



A Programmer's Guide to Jini™ Technology

Jini 技术开发指南

Jan Newmarch 著
林琪 译

- * Jini专家撰写，全面涵盖了Jini的功能
- * 解决了诸如应用体系结构、Jini服务的用户界面以及如何将硬件设备和CORBA纳入Jini框架等一系列重要问题
- * 由资深Java程序员和著名专栏作家Bill Venner完成技术审校

Apress软件开发精品图书系列



A Programmer's Guide
to Jini™ Technology

Jini 技术开发指南

Jan Newmarch 著
林琪 译

中国电力出版社

内 容 提 要

Jini 是最近出现的构建分布式应用的框架之一。它由 Sun Microsystems 创建, 采用 Java 编写, 并基于 Java 所支持的分布式面向对象原则实现应用的分布与组织。本书涵盖了所有的 Jini 概念, 并对一些高级问题进行了讨论, 如连接 Jini 与 CORBA 系统以及利用 Jini 使硬件设备在网络上可用, 等等。这是一本拿来即用, 学完后能够马上进行编程的书。

本书适合有一定 Java 和网络编程经验而且希望尽快掌握 Jini 的程序员和设计人员阅读。

图书在版编目 (CIP) 数据

Jini 技术开发指南/(美) 纽玛茨著; 林琪译. —北京:
中国电力出版社, 2002
ISBN 7-5083-1283-X

I. J... II. ①纽...②林... III. 分布型网络-软件工具
IV. TP393.03

中国版本图书馆 CIP 数据核字 (2002) 第 064450 号

著作权合同登记号 图字: 01-2002-2008 号

本书英文版原名: A Programmer's Guide to Jini™ Technology
Published by arrangement with APRESS

中国电力出版社出版、发行

(北京三里河路 6 号 100044 <http://www.infopower.com.cn>)

北京市地矿印刷厂印刷

各地新华书店经售

*

2003 年 1 月第一版 2003 年 1 月北京第一次印刷

787 毫米×1092 毫米 16 开本 23.5 印张 529 千字

定价 48.00 元

版 权 所 有 翻 印 必 究

(本书如有印装质量问题, 我社发行部负责退换)

前言

无论是商业领域还是学术界，长期以来都已接受了网络技术的应用，他们允许用户共享软件和应用，以及使用诸如电子邮件等网络服务来交换信息。**Internet** 的飞速发展使得每个人都清楚地认识到网络应用的重要性，同时也意识到随着家庭网络和移动网络的出现，这种重要性也将飞速增长。

对于开发人员来讲，构建分布式应用可能是一件复杂的事情。除了要应用划分为多个部分，使之不仅可以独立运行而且可以连接为更大规模的功能单元外，还需要考虑网络稳定性和可访问性等相关问题。由此出现了一大批各式各样的框架，其中既包括试验性的又包括商业性的，其目的都是为了使构建和部署分布式应用更为容易。

Jini 是最近出现的构建分布式应用的框架之一。它由 **Sun Microsystems** 创建，尽管建立在以往的经验基础上，但为了适应当今的面向对象世界，也引入了一些新的概念。**Jini** 采用 **Java** 编写，并基于 **Java** 所支持的分布式面向对象原则实现应用的分布与组织。它允许开发人员建立带有分布式垃圾回收机制的类型安全应用，这样得到的应用不仅可以应对网络失败，还可以根据需要发现和使用分布式服务。

这本书适合对于 **Java** 和网络编程有一定工作经验而且希望尽快掌握 **Jini** 的程序员和开发人员阅读。在此假设你对诸如远程过程调用等网络概念相当了解，而且熟悉 **Java** 语法，并对 **Java** 核心类有一定的使用经验。

