

INVESTIGACIÓN

HERMES PLATAFORMA DE DESARROLLO DE AGENTES MOVILES PARA DISPOSITIVOS MOVILESMg. José Luis Jurado M¹

¹ Grupo de Investigación y Desarrollo en Ingeniería de Software, Programa de Ingeniería de Sistemas, Facultad de Ingeniería, Corporación Universitaria Autónoma del Cauca, Popayán, Colombia

Recibido: 30 de mayo de 2010; Revisado: 23 de junio de 2010; Aceptado: 12 de Julio de 2010

Resumen -- Este artículo presenta el proyecto HERMES, un prototipo de una plataforma de agentes para el desarrollo e implementación de aplicaciones con agentes móviles, en dispositivos fijos y móviles. La descripción del proyecto HERMES, que se cita en este documento, detalla cada elemento que conforma la estructura interna de la plataforma HERMES. Se menciona además algunos ejemplos utilizados, para evaluar el correcto desempeño de la plataforma, en distintas situaciones que ponen a prueba la eficiencia de la API desarrollada. El proyecto HERMES, tiene como objetivo primordial proporcionar la migración de agentes móviles, entre equipos fijos y móviles. HERMES brinda además otras opciones adicionales como mensajería y clonación de agentes. Sin olvidar las características básicas que ofrece una plataforma de agentes como: creación, eliminación, registro y posterior gestión de los agentes residentes en la plataforma HERMES. El fundamento teórico del proyecto HERMES, se basa en proyectos similares como Aglets, Jade, Telescript, ARA. etc. La diferencia entre HERMES y los proyectos anteriormente mencionados, radica en que HERMES busca la migración de agentes entre maquinas fijas y móviles, y proyectos como Aglets, ARA y Telescript, solo realizan la migración de sus agentes entre maquinas fijas. En el caso de Jade existen agentes residentes en equipos móviles, pero sin la posibilidad de que migren. [1] HERMES, esta conformado por un conjunto de paquetes agrupados para un propósito específico. Es tarea del desarrollador de aplicaciones darle el mejor uso a cada componente para implementar aplicaciones que presten el mejor servicio a los usuarios finales.

Palabras Clave: Agentes Inteligentes, Plataformas de desarrollo, Dispositivos Móviles

Abstract—This paper presents the HERMES project, a prototype of an agent platform for developing and deploying applications to mobile operators, fixed and mobile devices. The HERMES project description, which is quoted in this document, detailing each element that forms the internal structure of the Hermes platform. It also mentions some examples used to assess the proper performance of the platform, in different situations that test the efficiency of the API developed. The HERMES project, aims primarily to provide the mobile agent migration between fixed and mobile equipment. Hermes also provides additional options such as messaging and cloning of agents. Not to mention the basic features offered a platform for agents such as creating, deleting, recording and subsequent management of resident agents in the HERMES platform. The theoretical foundation of the HERMES project is based on similar projects like Aglets, Jade, Telescript, ARA. etc. The difference between HERMES and the above projects, is that looking HERMES migration of agents between fixed and mobile machines, and projects such as Aglets, ARA and Telescript, only perform the migration of agents among machines fixed. In the case of Jade agents are living in mobile devices, but without the possibility of migrating. [1] HERMES, is comprised of a set of bundled packages for a specific purpose. It is the task of the application developer best to use each component to implement applications that provide the best service to end users.

I. INTRODUCCIÓN

El desarrollo de aplicaciones, servicios y soluciones para dispositivos móviles, es un área de trabajo que ofrece un amplio rango de opciones a los desarrolladores

y clientes, tanto en hardware como en software. El avance en los dispositivos móviles conlleva al aumento en la capacidad y atracción que se ofrece con la movilidad de estos aparatos y sus aplicaciones.

Un factor a tener en cuenta en el avance de las aplicaciones en dispositivos móviles, radica que tanto el cliente como el desarrollador de una solución software, aprovechen al máximo las características de estos dispositivos, con el fin de disminuir los costos y maximizar la efectividad de las soluciones. [2]

Una aplicación que se desarrolle con agentes móviles, se convierte en una herramienta útil para el desarrollo de servicios distribuidos, aportando nuevas capacidades que aun no han sido exploradas y que pueden contribuir valiosamente a estudios a nivel teórico y tecnológico, trazando un camino para la evolución natural que estos sistemas deben experimentar.

