

1

Comprender qué es un servicio en la red.

2


Aprender a planificar un servicio en la red.

3


Recordar el uso de *Socket* para la implementación de servicios en la red.

4

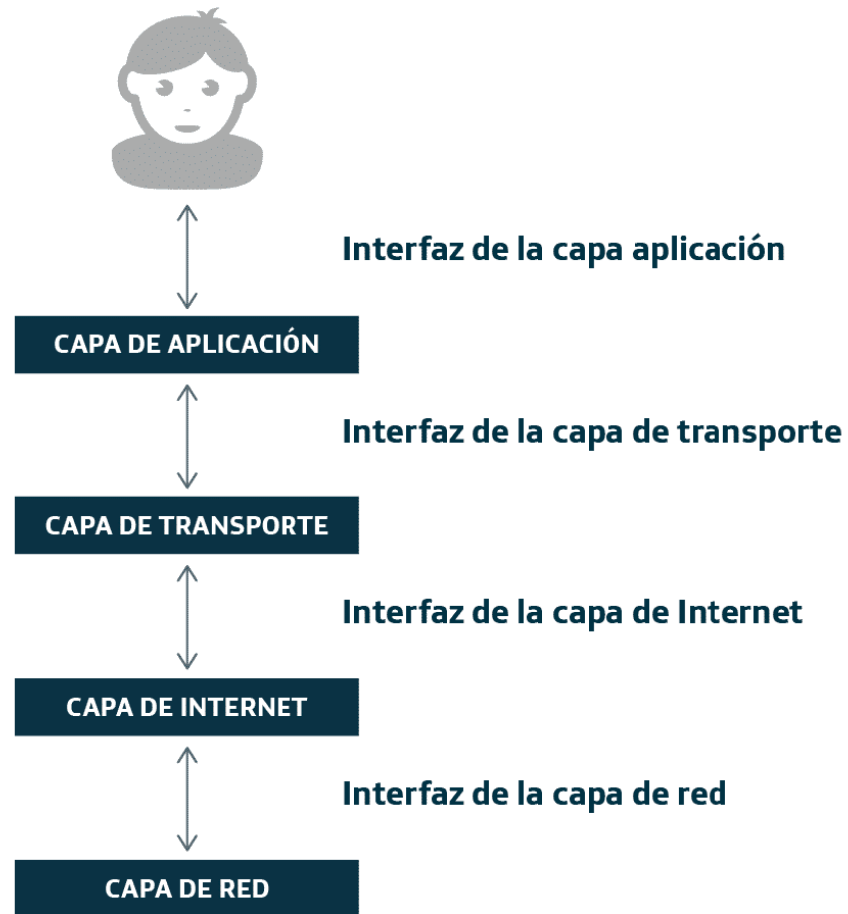
Aprender técnicas avanzadas para la implementación de servicios en la red (RMI).



Un servicio es un programa que se ejecuta en segundo plano y que no es utilizado directamente por el usuario, sino por otros programas que invocan a las funciones que dicho servicio deja disponibles a través de su interfaz.



Cada capa dentro de la pila de protocolos IP proporciona un servicio a la capa de nivel superior y una interfaz, a través de la cual puede comunicarse con ella.



Pila de protocolos IP

Servicios en la capa de aplicación

- Como futuro programador de servicios en la red, la capa que más debe preocuparte es la **capa de aplicación**. Para el resto de las capas contarás con diversos recursos que te permitirán olvidarte de ellas.
- La capa de aplicación debe contar con una interfaz para que el usuario pueda activar el servicio, desactivarlo, con gurararlo, etc.

Con nuestros conocimientos sobre servicios en la red podemos llegar a la conclusión de que la planificación para el desarrollo de un nuevo servicio requiere plantearse las siguientes preguntas:

1

¿Qué servicios debe prestar nuestro servidor?

2

¿Qué formato tiene la petición del cliente?

3

¿Qué formato tiene la respuesta del servidor?

4

¿Se resuelve con una sola petición/respuestas, o requiere de varios mensajes entre cliente y servidor?

5

¿Es necesario que nuestro servidor atienda varias peticiones simultáneas? Si la respuesta es afirmativa, será necesario el uso de hilos.

Desarrollo de una aplicación cliente/servidor para el cambio de divisas

1

¿Qué servicios debe prestar nuestro servidor? Se trata de un servidor que realiza el cálculo de cambio de divisa.