这是一本拿来即用，学完后能够马上进行编程的书。我的想法是为你介绍一些代码，它们不仅易于理解，而且可以拷贝到你自己的程序中使用。本书涵盖了所有的 **Jini** 概念，并对一些高级问题进行了讨论，如连接 **Jini** 和 **CORBA** 系统以及使用 **Jini** 使硬件设备在网络上可用等等。这本书以各种形式出现在 **Internet** 上已近两年，不仅受益于用户的反馈，同时还得到了许多新 **Jini** 程序员的帮助。

前 8 章介绍了 **Jini** 程序设计的基本问题，并建立了一个完整但简单的应用。后面的章节则讨论了更为高级的内容，如事件处理、安全性、事务和激活等等，此外还包括了 **Jini 1.1** 新加入的辅助类。另外，我还对其他 **Jini** 书中很少涉足的一些问题做了总结，如用户界面、与诸如 **CORBA** 等其他分布式系统的连接，以及利用 **Jini** 使用硬件设备等等。

这本书基于 2000 年末发布的 **Jini1.1** 版本，但代码用到了 **JDK 1.2** 和 **JDK 1.3**。

文件

本书中的程序源码可通过解压一个压缩文件 **programs.zip** 获得。已编译的类则可从另一个压缩文件 **classes.zip** 得到。这些文件均放在网站 <http://pandonia.canberra.edu.au/java/jini/tutorial/Jini.xml> 上。

其他资源

- ◆ Jini FAQ: <http://www.artima.com/jini/faq.html>
- ◆ Jini 社区网站: <http://www.jini.org/>
- ◆ Jini 主页: <http://www.sun.com/jini/index.html>
- ◆ RMI 主页: <http://java.sun.com/products/jdk/rmi/index.html>
- ◆ Jini 邮件列表: jini-user@java.sun.com
- ◆ Jini 邮件列表归档: <http://archives.java.sun.com/archives/jini-users.html>
- ◆ RMI 邮件列表: rmi-users@java.sun.com
- ◆ RMI 邮件列表归档: <http://archives.java.sun.com/archives/rmi-users.html>
- ◆ 标准化服务 Jini 接口库 (目前为空): <http://www.artima.com/jini/interrepo/>

致谢

本书作者向以下人员致以诚挚的感谢, 感谢他们对本书所做的评论:

Brian Jeltema, jeltema@richdist.east.sun.com

Roger Whitney, whitney@cs.sdsu.edu

Robbert van den Beld, rbe@mms-dresden.de

Chitrarasu Muthaiyan, chitrarasu@cswl.com

Stuart Remphrey, Stuart.Remphrey@Aus.Sun.COM

JJ Larrea, jjlarrea@redtop.com

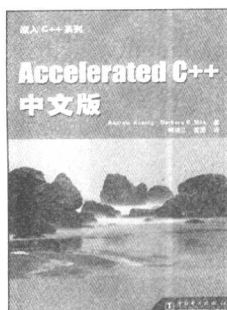
John McClain, John.McClain@East.Sun.COM

Bob Scheifler, rws@east.sun.com

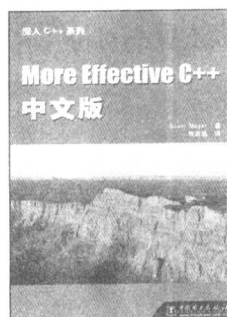
这本书的大部分工作都是作者在开发一个旨在研究分布式系统技术的“度假”程序时所完成的, 这个程序属于 CRC 项目 (<http://www.dstc.edu.au>), 而且本书中所介绍的一些工作由澳大利亚联邦政府工业、科学和旅游部门的合作研究中心项目提供了部分资助。