Todo agente móvil posee una serie de características como: eficiencia, tolerancia a fallos, personalización, transparencia de localización, interacción de nodos etc. Pero la característica más importante es el nivel de autonomía que poseen, tal propiedad es una característica inherente que reduce significativamente el acoplamiento entre componentes de un servicio. [3]

Otra ventaja que ofrece la autonomía de los agentes y que es aprovechada en el proyecto HERMES, está en las comunicaciones que se realizan a través de mensajes sin necesidad de un contacto muy estrecho entre los agentes. Es decir cada agente puede seguir ejecutando sus funciones mientras realiza un proceso de comunicación. Estas y otras propiedades son las que se buscan explorar y explotar en el desarrollo de servicios de aplicaciones en dispositivos móviles basados en la tecnología de agentes móviles.

Los agentes y su naciente aplicación en un sinnúmero de proyectos de vanguardia a nivel mundial, motivan el desarrollo de una herramienta que permita maximizar y mejorar los servicios en las aplicaciones de los dispositivos móviles y fijos, así como aumentar las posibilidades y funcionalidad del diseño de servicios basados en el paradigma de agentes móviles y que considere a los dispositivos móviles como los terminales que intervienen y acceden a la funcionalidad provista por un sistema de Agentes Móviles

II. DESCRIPCIÓN DEL SISTEMA HERMES

El proyecto HERMES, desarrolló dos versiones de la plataforma. La primera es un prototipo para equipos fijos, esta plataforma se denominó HERMES, funciona bajo la plataforma JSE. Los servicios que presta la versión HERMES para equipos fijos, se basan en las API de la plataforma J2SE, de este modo las tareas de los agentes que se desarrollen bajo la plataforma HERMES, tendrán que diseñarse e implementarse teniendo en cuenta la API de JSE.

La segunda versión que existe de la plataforma HERMES, se denomina HERMES Light, está desarrollada para equipos móviles, su plataforma base es JME, una versión muy reducida de la plataforma JSE, de ahí su nombre. El desempeño de los agentes, que se implementan bajo la versión de HERMES Light, tendrá una menor flexibilidad en cuanto a servicios prestados, ya que su API, es más reducida que la versión de HERMES para fijos.

Arquitectura del Ambiente

La arquitectura del ambiente, vista en la figura 1 muestra la composición en tres niveles de la plataforma HERMES. En el nivel superior, se encuentra la aplicación SMA (Sistema Multi Agente), en este nivel se desarrolla la lógica del negocio de la aplicación, la aplicación puede estar distribuida entre equipos fijos y móviles. La Interacción entre los agentes y la migración de los mismos de un sitio a otro, se desarrollan en este nivel.

En el nivel intermedio se encuentra la API de HERMES, en sus dos versiones (fija y móvil). Las responsabilidades de este nivel de la arquitectura del ambiente de la plataforma HERMES son:

- Ofrecer servicios de localización, migración, comunicación y registro.
- La gestión de los agentes y ambientes de ejecución.
- Ofrecer soporte a los procesos de Interacción y clonación de agentes.

En el nivel inferior, se encuentran las plataformas base de cada versión de HERMES. Ya sea la versión móvil o fija.

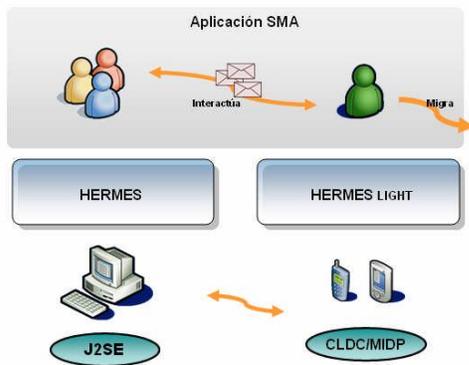


Figura 1. Arquitectura del Ambiente

Composición Interna de los Agentes Hermes

Los agentes de HERMES están compuestos por cuatro elementos básicos (identificador, estado, itinerario, y comportamiento).

La composición interna de un agente determina características tan importantes, como: El estado de un agente a lo largo de su ciclo de vida; los comportamientos y respectivas tareas que le han sido asignadas; su sitio residente y un identificador único, para posteriores procesos de registro y búsqueda.

Identificador. El identificador proporciona un nombre lógico como único valor para diferenciar los agentes y clones creados en la plataforma HERMES. Este identificador es necesario para procesos de registro y localización de los agentes y clones.

