

RMI规范--第三章 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/145/2021_2022_RMI_E8_A7_84_E8_8C_83-_c104_145025.htm 主题：Stub与skeleton 远程方法调用中的线程使用 远程对象的垃圾收集 动态类的加载 通过代理服务器透过防火墙的RMI 3.1Stub与skeleton 在与远程对象的通信过程中，RMI将使用标准机制（用于RPC系统）：stub与skeleton。远程对象的stub担当远程对象的客户本地代表或代理人角色。调用程序将调用本地stub的方法，而本地stub将负责执行对远程对象的方法调用。在RMI中，远程对象的stub与该远程对象所实现的远程接口集相同。调用stub的方法时，将执行下列操作：初始化与包含远程对象的远程虚拟机的连接。对远程虚拟机参数的进行编组（写入并传输）等待方法调用结果 解编（读取）返回值或返回的异常 将值返给调用程序 为向调用程序展示比较简单的调用机制，stub将参数的序列化和网络级通信隐藏了起来。在远程虚拟机中，每个远程对象都可以有相应的skeleton（纯JDK1.2环境中不需要skeleton）。skeleton负责将调用分配给实际的远程对象实现。它在接收入进入方法调用时执行下列操作：解编（读取）远程方法的参数 调用实际远程对象实现上的方法 将结果（返回值或异常）编组（写入并传输）给调用程序 由于推出于JDK1.2及附加的stub协议，使得在纯JDK1.2环境中无需使用skeleton。相反，应使用通用代码代替JDK1.1中的skeleton履行其职责。stub和skeleton由rmic编译器生成。3.2远程方法调用中的线程使用 RMI运行时分配给远程对象实现的方法可能在也可能不在独立的线程中执行。RMI运行时将无法担保远

程对象与线程的映射关系。因为同一个远程对象的远程方法调用可能会同时执行，所以远程对象实现需确保其实现是线程安全的。

3.3 远程对象的垃圾收集

与在本地系统中相同，在分布式系统中自动删除那些不再被任何客户机引用的远程对象是令人满意的。这可以将程序员从跟踪远程对象客户机以便适时终止的任务中解脱出来。RMI使用与Modula-3网络对象相似的引用计数的垃圾收集算法（参见1994年5月数字设备公司系统研究中心技术报告115中Birrell、Nelson和Owicki的“网络对象”）。要实现引用计数垃圾收集，RMI运行时需要跟踪每个Java虚拟机内的所有活动引用。当活动引用进入Java虚拟机时，其引用计数将加1。首次引用某对象时会向该对象的服务器发送“referenced”消息。当发现活动引用在本地虚拟机中并未被引用时，该数将减1。放弃最后的引用时，未被引用的消息将被发送到服务器。协议中存在很多微妙之处，其中大部分都与维护引用或未引用消息的次序有关，可确保对象不被过早地收集。当某远程对象不被任何客户机所引用时，RMI运行时将对其进行弱引用。如果不存在该对象的其它本地引用，则弱引用将允许Java虚拟机的垃圾收集器放弃该对象。通过保持对对象的常规引用或弱引用，分布式垃圾收集算法可与本地Java虚拟机的垃圾收集器以常规方式进行交互。只要存在对远程对象的本地引用，就不能将远程对象当作垃圾进行收集，而且该远程对象也可在远程调用中传送或返回客户机。传递远程对象也将同时把目标虚拟机的标识符添加到被引用集中。需要未引用通知的远程对象必须实现java.rmi.server.Unreferenced接口。当这些引用不再存在时，将调用unreferenced方法。当发现引用集为空时，也将调

用unreferenced。因此，unreferenced方法可能会被多次调用。只有当没有本地和远程引用时，才可收集远程对象。注意，如果在客户机和远程服务器对象之间存在网络分区，则可能会过早地收集、远程对象（因为传输可能认为客户机已失效）。由于可能会出现过早收集的现象，因此远程引用将不能保证引用的完整性。换句话说，远程引用实际上可能指向不存在的对象。使用此类引用时将抛出必须由应用程序处理的RemoteException。

3.4 动态类加载

RMI允许传入RMI调用中的参数、返回值和异常为任何可序列化对象。RMI使用对象序列化机制将数据从一个虚拟机传输到另一个虚拟机，同时用相应的位置信息注释调用流，以便在接收端上加载类定义文件。当解编远程方法调用的参数和返回值以使之成为接收虚拟机中的有效对象时，流中所有类型的对象都需要类定义。解编进程将首先尝试通过本地类加载上下文（当前线程的上下文类加载器）中的名称来解析类。RMI也提供动态加载作为参数和返回值传送的实际对象类型的类定义的手段（其中远程方法调用的参数和返回值来自传送终点所指定的网络位置）。这包括远程stub类的动态下载-该类对应于特定远程对象实现类（用于包含远程引用）及RMI调用中通过值传送的任何其它类型，例如在解编端的类加载上下文中尚不可用的，声明参数类型的子类。要支持动态类加载，RMI运行时应使用用于编组、解编RMI参数和返回值的编组流的特定java.io.ObjectOutputStream和java.io.ObjectInputStream子类。这些子类覆盖了ObjectOutputStream的annotateClass方法和ObjectInputStream的resolveClass方法，以便就何处定位包含对应于流中类描述符的类定义的文件交换信息。 100Test 下

载频道开通，各类考试题目直接下载。详细请访问
www.100test.com