C++ 之父 Bjarne Stroustrup 主编

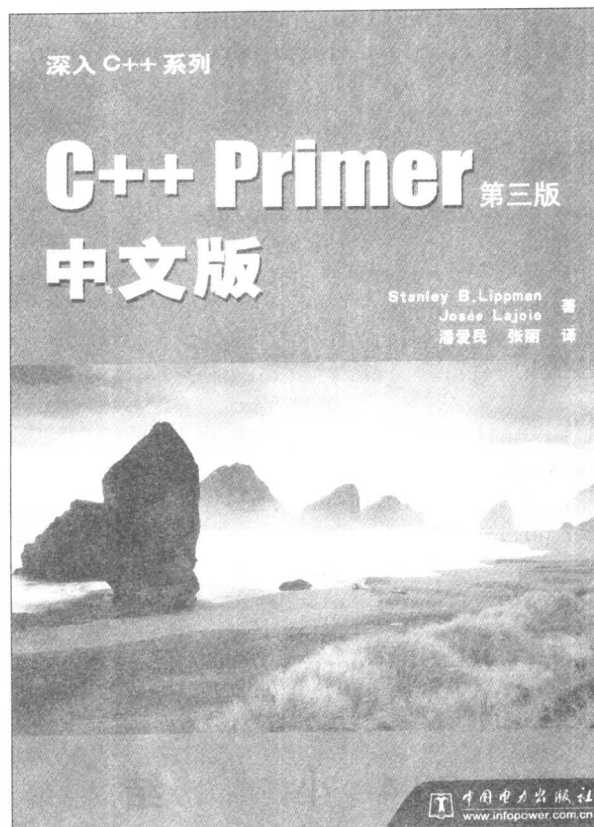
深入 C++ 系列



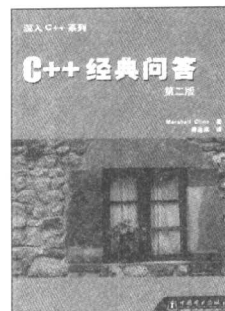
- * C++ 最佳入门书籍
- * 快速掌握 C++ 的全新方法
- * 单刀直入 C++ 核心部分



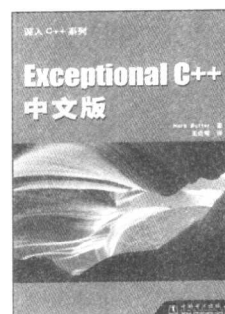
- * 影响深远的经典著作
- * 35 条专家经验
- * 条条精彩



- * C++ 最佳教程
- * C++ 最佳参考书
- * 名著名译
- * 相得益彰



- * comp.lang.c++. 精华荟萃
- * C++ 程序员进阶必备



- * 著名网络专栏 Guru of the Week 的精华荟萃
- * 47 条专家经验
- * 条条精彩



- * 斯坦福大学教授的呕心之作
- * 与《Art of Computer Programming》齐名的算法巨著



- * STL 之父权威之作



- * 最具权威性的 STL 书籍
- * 侯捷评语：“另辟蹊径”

经典著作
不容错过

循序渐进
可收宏效

目 录

前 言

第 1 章 Jini 概述	1
1.1 Jini	1
1.2 组件	2
1.3 服务注册	3
1.4 客户查找	4
1.5 代理	6
1.6 客户结构	7
1.7 服务器结构	8
1.8 应用划分	9
1.9 支持服务	11
1.10 小结	13
第 2 章 Jini 配置问题疑难解答	14
2.1 Java 包	14
2.2 Jini 版本	15
2.3 Jini 包	16
2.4 查找服务	17
2.5 RMI 存根	17
2.6 调试	18
2.7 小结	19
第 3 章 发现查找服务	20
3.1 运行查找服务	21
3.2 单播发现	23
3.3 广播发现	28
3.4 ServiceRegistrar	34
3.5 小结	36

第 4 章 Entry 对象.....	37
4.1 Entry 类	37
4.2 Entry 的限制	39
4.3 辅助类	40
4.4 Entry 的进一步应用	41
4.5 小结	41
第 5 章 服务注册	42
5.1 ServiceRegistrar.....	42
5.2 ServiceItem.....	43
5.3 注册	44
5.4 ServiceRegistration.....	44
5.5 SimpleService 程序.....	45
5.6 Entry	47
5.7 小结	48
第 6 章 客户搜索	49
6.1 使用 ServiceRegistrar 搜索服务.....	49
6.2 接收 ServiceMatches 对象.....	52
6.3 匹配服务	52
6.4 小结	53
第 7 章 租用	54
7.1 请求与接收租用	55
7.2 续租租用	56
7.3 租用的授予和处理	57
7.4 小结	69
第 8 章 一个简单示例	70
8.1 问题描述	71
8.2 服务说明	73
8.3 通用类	74
8.4 客户	76