Estado. Componente que representa cada uno de los cambios, por los cuales pasa un agente durante su ciclo de vida. Este componente es el encargado de controlar, los cambios ocurridos en el agente y mantener un registro del valor de cada uno de ellos. Su tarea Comienza desde la creación el agente, hasta su muerte.

El agente una vez es creado, tiene un estado inicial, a menos que el desarrollador de aplicaciones determine lo contrario. Este componente tiene una serie de estados como (inicial, en ejecución, arriba y desactivación) que pueden cambiar por la percepción del ambiente y los comportamientos del mismo, hasta llegar a un estado final una vez terminado su ciclo de vida.

Itinerario. Componente encargado de gestionar, controlar y manipular la forma en que se realizaran los viajes de los agentes. Para esto mantiene una lista de lugares (entornos) donde se presume migraran los agentes. El itinerario ofrece mecanismos para manipular los procesos de migración de los agentes. Estos mecanismos permiten controlar el manejo de los sitios donde viajara el agente como el orden y la prioridad para realizar un viaje, entre otros.

El itinerario utiliza las direcciones IP de los agentes, para direccionar y gestionar los sitios donde se realizara el viaje del agente.

Comportamiento: El comportamiento es el componente que facilita agrupar las tareas del agente. Tiene un nombre lógico, que lo define el desarrollador cuando crea un comportamiento determinado. El propósito del comportamiento es adicionar cada una de las tareas creadas para el agente, de modo que las tareas agrupadas en un comportamiento tengan una característica común. Para gestionar el manejo de cada una de las tareas el comportamiento mantiene un listado de las mismas.

Una tarea representa las acciones que el agente va a ejecutar en la aplicación. Cada acción del agente es vista como un procedimiento a nivel de código, un ejemplo sería una consulta a una base de datos, la impresión en pantalla de una operación matemática etc.

El conjunto de tareas y su éxito en la ejecución de las mismas, determinaran que un comportamiento sea ejecutado con éxito. Las tareas son realmente quienes ejecutan las labores del agente, son las que manipulan la información de los agentes y de la aplicación.

Existe dos modos de definir la forma en que se van a ejecutar los comportamientos y más exactamente las tareas. Las tareas pueden hacerse de forma local o remota, de esta manera se deja la decisión al programador de donde se llevara a cabo la ejecución de la tarea.

Ejecución Local: El modo de ejecución local, insinúa que el Agente deberá realizar sus tareas en la maquina residente, es decir en donde el agente fue creado. De esa forma el Agente hará uso de los recursos locales de la maquina donde se esté ejecutando (BD, memoria discos, impresoras, etc.)

Ejecución Remota: Este modo de ejecución, es un poco más complejo, involucra procesos de migración, sincronización y comunicación entre los distintos entornos y Agentes involucrados en el proceso de ejecución remota. El diseñador de aplicaciones sabe que instrucciones, serán ejecutadas en maquinas remotas y con qué fin. De este modo, se evita problemas de incompatibilidad en las API de las plataformas involucradas en el proceso de migración.

IV. ARQUITECTURA DE LA PLATAFORMA HERMES

La Figura 2, muestra la arquitectura HERMES. Este esquema lo componen tres niveles. El entorno, ASH y el sistema de soporte de la plataforma (JSE, JME y el sistema operativo)

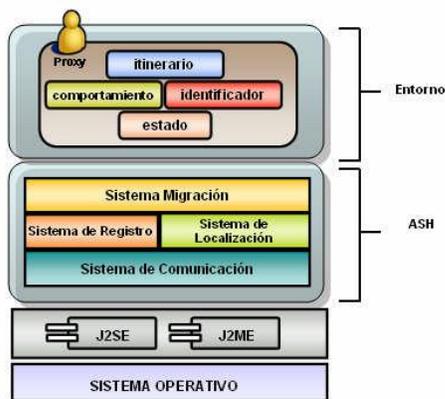


Figura 2. Arquitectura HERMES

Entorno: El componente denominado Entorno que se muestra en la figura 2., lleva este nombre porque cumple con una tarea primaria, agrupar una serie de agentes en un sitio específico, para ofrecer un ambiente de ejecución propicio para los agentes y el desarrollo de su ciclo de vida. El concepto de entorno es ampliamente conocido en las plataformas de agentes, para citar algunos casos está la plataforma JADE, en la cual este tipo de componente es llamado contenedor y cumple propósitos similares que el entorno en HERMES. De igual forma en la plataforma AGLETS, existen los contextos.