2

¿Qué formato tiene la petición del cliente? El mensaje de la petición del cliente estará formado por los siguientes elementos: una cantidad, divisa de dicha cantidad, divisa para el calculo del cambio. Ejemplo: *100;E;D* (donde E=Euros y D=Dólares).

3

¿Qué formato tiene la respuesta del servidor? La respuesta del servidor será una línea de texto similar a esta: *100 Euros = 115,64 Dólares*.

4

¿Se resuelve con una sola petición/respuestas, o requiere de varios mensajes entre cliente y servidor? Un mismo cliente podrá solicitar varios cambios de divisa, por lo tanto, harán falta varios mensajes entre cliente y servidor.

5

¿Es necesario que nuestro servidor atienda varias peticiones simultáneas? Sí, ya que varios clientes solicitar cambios de divisa al mismo tiempo.

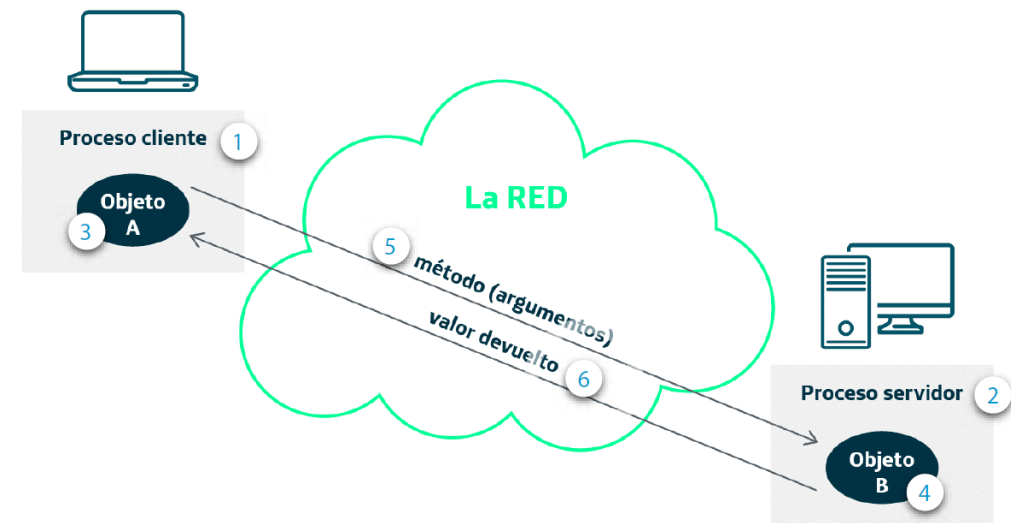
Invocación de métodos remotos (RMI)

Hasta ahora hemos conocido los fundamentos de la programación distribuida.

Pero son demasiado laboriosos para desarrollar aplicaciones distribuidas complejas.

Para eso necesitaremos técnicas de más alto nivel.

La primera técnica que estudiaremos es la **invocación a métodos remotos**, cuya abreviatura es RMI (*Remote Method Invocation*).



1. Proceso cliente.

Programa en ejecución que consume los servicios que provee el servidor, invocando a los métodos remotos que este pone a disposición.

2. Proceso servidor.

Programa en ejecución que comparte objetos cuyos métodos pueden ser invocados de manera remota.

3. Objeto A.

Objeto en el proceso cliente que se encarga de invocar a los métodos remotos que el servidor pone a disposición. Para ello, realizará una petición a través de la red.