8.5	服务代理	83
8.6	上载一个完整的服务	83
8.7	小结	90
第 9 章	服务体系结构的选择	91
9.1	代理的选择	92
9.2	用于 FileClassifier 的 RMI 代理	96
9.3	用于 FileClassifier 的非 RMI 代理	103
9.4	用于 FileClassifier 的 RMI 和非 RMI 代理	111
9.5	使用其他服务	117
9.6	小结	127
第 10 章	发现管理	128
10.1	寻找查找定位器	128
10.2	LookupLocatorDiscovery	130
10.3	LookupDiscoveryManager	132
10.4	小结	133
第 11 章	加入管理器	134
11.1	Jini 1.1 JoinManager	134
11.2	Jini 1.0 JoinManager	136
11.3	小结	139
第 12 章	安全性	140
12.1	从没有安全开始	141
12.2	为什么 AllPermission 不好	141
12.3	取消 AllPermission	143
12.4	Jini 保护	144
12.5	服务需求	145
12.6	客户需求	146
12.7	RMI 参数	148
12.8	ServiceRegistrar	149
12.9	事务管理和其他可激活服务	149
12.10	rmid	152

12.11	极端安全	154
12.12	小结	159
第 13 章	复杂示例	160
13.1	类文件的位置	161
13.2	运行发现线程	170
13.3	不完全服务匹配	175
13.4	使用本地服务实现匹配	178
13.5	只找到一次服务	185
13.6	服务修改的租用	188
13.7	小结	194
第 14 章	远程事件	195
14.1	事件模型	195
14.2	远程事件	196
14.3	事件注册	198
14.4	监听者列表	198
14.5	监听者源	201
14.6	利用事件的文件分类器	202
14.7	监视服务中的变化	207
14.8	小结	212
第 15 章	ServiceDiscoveryManager	213
15.1	ServiceDiscoveryManager 接口	213
15.2	ServiceItemFilter 接口	214
15.3	立即寻找服务	215
15.4	使用过滤器	217
15.5	建立服务的高速缓存	219
15.6	利用 Cache 监视变化	223
15.7	小结	226
第 16 章	事务	227
16.1	事务标识符	228
16.2	TransactionManager	228

16.3	TransactionParticipant	229
16.4	Mahalo	229
16.5	一个事务示例	230
16.6	小结	247
 第 17 章 LEGO MINDSTORMS		248
17.1	将硬件纳入 Jini 服务	248
17.2	MindStorms	249
17.3	将 MindStorms 实现为 Jini 服务.....	249
17.4	RCXPort	250
17.5	RCX 程序	251
17.6	Jini 类	253
17.7	完成运行	259
17.8	机器人的 Entry 对象	266
17.9	客户端 RCX 类	266
17.10	更高级的机制: Not Quite C.....	267
17.11	小结	271
 第 18 章 CORBA 与 Jini.....		272
18.1	CORBA	273
18.2	将 CORBA 映射为 Java	275
18.3	Jini 代理	276
18.4	一个简单 CORBA 示例	277
18.5	房间预订示例	284
18.6	将 CORBA 客户引入 Jini.....	297
18.7	将 Jini 服务实现为 CORBA 服务.....	298
18.8	小结	298
 第 19 章 Jini 服务的用户界面.....		299
19.1	将用户界面实现为 Entry	300
19.2	工厂对象的用户界面	301
19.3	现有的工厂	302
19.4	工厂编组	302
19.5	UIDescriptor	303

