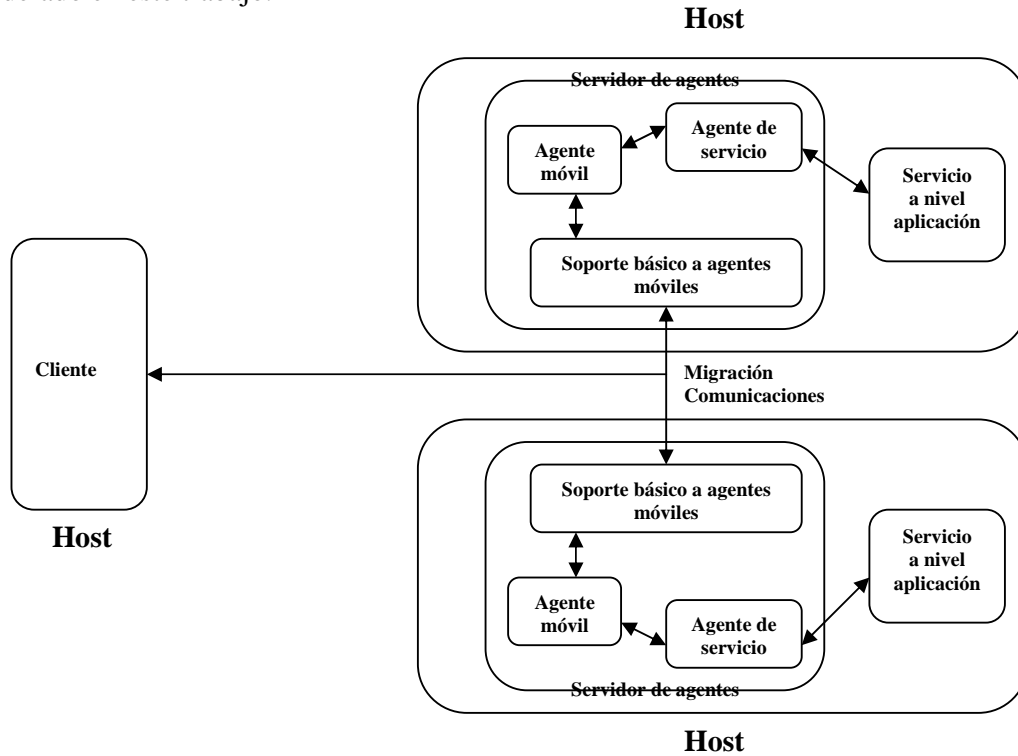


Fig.1 Estructura general de un sistema de agentes móviles

3. Estructura y componentes del sistema

Con el objeto de permitir la independencia entre los módulos que ofrecen las diferentes funciones, y de ellos con el servidor de agentes y con la aplicación (representada por los agentes móviles), el sistema se estructuró en tres niveles.

El nivel superior representa a la aplicación, compuesta por el cliente y los agentes móviles. Estos últimos, que son los que revisten interés para nuestro trabajo, deben implementar una interfaz muy simple para poder acceder a los servicios provistos por los servidores.

El nivel medio está compuesto por el servidor de agentes, que provee funcionalidad genérica a los mismos. Para ello implementa funciones básicas que le permiten interactuar con los agentes posibilitando, entre otras funciones, su migración, clonación y envío de mensajes. Cuenta además con un administrador de conexiones con servidores remotos y un módulo de comunicaciones encargado de implementar dichas conexiones. Adicionalmente, soporta el agregado de módulos independientes, destinados a implementar las funciones adicionales que se desea probar, en nuestro caso un módulo de soporte de comunicación multicast que en el futuro será complementado por un

módulo de chequeo de red para mejorar la performance del árbol de distribución y un módulo de multicast confiable para ofrecer confiabilidad a la aplicación.

El nivel inferior está constituido por el soporte de comunicaciones provisto por el nivel de transporte de TCP/IP, en particular, por el protocolo TCP, accedido a través de la API de Sockets [Java API, 2004] ofrecida por Java. La estructura del servidor se muestra en la figura 2

3.1 Aplicación

El nivel de aplicación está compuesta por el cliente, que representa la aplicación del usuario (por ejemplo una aplicación de búsqueda de cierta información) y los agentes móviles que éste genera en forma directa, localmente o en servidores remoto, y en forma indirecta, es decir, los posibles clones de dichos agentes. Los servidores de agentes móviles proveen, además de las funciones necesarias para la ejecución, migración y comunicación de los agentes móviles, acceso a agentes estáticos, que permiten a los agentes móviles interactuar con el software que les presta servicios en cada lugar que visitan. Las características de la aplicación y de dichos agentes están fuera del alcance este trabajo.

Los agentes móviles resultan de interés, en particular los aspectos referidos a su movilidad y necesidades de comunicación grupal. Estos pueden implementar cualquier tipo de funcionalidad, respondiendo a diferentes sistemas de agentes móviles reales. Las funciones provistas por el servidor están limitadas a las necesarias para la movilidad de los agentes con el objeto de mantener la simplicidad de la implementación, por lo cual debe considerarse que en algunos casos algunas aplicaciones podrán tener que modificarse, relegando funcionalidad tal como aspectos de seguridad, para correr sobre el mismo. La interfaz provista por el servidor se describe con más detalle en el punto 3.2. Debe remarcar que de acuerdo con el objetivo del trabajo, los aspectos de interés a evaluar son aquellos relacionados con la comunicación grupal entre los agentes, pudiendo omitirse otros aspectos cuando se modifica una aplicación existente o se desarrolla una nueva.