La diferencia existente entre las plataformas mencionadas anteriormente y otras plataformas con HERMES, radica específicamente en la estructura interna que ofrece un entorno para los agentes que residen en el. [4]

El entorno tiene la característica de contener en su estructura interna un elemento básico denominado Proxy que representa al agente. En la plataforma HERMES este elemento es esencial para el ciclo de vida de un agente, puesto que colabora como intermediario entre el agente y el SMA para el proceso de comunicación. La creación de un agente, es precedida por la adición de dicho agente a un entorno específico.

De este modo el nuevo agente pasa a ser residente del entorno escogido. Los agentes viajeros, que llegan a una maquina pasan a ser registrados en un entorno escogido previamente en el proceso de migración.

ASH (Ambiente de Servicios HERMES). El ASH es el núcleo de la plataforma HERMES, ofrece servicios de migración, comunicación, registro y localización.

El ambiente de servicios HERMES integra el protocolo de comunicación usado para migrar agentes. Para la transmisión de agentes se hace a través de sockets, la razón se debe a la facilidad que prestan en la implementación del empaquetamiento de información en agentes móviles y la garantía de que la información llega a su destino en el mismo orden que ha sido transmitida.

Sistema de Migración: El sistema de Migración se ocupa del proceso de envío y recepción de los agentes. El proceso de migración no solo involucra envío y recepción de agentes, sino otros procesos mas como marshalling y unmarshalling de la información transferida por los agentes.

La migración consiste básicamente en que el agente encomendado a realizar un viaje determinado, tendrá que recuperar la información necesaria para viajar a otras plataformas y entornos. Esta información debe contener los suficientes datos para restaurar e iniciar la ejecución de una tarea en el sitio escogido para viajar.

Entre las responsabilidades más importantes que tiene el sistema de migración está la realización del proceso de serialización y deserialización. También el sistema de migración se ocupa de despachar y recibir Agentes remotos y locales.

La composición interna del sistema de migración está distribuida en tres componentes (Cliente, Servidor y Serialización). La cooperación e interacción de cada

componente del sistema de migración, permite que la información manipulada sea tratada adecuadamente para ser enviada y recibida de la mejor forma. Obteniendo así los datos necesarios en el momento requerido por el sistema MultiAgente.

Los componentes del sistema de migración son:

Serialización: Este componente se encarga de los procesos de escritura y lectura de la información de cada proceso que se desea enviar a un sitio específico. Además contribuye con el cliente y el servidor en los procesos de marshalling y unmarshalling del agente.

Cliente: Este Componente se ocupa de enviar el agente al servidor que recepcionará la información recibida. Existe un cliente por cada agente viajero, cada cliente utiliza el mismo puerto para el envío de los agentes al servidor.

Servidor: Este componente es el encargado de recibir a los agentes viajeros y volver a reconstruir la información que traen. La información que se reconstruye es necesaria para reanudar la ejecución del agente en la maquina escogida.

El servidor se ocupa del proceso de unmarshalling y el cliente del marshalling. El servidor se mantiene activo durante toda la aplicación y está a la espera de la llegada de un nuevo agente, pero el cliente solo será accedido en el momento de enviar un agente.

Sistema de Registro: El sistema de registro mantiene un directorio de registros, con nombres e identificadores que le permitan distinguir todos los agentes y entornos creados en la plataforma HERMES.

En la plataforma HERMES, Donde además de identificar y registrar los Agentes locales y remotos, controla el registro de posibles agentes maliciosos que lleguen a la plataforma. De este modo se mantiene un mecanismo de seguridad básico para el acceso e interacción de los Agentes. El sistema de registro tiene un prototipo de seguridad que consiste básicamente en el manejo adecuado de la información primaria del agente.

El mecanismo de seguridad que implementa el sistema de registro permite evitar que nuevos entornos y agentes sean creados con nombres lógicos iguales. Impidiendo la ambigüedad de la información y el colapso del sistema multiagente al momento de registrar y localizar agentes y entornos.

La asignación del nombre lógico es tarea del desarrollador y la gestión de este identificador le corresponde al sistema de registro.

Sistema de Localización: Este componente, facilita la localización y búsqueda de entornos y Agentes en las aplicaciones residentes en la plataforma HERMES. Permite acceder a información primaria, de entornos y agentes; facilitando la interacción entre agentes y mejorando el desempeño del comportamiento de un agente.

