


INVOCACIÓN DE MÉTODOS REMOTOS (RMI)

BY OSCAR YIMA 2012

JAVA RMI

Es un mecanismo ofrecido por Java para invocar un método de manera remota. Forma parte del entorno estándar de ejecución de Java y provee de un mecanismo simple para la comunicación de servidores en aplicaciones distribuidas basadas exclusivamente en Java.



OBJETIVO

El objetivo principal de RMI es la de permitir a los programadores el desarrollo de programas distribuidos en JAVA, usando la misma sintaxis y semántica de los programas no distribuidos.



JAVA RMI Y SU CONEXIÓN

RMI como extensión de Java, es una tecnología de programación, fue diseñada para resolver problemas escribiendo y organizando código ejecutable. Así RMI constituye un punto específico en el espacio de las tecnologías de programación junto con C, C++, Smalltalk, etc.




DIAGRAMA GENERAL.

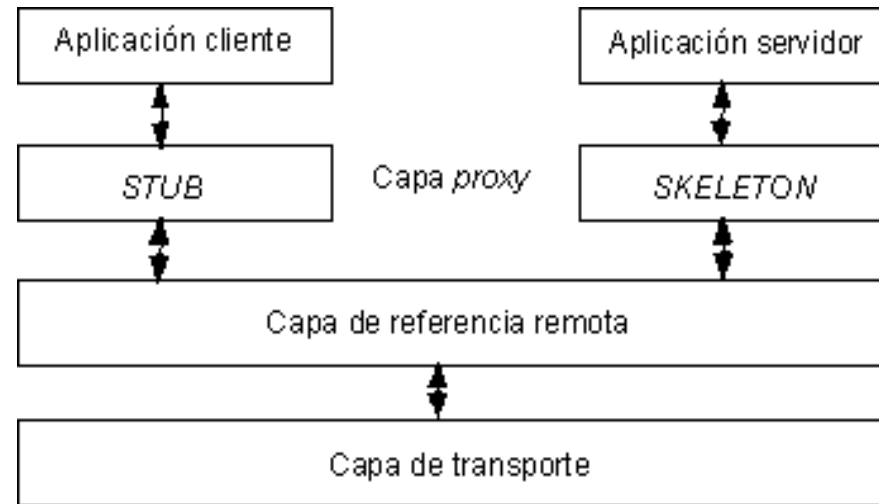


Diagrama general se puede ver como un modelo de 4 capas.

CAPAS

La **primera** capa es la de aplicación y se corresponde con la implementación real de las aplicaciones cliente y servidor

La **segunda** capa es la capa proxy, o capa stub-skeleton. Esta capa es la que interactúa directamente con la capa de aplicación. Todas las llamadas a objetos remotos y acciones junto con sus parámetros y retorno de objetos tienen lugar en esta capa.

La **tercera** capa es la de referencia remota, y es responsable del manejo de la parte semántica de las invocaciones remotas.

La **cuarta** capa es la de transporte. Es la responsable de realizar las conexiones necesarias y manejo del transporte de los datos de una máquina a otra. El protocolo de transporte subyacente para RMI es JRMP (*Java Remote Method Protocol*), que solamente es "comprendido" por programas Java.



CARACTERÍSTICAS

- **RMI se caracteriza por la facilidad de su uso en la programación por estar específicamente diseñado para Java;**
- **Proporciona paso de objetos por referencia (no permitido por SOAP),**
- **Recolección de basura distribuida (Garbage Collector distribuido) y paso de tipos arbitrarios (funcionalidad no provista por CORBA).**
- **Por medio de RMI, un programa Java puede exportar un objeto, lo que significa que éste queda accesible a través de la red y el programa permanece a la espera de peticiones en un puerto TCP. A partir de este momento, un cliente puede conectarse e invocar los métodos proporcionados por el objeto.**

LIMITACIONES

RMI tiene algunas limitaciones debido a su estrecha integración con Java, la principal de ellas es que esta tecnología no permite la interacción con aplicaciones escritas en otro lenguaje.



LA INVOCACION SE COMPONE DE

- **Encapsulado (marshalling) de los parámetros (utilizando la funcionalidad de serialización de Java).**
- **Invocación del método (del cliente sobre el servidor). El invocador se queda esperando una respuesta.**
- **Al terminar la ejecución, el servidor serializa el valor de retorno (si lo hay) y lo envía al cliente.**
- **El código cliente recibe la respuesta y continúa como si la invocación hubiera sido local.**

FIN