19.6	文件分类的用户界面示例	306
19.7	图像	314
19.8	ServiceType	315
19.9	MindStorms UI 示例	315
19.10	小结	331
 第 20 章 激活机制		332
20.1	采用激活机制的服务	333
20.2	LeaseRenewalService	348
20.3	LookupDiscoveryService	356
20.4	小结	365

Jini 概 述

内 容

- Jini
- 组件
- 服务注册
- 客户查找
- 代理
- 客户结构
- 服务器结构
- 应用划分
- 支持服务
 - ◆ HTTP 服务器
 - ◆ RMI 守护进程
- 小结

Jini 是利用 Java 构建分布式系统的中间件。Jini 的基础在于 Socket 和远程方法调用 (Remote Method Invocation, RMI) 的分布式计算机制。其根本目的就是提供“网络即插即用”，从而使新服务可以加入到其他服务网络，并立即可用，而客户则可以查找并使用这些服务。写这本书的时候，距 Jini 发布不过一年多一点的时间，它为构建分布式系统引入了全新的思想和技术。本章将对 Jini 系统中的组件及它们之间的关系作概要性的介绍。

1.1 Jini

Jini 可算是能够提供“网络即插即用”的分布式计算环境的代名词。在这种环境中，设备或软件服务可加入网络并声明其存在，而需要此服务的客户则可以找到并调用服务来完成任务。Jini 能够用于完成移动计算任务（这里，服务通常只能短时间联网），不过对于存在某种程度变化的网络而言，Jini 有着更广泛的应用。例如，下列情形就可以体现出 Jini 的作用。

- 一台新的打印机联网并对其位置和功能进行声明后，客户无需作任何额外的配置工作就可以使此打印机。
- 数码相机联网后，通过其提供的用户界面，不仅可以摄像，还可以了解到打印机的情况，从而将摄制的图像打印出来。
- 从各个主机上拷贝的配置文件，作适当的修改就可以成为来自单机的网络服务，从而减少维护带来的开销。
- 对现有功能进行了扩充后，可以将改进后的功能直接加在正在运行的系统中，而无需中断现有服务，也无需对客户进行重新配置。
- 服务可以声明状态的改变，例如，打印机缺纸，等等，监听者（一般是管理员）则会监视状态变化并作出标志以示提醒。

Jini 并不是一个缩写词，也没有什么特殊的含义（不过即然 Jini 并非字首缩写，对 Jini 就有了一种有趣的解释“Jini Is Not Initials”）。Jini 系统，也称为联邦或联盟，是指客户与服务的集合，它们之间通过 Jini 协议进行通信。一般来讲，Jini 系统中包含用 Java 写的應用，而应用之间通过 Java 远程方法调用（RMI）机制进行通信。尽管 Jini 是用纯 Java 开发的，但对客户以及服务而言并没有强制要求只能用纯 Java 来开发。它们可以包含本地代码方法，这时，它们就相当于非 Java 对象的包装器，甚至还可以用其他语言直接编写。对此 Jini 提供了一个“中间件”层以连接不同来源的服务和客户。

1.2 组 件

需要指出的是 Jini 只是大量分布式系统体系结构中的一种，而目前在工业领域广泛应用的 CORBA 和 DCOM 也属分布式系统体系结构。与其他结构不同的是，Jini 是基于 Java 的，因此有很多性质是从 Java 基础派生的。后面部分章节中将讨论怎样在 Jini 和 CORBA 之间搭桥，以此说明如何连接不同的分布式体系结构。

除了 Jini 以外，Sun 公司还推出了其他 Java 框架，如 EJB（企业 Java Bean）。利用 EJB 可以很方便地建立业务逻辑服务器，而 Jini 更适用于将这些服务以“网络即插即用”的方式进行分布。

注意，Jini 只是这个绝不冷清的市场中参与竞争的一个产品，其成功与否部分取决于市场运作，更关键的还取决于其技术功能本身（就 Jini 技术的先进性而言，其成功是大有希望的），本书将讨论应用 Jini 的一些技术问题。