3.2 Servidor

3.2.1 Generalidades

Los servidores de agentes tienen como función principal proveer una plataforma para la ejecución de los agentes móviles. Adicionalmente, soportan y proveen funciones destinadas a su uso por parte módulos que implementan funcionalidad adicional (multicast, etc).

Cada servidor requiere de una máquina virtual java, pudiendo soportarse varios servidores en el mismo host. Un servidor se identifica de manera única a través de la dirección IP del equipo y su puerto de escucha, es decir, aquél donde espera requerimientos de conexión por parte de servidores remotos - por ejemplo, el String: "201.12.98.2:4352" identifica al servidor corriendo en el equipo con la dirección IP 201.12.98.2 atendiendo requerimientos de conexiones en el puerto 4352.

Los servidores son responsables de toda la comunicación, ya sea originada por ellos, por los agentes o por los diferentes módulos. En la figura 2 se indican con líneas punteadas las alternativas de comunicación entre los diferentes elementos que componen el sistema (agentes, módulos y servidores), en todos los casos dicha comunicación (ya sea local o remota) es llevada a cabo con la intermediación de los servidores, a requerimiento de los interlocutores. De esta forma, un agente o módulo residente en el servidor, deberá solicitar a éste el envío de cualquier mensaje a su entidad par correspondiente en un servidor remoto o a otra entidad en el mismo servidor. Si bien esta

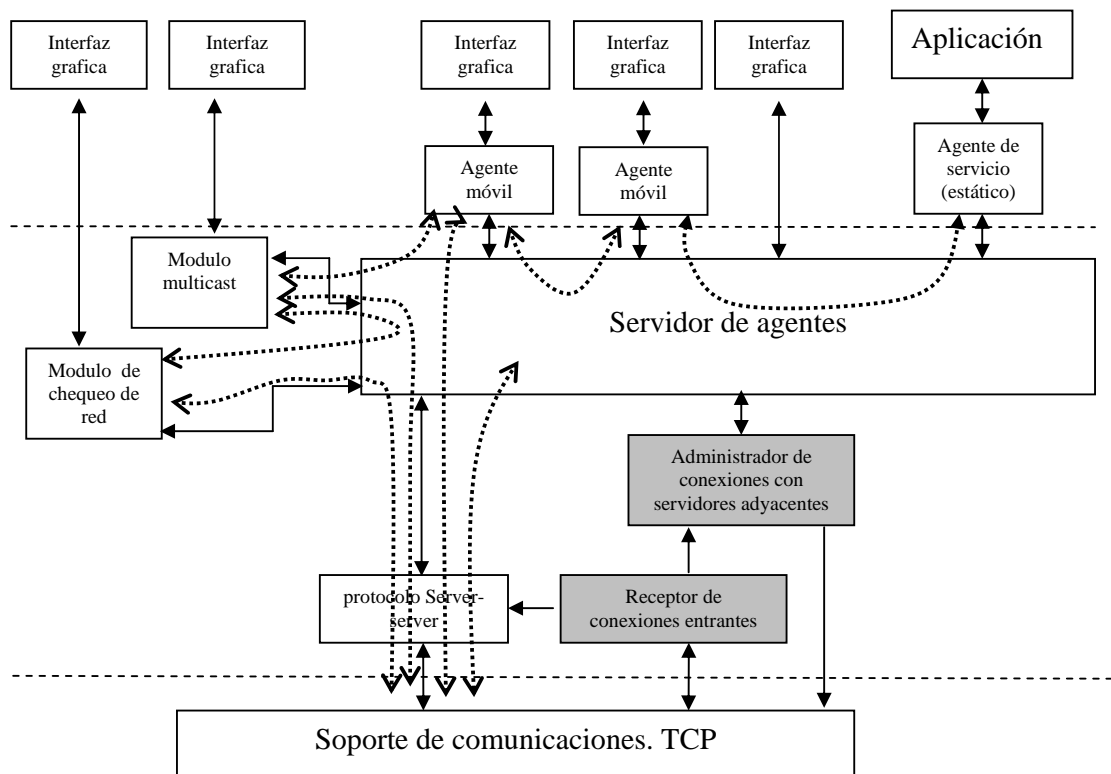


Fig. 2. Estructura del servidor y componentes adicionales.

característica agrega carga en el caso de la comunicación entre módulos remotos, permite el agregado de módulos sin necesidad de definir nuevos hilos de comunicación, ya que esta es realizada a través de la única conexión que mantiene un par de servidores.

La comunicación entre servidores se lleva a cabo a través de conexiones TCP. Un par de servidores estará ligado por una única conexión TCP, que será mantenida solamente mientras sea necesario, por ejemplo, una conexión TCP es establecida para la transferencia de un agente, y es terminada luego de que ésta se completa. Un caso particular lo constituyen las conexiones entre servidores que conforman los vínculos entre nodos del árbol de distribución multicast; éstas, a solicitud del módulo multicast, son mantenidas mientras dicho nodo forme parte del árbol de distribución³.

