

## 1. RPC

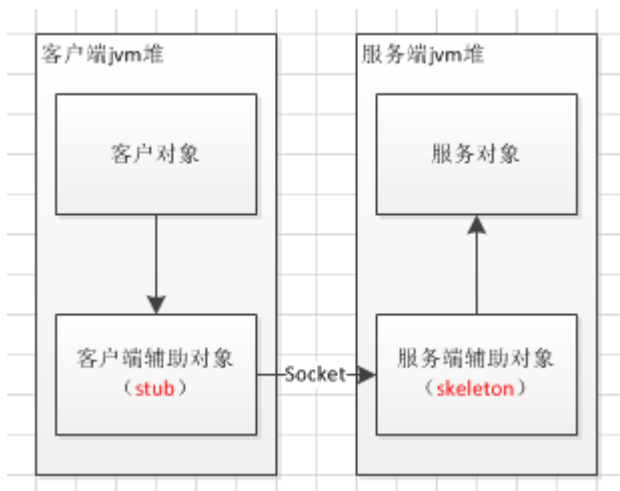
RPC (Remote Procedure Call Protocol) 远程过程调用协议，通过网络从远程计算机上请求调用某种服务。

一次RPC调用的过程大概有10步：

1. 执行客户端调用语句，传送参数
2. 调用本地系统发送网络消息
3. 消息传送到远程主机
4. 服务器得到消息并取得参数
5. 根据调用请求以及参数执行远程过程（服务）
6. 执行过程完毕，将结果返回服务器句柄
7. 服务器句柄返回结果，调用远程主机的系统网络服务发送结果
8. 消息传回本地主机
9. 客户端句柄由本地主机的网络服务接收消息
10. 客户端接收到调用语句返回的结果数据

## 2. RMI

RMI: 远程方法调用 (Remote Method Invocation)。能够让在客户端Java虚拟机上的对象像调用本地对象一样调用服务端java 虚拟机中的对象上的方法。



RMI远程调用步骤：

- 1, 客户调用客户端辅助对象stub上的方法
- 2, 客户端辅助对象stub打包调用信息（变量，方法名），通过网络发送给服务端辅助对象skeleton
- 3, 服务端辅助对象skeleton将客户端辅助对象发送来的信息解包，找出真正被调用的方法以及该方法所在对象
- 4, 调用真正服务对象上的真正方法，并将结果返回给服务端辅助对象skeleton

- 5, 服务端辅助对象将结果打包, 发送给客户端辅助对象stub
- 6, 客户端辅助对象将返回值解包, 返回给调用者
- 7, 客户获得返回值

### 3. 区别

#### a. 方法调用方式不同:

RMI中是通过在客户端的Stub对象作为远程接口进行远程方法的调用。每个远程方法都具有方法签名。如果一个方法在服务器上执行, 但是没有相匹配的签名被添加到这个远程接口(stub)上, 那么这个新方法就不能被RMI客户方所调用。

RPC中是通过网络服务协议向远程主机发送请求, 请求包含了一个参数集和一个文本值, 通常形成“classname.methodname(参数集)”的形式。RPC远程主机就去搜索与之相匹配的类和方法, 找到后就执行方法并把结果编码, 通过网络协议发回。

#### b: 适用语言范围不同:

RMI只用于Java;

RPC是网络服务协议, 与操作系统和语言无关。

#### c: 调用结果的返回形式不同:

Java是面向对象的, 所以RMI的调用结果可以是对象类型或者基本数据类型;

RMI的结果统一由外部数据表示(External Data Representation, XDR)语言表示, 这种语言抽象了字节序类和数据类型结构之间的差异。