El proceso de localización se realiza a través del uso de un directorio, que registra la ubicación⁴ donde residen los Agentes. El sistema de localización utiliza información que sustrae del sistema de registro, con el fin de construir un directorio de agentes y entornos que facilite la búsqueda en procesos de registro y comunicación.

Los procesos de búsqueda y ubicación de agentes, clones y entornos en una aplicación residente en HERMES, son transparentes para el desarrollador.

Sistema de Comunicación: El componente encargado de la comunicación en HERMES es el sistema de Comunicación, su propósito es facilitar el proceso de interacción de los Agentes.

El sistema de comunicación proporciona un mecanismo de mensajería entre Agentes remotos y locales. Este mecanismo utiliza mensajes como elemento único para el envío de información entre agentes emisores y receptores. Durante el proceso de interacción entre agentes a través del paso de mensajes, el sistema de comunicación hace uso del nombre lógico de cada agente, para distinguir a quien o quienes es enviado un determinado mensaje.

El Sistema de Comunicación requiere del sistema de localización para encontrar los agentes emisores y receptores de los mensajes enviados. En la plataforma HERMES se ha implementado dos modelos de comunicación necesarios para la interacción⁵ entre agentes y clones.

A continuación se mencionan los modelos implementados en la plataforma HERMES.

- **Unicast:** Un agente emisor envía un mensaje a un agente receptor del mismo entorno. Si el receptor lo requiere responderá al agente emisor, de la misma forma como recibió el mensaje.
- **Broadcast:** Un agente emisor envía un mensaje a un SMA receptor, del mismo entorno. Cada agente del SMA receptor puede responder el mensaje al agente emisor.

Existe una variación de los modelos planteados anteriormente. Esta variación se da cuando ocurre un envío de un mensaje a un solo agente emisor y este puede replicar dicho mensaje a otro agente emisor o a un SMA.

La comunicación entre agentes y clones solo es posible entre inquilinos de un mismo entorno. Si se desea establecer una comunicación con otro agente perteneciente a otro entorno, será necesario migrar a ese entorno.

V. CONCLUSIONES

El objetivo principal en el momento de plantearse el proyecto HERMES, fue desarrollar una plataforma que soportara la migración de agentes entre dispositivos móviles a fijos y viceversa.

Se obtuvieron resultados satisfactorios en la transferencia de agentes entre dispositivos fijos y móviles y en el desarrollo de una API que brindara herramientas al desarrollador para implementar agentes y ambientes de ejecución propicios para ejecutar dichos agentes. Además se alcanzó el desarrollo de mecanismos de comunicación entre agentes y clones implementados con la API de HERMES.

A medida que se avanzaba en el desarrollo del proyecto, surgieron diversas dificultades que se fueron solventando satisfactoriamente, para citar un solo caso, el problema de la incompatibilidad de las API de JME y JSE. Problema que se solventó teniendo dos versiones de cada plataforma.

Otro problema encontrado a lo largo del desarrollo del proyecto, fue el inconveniente en la ejecución de procedimientos en máquinas distintas con API incompatibles; problema que se solventó con el manejo del concepto de tareas remotas y locales.

Se obtuvo un producto que cumple con los requerimientos mínimos de un prototipo de una plataforma de agentes. El proyecto HERMES, proporciona al desarrollador de aplicaciones, una herramienta para crear y manipular agentes, con el fin de migrar procedimientos y agentes hacia dispositivos fijos y móviles.

VI. REFERENCIAS

- [1] Gian Pietro Picco, Mobile agents: an introduction, *Microprocessors and Microsystems* 25 (2001), 65-4.
- [2]] Antunes, P., Acosta, C. Handheld CSCW in the Meeting Environment. *CRIWG 2002, LNCS 24440*, pp.47-60. (2003)
- [3]] M^a Celeste Campo Vázquez, Departamento de Ingeniería Telemática Universidad Carlos III de Madrid, Avda. Universidad 30 28911 Leganés, Madrid España
- [4] William R. Cockayne and Michael Zyda. *Mobile Agents*, Manning Publications Co.1997

José Luis Jurado M, Ingeniero de Sistemas – Universidad del Cauca, Magister en Ciencias de la Computación Universidad del Cauca, Experiencia en Ingeniería de Software, Coordinador del Grupo de Investigación GIDISA y miembro del grupo IDIS – Universidad del Cauca. Docente Investigador Corporación Universitaria Autónoma del Cauca.