La elección del protocolo TCP, en lugar de UDP, se debe a que resulta conveniente para los vínculos entre nodos del árbol multicast, ya que además de esperarse que por ellos fluya información de manera regular (en general, las heurísticas de construcción y mantenimiento de los árboles utilizan mensajes heartbeat), permitirá implementar en el futuro más fácilmente la transmisión grupal confiable.

3.2.2 Funciones previstas

³ Existirá una conexión con el nodo padre y una por cada nodo hijo en caso que los hubiera.

El servidor de agentes provee funciones definidas en sus interfaces, a los agentes móviles, a los módulos de apoyo y a la interfaz gráfica utilizada para testeo:

3.2.2.1 Funciones provistas a los agentes móviles

Están definidas en la interfaz `MobileAgentInterface`. Incluyen, funciones básicas para el desenvolvimiento de los agentes móviles, que son provistas por el servidor y funciones adicionales que dependen de la existencia de los módulos que completan la funcionalidad. Desde el punto de vista de los agentes, es el servidor quien las provee, y en caso de solicitar una función no implementada (el módulo correspondiente no se ha registrado con el servidor), éste devuelve el código de error correspondiente. A continuación se describen las funciones más importantes

Migración: permite al agente solicitar su traslado a un nuevo servidor, que debe especificar a través de su identificación única (dirección IP y puerto de escucha de requerimientos de conexión). Debido a la simplicidad buscada, no se realizan chequeos para comprobar autenticidad, ni cifrado de la información. El tipo de migración provisto es débil, pudiendo migrar código y datos, pero no el estado de la ejecución del agente.

Clonación: el agente solicita la creación de una copia de sí mismo (`agentClonate`), que es creada por el servidor y a la cual se le asigna una nueva identificación única de agente.

Comunicación: En principio, la comunicación ofrecida es punto a punto entre agentes, ya sean remotos o locales entre sí. Una alternativa ya implementada es el envío de mensajes a otro agente (`agentSendMessage`), debiendo especificar la identificación única del destino y su servidor; una alternativa a implementar en el futuro, sujeta a disponer de un sistema de localización de agentes⁴, es el envío de mensajes especificando sólo la identificación única del agente destino. En el caso de la existencia de un módulo de soporte multicast, está previsto soportar la comunicación grupal:

3.2.2.2 Funciones provistas a los módulos

Las funciones provistas por el servidor a los módulos son por el momento simples, se espera ir aumentándolas en la medida que la implementación de nuevos módulos lo requiera.

Registro con el servidor: Un módulo debe registrarse (`registerModule`) para ser reconocido por el servidor. Esto posibilita que el servidor obtenga una referencia al módulo y habilite para uso de los agentes las funciones implementadas por él.

Envío de información a otros módulos (`moduleDataRequest`): éstos pueden ser locales o remotos; en ambos casos se debe especificar el servidor donde reside el módulo interlocutor y el tipo de módulo. La comunicación que ofrece el servidor consiste sólo en la transferencia de datos, no hace ninguna interpretación de las PDUs intercambiadas, función que queda a cargo de los módulos.

Información adicional: El servidor provee una lista de los módulos que se hallan registrados (`getModules`).

Comunicación entre diferentes tipos de módulo: esta característica no está implementada por el momento, y se prevé que será soportada a través de interacciones entre cada módulo y el servidor. Es necesaria ya que ciertos módulos requerirán información de otros para cumplir eficientemente sus funciones. Por ejemplo, el módulo multicast deberá ser capaz de solicitar al módulo de chequeo de red información relativa a la topología de la red, para construir de manera eficiente el árbol de distribución.

3.2.2.3 Funciones provistas a la interfaz gráfica

La interfaz grafica, cuyo único objeto es monitorear el estado del servidor y solicitar la ejecución de comandos para comprobar el funcionamiento del sistema, utiliza las siguientes funciones:

⁴ Un sistema de localización con bajo costo adicional podría ser implementado disponiendo de facilidades multicast.

Obtención de información del servidor: entre otras, obtención de los agentes residentes (getAgents), de la identificación del servidor (getAddress), obtención de grupos multicast (getGroups).

Parsing y proceso de comandos al servidor ingresados a través de la interfaz: permite la ejecución de comandos ingresados por el usuario. Estos se encuentran definidos en la interfaz MobileAgentCommandsInterface, e incluyen, entre otras funciones, creación, migración y clonación de agentes, envío de información a otros servidores, manejo de conexiones con otros servidores, etc.