4. Objeto servidor.

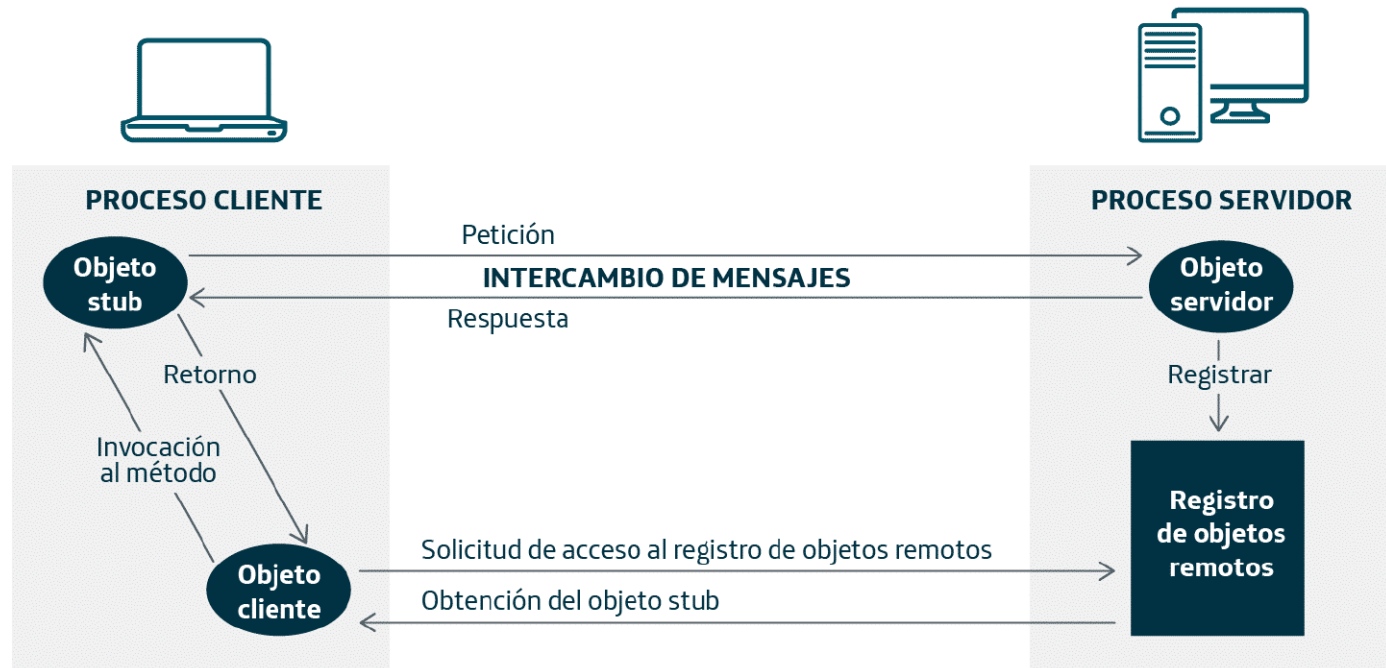
Se trata de un objeto localizado en el proceso servidor que tiene como objeto prestar algún servicio remoto. Sus métodos serán invocados de manera remota por parte del objeto cliente.

5. Método invocado.

Cuando un objeto cliente invoca a un método remoto, dicho método, junto con sus argumentos, debe ser convertido a mensaje para poder ser transmitido a través de la red al objeto servidor a modo de petición.

6. Valor devuelto.

Una vez que el objeto servidor ha terminado de ejecutar el método invocado, el resultado (valor devuelto) será convertido a mensaje para ser enviado a través de la red al proceso cliente.



- En toda aplicación distribuida desarrollada con RMI hay que resaltar dos componentes fundamentales:
- **Los Stubs:** son objetos localizados en el proceso cliente que sustituyen a los objetos remotos. Un objeto *stub* tiene sus métodos como un reflejo de los métodos del objeto servidor, pero cuando recibe la invocación a un método, no lo ejecuta directamente, sino que envía un mensaje al servidor para que ejecute dicho método y envíe el resultado, luego recoge dicho resultado y lo devuelve como si lo hubiera ejecutado él mismo.
- **El registro de objetos remotos:** controla todos los objetos remotos que presta el servidor y que conforman los servicios que este brinda dentro del sistema distribuido. El proceso servidor deberá registrar cada objeto remoto que forme parte del servicio. Por otra parte, el cliente deberá localizar el registro y buscar en él el objeto deseado.

El **proceso servidor** debe efectuar las siguientes tareas:

- 1 **Crear el registro de objetos remotos** asociado a un determinado puerto.
- 2 **Crear los objetos que forman parte del servicio prestado.** Estos objetos se crean de manera habitual.
- 3 **Registrar (inscribir) los objetos creados** en el registro de objetos remotos, dejando así iniciado el servicio.

El **proceso cliente** efectúa las siguientes tareas:

1

Solicitar al servidor el *stub* que sirva de espejo para comunicarse con el objeto servidor. Esto se solicita al registro de objetos remotos del servidor a través de la dirección IP y puerto donde está localizado.

2

Invocar a los métodos del *stub* y recibir los resultados de dichos métodos.

3

El *stub* intercambiará mensajes con el servidor para solicitar la ejecución de los métodos y obtener los resultados.

Ejemplo práctico RMI: el programa servidor

Desarrollaremos una aplicación distribuida con RMI. El programa servidor funcionará como un buscador de canciones, utilizando como clave de búsqueda el título de la canción, la banda, el álbum o el año de producción.