在一个运行着的 Jini 系统中，有三个主要部分，首先是服务，如打印机、烤箱或婚介所；再有则是客户，需要以上服务的就是客户；最后是查找服务（服务定位器），它相当于服务与客户之间的中介/Trader/定位器（译注 1）。要连接这三方，还需要另外一个组件——网络。网络通常运行在 TCP/IP 协议基础之上（Jini 规范应该是与网络协议无关的，但目前仅实现了对 TCP/IP 的支持）。见图 1.1。

译注 1：本书中多次出现了“查找服务”、“服务定位器”、“查找定位器”，还有“定位器”为了尊重原文，没有对此进行统一，但这几个词按作者原义是相同的。

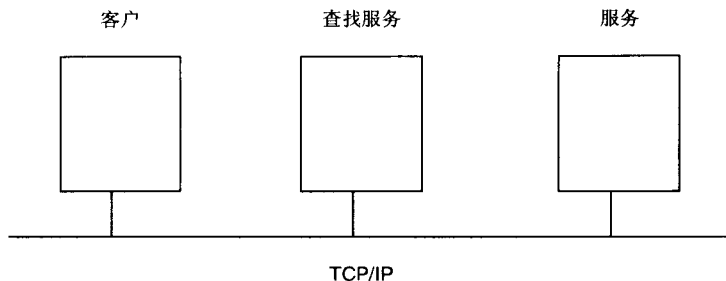


图 1.1 Jini 系统的组件

通过对象编组，代码可以在这三方之间迁移。具体来讲，需要以某种形式对对象进行串行化，使之能够在网络上移动，再以一种“缩水”的形式加以存储，然后再利用实例数据以及所包括的有关类文件的信息实现对象重构。可以利用 Java 的 Socket 对发送和接收对象的支持来实现网络迁移。

此外，在一个 Java 虚拟机（Java Virtual Machine, JVM）中的对象可能需要调用另一个 JVM 中对象的方法，通常这是通过 RMI 完成的，对于 Jini 而言，RMI 并不是必需的，Jini 规范还提供了可以实现这一目的的其他方法。

1.3 服务注册

服务是一个逻辑概念，可以是任何东西，比如一种聊天服务或一个磁盘，甚至可以是一个搅拌器。通常，服务都是由 Java 接口定义的，而且接口也用于通告服务。另外，接口还可以用于定位服务。每种服务可以由不同的开发商用不同的方法实现。例如，Joe 可以有他的约会服务，Mary 也可以有她的约会服务，同理，其他人同样也有自己的约会服务。这些服务的共同点就在于它们实现相同的接口，而它们之间的区别则在于不同的实现中所使用的属于不同类的不同对象集（也可能只有一个对象）。

服务由服务提供者（Provider）创建，服务提供者主要完成以下工作：

- 创建实现服务的对象。
- 在查找服务中注册某个服务对象，此服务对象是服务中公开可见的部分，并且可以由客户下载。
- 以服务器的身份实现其他任务，例如保证该服务有效。

服务提供者为了在查找服务中完成服务对象的注册，需要先找到查找服务本身，可以有两种实现方法：如果查找服务的位置是已知的，服务提供者就可以使用单播 TCP 直接与之联系；如果位置并不确定，服务提供者则需要先组播 UDP 请求，查找服务将响应此请求。查找服务的监听端口为 4160 端口，对单播和组播请求的监听都在此端口上完成。[4160 是十六进制数（CAFE-BABE）的十进制表示]。此过程如图 1.2 所示。

当查找服务在此端口上监听到一个请求，它将向服务器回送一个对象，如图 1.3 所示。此对象称为注册对象，在服务的 JVM 中执行，其作用就相当于查找服务的代理。服务提供

者向查找服务发出的所有请求都要经过这个作为代理的注册对象处理。要实现这个目的，可以选用的协议很多，但实际上你所能得到的查找服务的实现（例如 Sun 提供的实现）都是采用 RMI。