3.3 Módulos

Con el objeto de mantener la simplicidad del servidor aunque se agreguen funciones adicionales de soporte a los agentes móviles, éstas se localizan en módulos independientes del primero. Las funciones provistas de esta manera pueden complementarse unas con otras, pero son independientes entre sí. Un ejemplo lo constituyen la función de construcción y mantenimiento del árbol multicast, localizada en el módulo multicast, y la función de chequeo de red, localizada en el módulo de chequeo, y cuyo objetivo es informar acerca de las condiciones de la red, por ejemplo distancia en saltos entre dos equipos, demora, carga de vínculos, etc. Estos dos módulos son independientes entre sí pero cada uno utiliza funcionalidad provista por el otro para cumplir con su objetivo: el módulo multicast solicita al de chequeo de red métricas para adecuar el árbol de distribución a la topología de la red, mientras que el módulo de chequeo utiliza funciones del módulo multicast cuando debe determinar métricas sobre el árbol de distribución.

Al ser iniciado el servidor, los módulos presentes deben registrarse para ser reconocidos y poder acceder a las funciones necesarias para su operación y para poder ser utilizados por los agentes móviles y eventualmente por otros módulos.

Para mantener la independencia entre los componentes del sistema, se decidió que los módulos no interactúen en forma directa con los agentes móviles ni entre sí, sino que lo hagan íntegramente a través del servidor. Este incorporará en su interfaz funciones adicionales de acuerdo a los servicios provistos por cada módulo.

3.4 Interfaces gráficas

De acuerdo con el objetivo de este trabajo, evaluación de aspectos de soporte a ser provistos a los sistemas de agentes móviles, se prevé la implementación de interfaces gráficas en todos los componentes del mismo, de manera que provean asistencia para depuración en las etapas de desarrollo prueba y evaluación de las diferentes heurísticas a implementar.

Se desarrolló una interfaz gráfica simple en el servidor de agentes. Esta permite monitorear el estado del servidor a través de la consulta de sus variables más relevantes. Es posible además consultar por los módulos de soporte activos en el servidor, por los agentes móviles residentes en él y por el estado de las conexiones con otros servidores. Desde esta interfaz es posible acceder a las interfaces ofrecidas por los dos primeros elementos mencionados, si es que éstos las proveen. Se incluye una interfaz gráfica para los agentes de prueba y una para cada módulo de soporte en desarrollo. Estas son descritas brevemente en las secciones correspondientes.

La interfaz permite además invocar a funciones del servidor, destinadas a comprobar su funcionamiento. Entre los comandos que pueden invocarse se encuentran la conexión y desconexión con otros servidores, creación, clonación, migración y eliminación de agentes, etc. La interfaz gráfica desarrollada para el servidor se muestra en la figura 3.

4. Comunicación entre componentes

Los diferentes elementos que integran el sistema y necesitan comunicarse entre sí son los agentes móviles, los servidores de agentes y los módulos de soporte. Adicionalmente, es necesaria la comunicación entre los agentes móviles y la aplicación cliente que los genera, y de aquellos con

la aplicación que les provee servicios en los lugares que visitan. Estos dos últimos casos son mencionados brevemente ya que no revisten interés para nuestro objetivo. En todos los casos, la comunicación se realiza con intervención del o de los servidores de agentes correspondientes. Estas comunicaciones pueden ser locales si los interlocutores son soportados por el mismo servidor, o remotas, si intervienen diferentes servidores. En este caso se ponen en juego los mecanismos de comunicación entre servidores, basados en TCP (en el caso de dos servidores residentes en el mismo host, se establece una conexión TCP local).

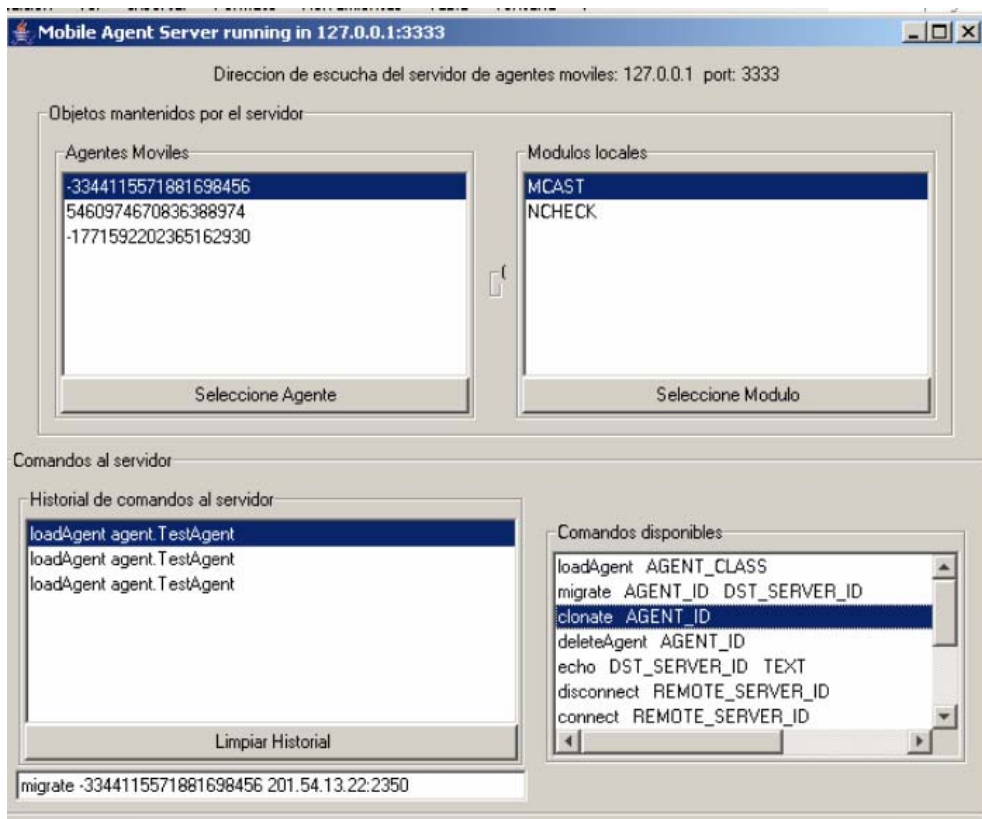


Fig. 3. Interfaz gráfica para depuración provista por el servidor

4.1 Comunicación agente móvil-agente móvil

El objeto es que los agentes puedan intercambiar información entre sí, a efectos de llevar a cabo las funciones de la aplicación en forma colaborativa. Los agentes que se comunican pueden ser locales (residentes en el mismo servidor) o remotos (en distintos servidores) entre sí. Es posible, en la medida que se cuente con el soporte apropiado, realizar comunicaciones grupales. En caso de que se cuente con un sistema de localización de agentes, el destino puede identificarse independientemente de su ubicación en la red. Otro aspecto que debe tenerse en cuenta es la confiabilidad del envío, que debe ser preservada aunque los agentes estén trasladándose en el momento en que se les envía la información (esto tiene consecuencias sobre la comunicación grupal confiable, agregando un nuevo requerimiento – soporte para movilidad - al soporte de multicast confiable tradicional). Los dos últimos aspectos serán abordados una vez concluida la presente etapa de este trabajo. Dicho tratamiento se llevará a cabo a través de la inclusión de los módulos de soporte correspondiente. El protocolo de comunicación entre los agentes móviles es dependiente de la aplicación y no es tratado aquí. La comunicación es soportada en su totalidad por el nivel de

servidores de agentes. Las PDUs generadas son encapsuladas y enviadas a destino cuando el agente lo solicita al servidor.

4.2 Comunicación módulo-módulo

Este tipo de comunicación se produce entre módulos del mismo tipo, situados en diferentes servidores. Su objeto es posibilitar la colaboración entre ellos, para llevar a cabo su función. Por ejemplo, si a un módulo de chequeo de red situado en el servidor S1 se le solicita que determine la demora hasta otro servidor S2, el primero enviará una solicitud de eco a su módulo remoto, para medir el tiempo que se demora en obtener la respuesta. Este tipo de comunicación es más simple que la necesaria para los agentes móviles, ya que los módulos son estáticos y existe uno o ninguno por servidor, no requiriendo funciones de localización del interlocutor ni de almacenamiento temporario del mensaje como en el caso de movilidad. Cada tipo de módulo debe definir un protocolo propio de comunicación. Las PDUs intercambiadas son encapsuladas en PDUs de nivel servidor de agentes, ya que la comunicación módulo-módulo es soportada en su totalidad por el servidor. Debe aclararse que en algunos casos los módulos podrán originar tráfico en la red con autorización pero sin intervención del servidor, por ejemplo, en el caso en que se solicite a un módulo determinar la cantidad de saltos a través de routers que insume llegar de un servidor a otro, dicho módulo utilizará medios proporcionados por niveles inferiores para determinarlo (por ejemplo pings o ICMP echo requests). En general, si al módulo de chequeo de red se le solicitan métricas sobre el árbol de distribución, esto desencadenará comunicación módulo-módulo, soportada por los servidores; en cambio, si se le solicitan métricas sobre la red, deberá recurrir a otros mecanismos que no conciernen al servidor, como los mencionados. El módulo multicast en cambio, no requiere de estos últimos, y la totalidad de su actividad de comunicaciones se produce a través del servidor.

4.3 Comunicación servidor-servidor

Este tipo de comunicación permite el intercambio de información entre los servidores. Como se mencionó en las secciones previas, los servidores soportan las comunicaciones de las demás entidades, transportando sus PDUs encapsuladas como datos en las propias, y sin hacer ningún tipo de interpretación sobre aquellas. Adicionalmente se requiere el intercambio de información de control entre los servidores, para que éstos puedan llevar a cabo sus funciones. A nivel servidor-servidor se ha definido un protocolo muy simple, que en el futuro será mejorado. El soporte utilizado para la comunicación entre servidores es TCP. La elección de este protocolo que provee un servicio orientado a conexión y confiable en lugar de su alternativo UDP, que al proporcionar un servicio no orientado a la conexión, no confiable produce menor carga, se debe a las siguientes razones:

1-Las interacciones entre los servidores consisten generalmente en el intercambio de varias PDUs a nivel transporte (por ejemplo, si se migra un agente, debe trasladarse su código y estado), requiriendo en la mayoría de los casos confirmación a nivel del servidor- Esta característica justifica el establecimiento de una conexión por transacción a efectuarse entre servidores. Un caso donde no sería necesario el uso de TCP (bastaría con UDP) se daría cuando un agente solicita el envío de una cantidad pequeña de caracteres, de modo no confiable a otro agente.