服务提供者通过注册对象所做的工作就是向查找服务登记注册其提供的服务，具体包括复制服务对象和在查找服务中保存它。如图 1.4 所示。

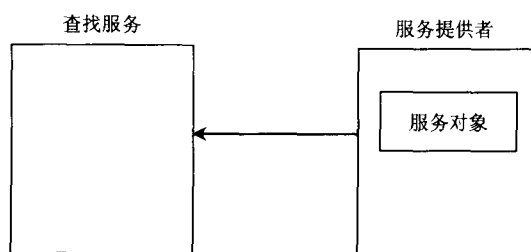


图 1.2 请求服务定位器

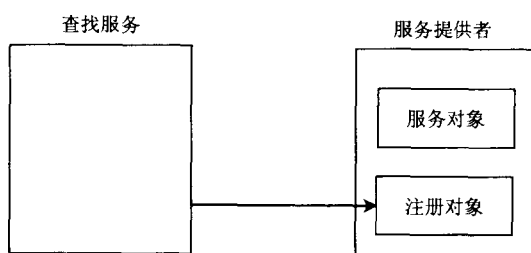


图 1.3 返回注册对象

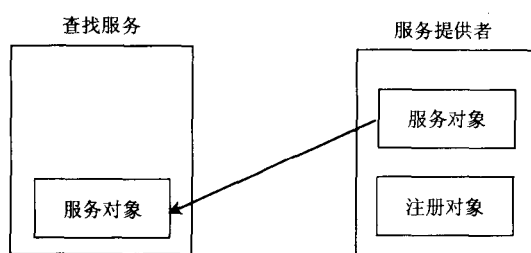


图 1.4 服务上载

1.4 客 户 查 找

另一方面，客户也要获得服务的拷贝，并放入自己的 JVM 中。它通过同样的机制在查找服务上获得注册对象，如图 1.5 和图 1.6 所示。

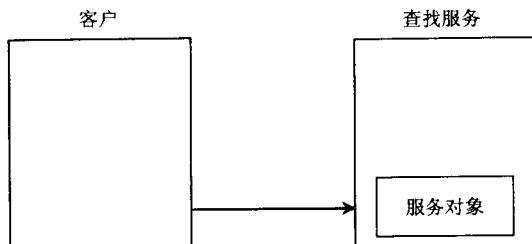


图 1.5 请求服务定位器

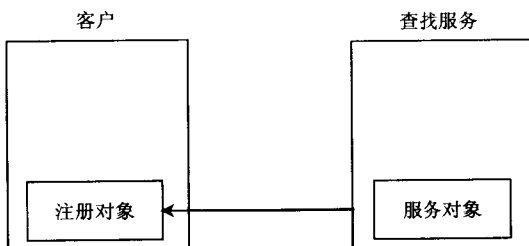


图 1.6 返回注册对象

不过客户利用注册对象完成的工作与服务器有所不同，它请求将服务对象进行拷贝。如图 1.7 和图 1.8 所示。

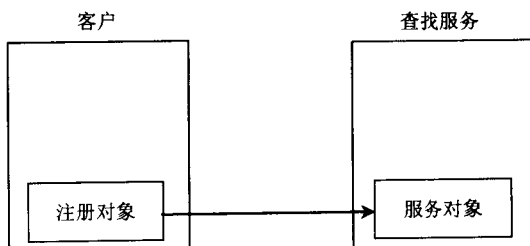


图 1.7 请求服务

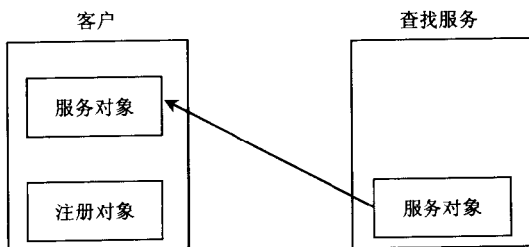


图 1.8 返回服务对象