2-El contar con la confiabilidad ofrecida por TCP posibilita que el código del servidor sea mucho más simple, ya que no es necesario incluir funciones para ofrecer confiabilidad en el servidor..

3-Si bien los vínculos establecidos entre servidores son de duración limitada (por ejemplo, si un servidor migra un agente móvil a otro, se establecerá una conexión TCP que será finalizada cuando el servidor de origen haya recibido la confirmación de que el agente fue aceptado), en nuestro caso particular, donde para soportar comunicaciones grupales debe construirse y mantenerse un árbol de distribución, las vinculaciones entre nodos del árbol deben ser mantenidas mientras éstos conserven

la relación de “adyacencia” (padre-hijo), ya que además del intercambio de información grupal producida por la aplicación, fluirán por el árbol de distribución información de control destinada a mantenerlo y mejorarlo, dependiendo de la heurística que se utilice. Esto justifica el mantenimiento de conexiones TCP entre nodos adyacentes en el árbol de distribución⁵

4-El contar con transmisión confiable punto a punto entre los nodos del árbol de distribución multicast hace posible implementar el servicio de multicast confiable a nivel servidores de agentes móviles, sin incurrir en los problemas (aún no resueltos de manera general) relativos a multicast confiable como capa adicional al nivel de red [varias mcast confiable].

La comunicación entre servidores es implementada como se muestra en la figura 3. Cada servidor cuenta con un administrador de conexiones, que es el encargado de llevar un registro de los servidores remotos con los cuales se mantiene una conexión. Los requerimientos de nuevas conexiones y de eliminación de conexiones existentes originados localmente y los requerimientos de conexión o desconexión por parte de servidores remotos son entregados al administrador de conexiones, quien se encarga de impedir el establecimiento de conexiones duplicadas (colisión de requerimientos de conexión entre servidores), y de mantener un registro de las conexiones existentes asociándolas con el servidor correspondiente.

Al comienzo de la ejecución, el servidor de agentes crea un thread de ejecución que se encargará de atender requerimientos de conexión originados por servidores remotos, en el port predefinido (la identificación de un servidor está integrada por la una de las direcciones IP del host más el port de escucha de conexiones entrantes). Este thread se encargará de la creación de un nuevo socket por requerimiento de conexión, entregándolo al administrador de conexiones para su chequeo e inclusión como nueva conexión en la lista de conexiones.

5. Ejemplo, módulo multicast

En esta sección se describe brevemente, a modo de ejemplo, la implementación en curso de una heurística desarrollada previamente [Rigotti, 2004] y probada a través de simulación en el ambiente provisto por el simulador NS (Network Simulator) [REF NS]. Esta implementación permite adquirir experiencia en la incorporación de módulos al medioambiente. En particular, es de utilidad para definir la interfaz a ser provista por el módulo multicast y para determinar las funciones adicionales a proveer por parte del servidor a los agentes móviles para que éstos utilicen facilidades multicast.

El módulo multicast, como el resto de los módulos, permanece oculto a los agentes; sus funciones son accedidas a través de invocaciones al simulador, quien debe ofrecer en su interfaz a los agentes métodos para hacerlo. Estos métodos (por ejemplo la solicitud de agregado a un grupo – join-) devolverán un código de error en caso de que el módulo multicast no se haya registrado y por lo tanto el servidor no tenga manera de proveer tal funcionalidad.

La interfaz ofrecida por el módulo multicast, utilizada por el servidor de agentes, y ofrecida por éste en gran parte a los agentes móviles es independiente de cualquier heurística que se implemente. Las funciones relacionadas se describen a continuación.

1-Agregado de un agente local a un grupo (joinGroup): produce el agregado del agente local al grupo especificado, de manera que cualquier mensaje recibido dirigido al grupo, es entregado a dicho agente. Puede ocasionar, en caso de que el nodo no esté integrando el grupo (es decir, que no sea parte del árbol de distribución) la activación de los mecanismos necesarios para agregarse. Estos son totalmente dependientes de la heurística que se utilice.

⁵ Cuando nos referimos a nodos adyacentes en el contexto de multicast implementado en la red overlay, nos referimos a pares de nodos que se consideran adyacentes de acuerdo con la heurística utilizada. En cambio, en un soporte multicasts a nivel de red, la adyacencia se determina físicamente.

2-Eliminación de un agente local como integrante de un grupo (leaveGroup): produce la baja del agente para el grupo especificado. En los casos en que no existan más agentes locales para el grupo y el nodo no tenga hijos para dicho grupo en el árbol de distribución, dejará de pertenecer a dicho grupo⁶. El mecanismo utilizado para hacerlo depende de la heurística multicast en particular.

3-Solicitud de envío al grupo (send): esta función es invocada por un agente al servidor, quien se encarga de transmitirla al módulo multicast. Este consultará la lista de agentes locales para el grupo, y por cada uno de sus integrantes, excepto aquél que produjo el envío, solicitará al servidor el envío del mensaje correspondiente; de manera similar, chequeará la existencia de nodos hijos en el árbol de distribución del grupo, solicitando al servidor el envío de los datos separadamente a cada uno de ellos, por último, enviará los datos multicast al nodo padre (excepto en caso de ser el nodo raíz). Debe destacarse que como consecuencia de la decisión de mantener la independencia de los módulos con el objeto de permitir su fácil agregado y reemplazo, se producen interacciones servidor-módulo, que en otro caso no serían necesarias: cuando un agente debe enviar un paquete multicast, lo solicita al servidor ya que se desea mantener ocultos los módulos a los agentes; el servidor debe entregar esta solicitud al módulo, quien se encarga, en base a la información multicast que él administra (integración de grupos y relativa al árbol de distribución) de determinar a qué agentes locales y a qué otros módulos multicast en servidores adyacentes debe enviárseles la información, por último, solicita al servidor el envío individual de esta información a cada destino, ya que ni el servidor maneja envíos multicast ni el módulo multicast es capaz de acceder a los agentes locales ni a las conexiones con servidores adyacentes. Consideraciones similares se aplican a la recepción de información multicast descrita a continuación.

4-Recepción de información multicast: la capacidad de recibir información multicast por parte de los agentes móviles no se halla reflejada por una función especial, sino que dicha información será entregada a los agentes por el servidor, quien para ello invocará el método receive de los agentes, utilizado también para la recepción de información unicast. Debido a que los servidores no manejan información multicast, un paquete con destino multicast será entregado al servidor, por el módulo multicast local, como varios paquetes (con dirección de destino multicast) a ser entregado a algunos agentes locales y/o a algunos módulos multicast en servidores adyacentes. Cuando un agente invoca al servidor para realizar el envío multicast, o bien cuando el servidor recibe de uno adyacente un paquete destinado al módulo multicast, éste es notificado y se le entrega la información, luego, el propio módulo indicará al servidor cómo distribuir la misma.

Las funciones más importantes del módulo multicast se refieren a siguientes aspectos

1-Armado y mantenimiento del árbol de distribución: esta función depende enteramente de la heurística a utilizar. El módulo multicast interactúa con módulos pares en otros servidores y con otros módulos locales con la finalidad de seleccionar, de acuerdo a métricas determinadas, los nodos padre e hijo y así formar parte del árbol de distribución.

2-Administración de grupos, agentes y conexiones con servidores adyacentes: esta función no depende de la heurística utilizada en la construcción del árbol multicast. Puede en cambio depender del paradigma multicast utilizado (árboles por emisor, como DVMRP [Waitzman, 1988] o por grupo como PIM [Estrin,1988] en el nivel de red). El módulo multicast debe mantener información relativa a grupos manejados en el equipo (aquellos de los cuales el nodo forma parte); para cada uno de ellos, información respecto a agentes locales pertenecientes al grupo, información sobre el/los árbol/es de distribución y nodos adyacentes en cada caso, y estado del nodo (hoja, nodo intermedio o raíz)

⁶ En el caso de eliminación del agente del grupo por migración del primero, se requerirá intervención de un módulo que provea transmisión confiable (referida a movilidad del agente), el cual podrá diferir la baja del grupo hasta asegurar el join del agente en el nodo de destino de la migración.

3-Distribución de información, esta función consiste en poder aceptar información multicast, ya sea generada localmente o proveniente de un servidor adyacente, y distribuirla a los agentes locales y nodos adyacentes, dependiendo del paradigma multicast utilizado.

En la figura 4 se muestra un esquema simplificado de la interacción entre los agentes móviles, el servidor y el módulo multicast, y el uso de las conexiones con servers adyacentes mantenidas por el servidor. Una solicitud de agregado o eliminación de un agente producirá una interacción con las funciones de mantenimiento de agentes y grupos en el módulo multicast, y posiblemente con la heurística de construcción y mantenimiento del árbol (en caso de un nuevo grupo o salidas del grupo por parte del nodo), mientras que un requerimiento de comunicación involucra a la parte de envío y recepción de información del módulo, la que solicita información respecto del árbol de distribución y de los agentes locales a los otros dos componentes del módulo. Se observa también el uso (y posible solicitud de creación) de conexiones TCP con servidores “adyacentes” desde el punto de vista del módulo multicast.

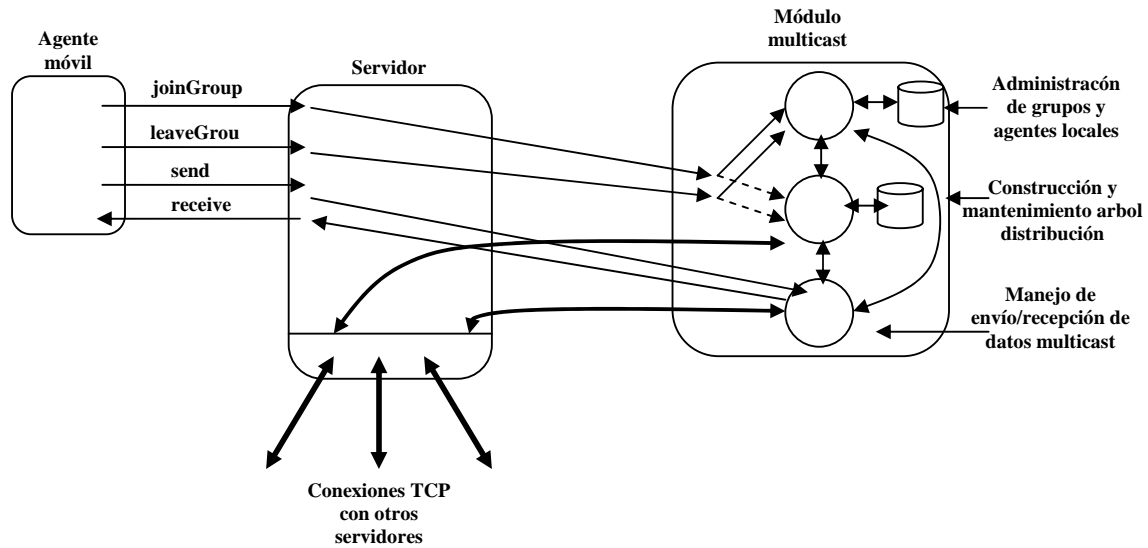


Fig 4. Interacciones entre agentes móviles, servidor y módulo multicast

6. Conclusiones

Se presentó un servidor de agentes móviles en desarrollo cuyo objetivo se limita a servir de banco de pruebas a heurísticas que implementen funciones de soporte dirigidas a los agentes móviles. No es la intención desarrollar un servidor con funcionalidad completa, incorporando aspectos tales como seguridad, etc., propios de cualquier servidor de agentes móviles, sino que se hace énfasis en la facilidad para agregar nueva funcionalidad al soporte implementado. Esta puede abarcar aspectos diversos y específicos en cada caso permitiendo evaluar en un medioambiente real aquellos de interés.

Una de las causas que motivó la realización de este trabajo fue la experiencia previa en simulación de sistemas de agentes móviles (ref) Se comprobó que resulta tedioso adaptar una aplicación al soporte de simulación, ya que éste no permite el uso directo de las facilidades de las APIs Java. Mediante el uso del medioambiente implementado, es posible desarrollar una aplicación simple (o tomar la parte de interés de una aplicación real) y desarrollar el soporte a proveer en

forma de un módulo. Se puede evaluar así, en una primera etapa, el comportamiento de la facilidad provista en un ambiente real (con un número limitado de servidores). En la medida en que resulte de interés comprobar el funcionamiento a gran escala (tanto en la extensión de la red como en el número de agentes y servidores), será necesario, por razones operativas, recurrir a la simulación. En este aspecto, se están evaluando dos alternativas para integrar herramientas de simulación al medioambiente; una de ellas consiste en la adaptación del Java Network Simulator (JNS), mientras que la otra consiste en la inclusión de un nivel de simulación de red totalmente independiente y accedido por cada máquina virtual Java, en lugar de las facilidades de comunicación provistas por el sistema operativo.

En lo inmediato, se probará el sistema desarrollado mediante la incorporación de un módulo multicast que implementará la heurística mencionada anteriormente y de un módulo de chequeo de las condiciones de red, que colaborará con el módulo multicast para que éste pueda ajustar la topología del árbol de distribución a las características de la red física.

7. Bibliografía

- [Castro, 2002] M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron, "Scribe: A large-scale and decentralized application-level multicast infrastructure," IEEE JSAC, vol. 20, no. 8, October 2002.
- [Chu, 2000] Y.-H. Chu, S. G. Rao, and H. Zhang, "A case for end system multicast," in Proc. of ACM Sigmetrics, June 2000.
- [Deering, 1989] Deering, S., "Host Extensions for IP Multicasting", RFC 1112, August 1989.
- [Estrin, 1988] Estrin, D., Farinacci, D. "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification", RFC 2362, June 1998.
- [Java API, 2004] Java Documentation, <http://java.sun.com/j2se>
- [Janotti, 2000] J. Jannotti, D. K. Gifford, K. L. Johnson, F. Kaashoek, and J. W. O'Toole, "Overcast: Reliable Multicasting with an Overlay Network," in Proc. of OSDI, October 2000.
- [JNS] <http://jns.sourceforge.net/>
- [Kermode, 2002] R. Kermode L. Vicisano, "Author Guidelines for Reliable Multicast Transport (RMT) Building Blocks and Protocol Instantiation documents", RFC 3269, IETF, April 2002
- [Postel, 1980] J. Postel, "User Datagram Protocol (UDP) ", RFC 768, IETF, August 1980,
- [Rigotti, 2003] "Infraestructura de comunicación para sistemas de agentes móviles". Rigotti, G, Proceedings AST 2003, 32 JAIIO (Jornadas Argentinas de Informática e Investigación Operativa), Buenos Aires, Argentina, September 2003.
- [Rigotti, 2004] "Un soporte de comunicación grupal para agentes móviles". (Rigotti, G.) 30 Conferencia Latinoamericana de Informática (CLEI 2004). Arequipa, Peru, 27 de septiembre al 1 de octubre de 2004.
- [Savetz, 1998] Savetz, K. "MBONE: Multicast Tomorrow's Internet", 1996 <http://www.savetz.com/mbone/>
- [Waitzman, 1988] D. Waitzman, C. Partridge, S. Deering, "Distance Vector Multicast Routing Protocol (DVMRP)", RFC 1075, IETF, November 1988