

文章编号: 1005-0523(1999)01-0050-06

# 基于 Java 的远程围棋对弈软件的设计

丁振凡

(华东交通大学 现代教育技术中心, 江西 南昌 330013)

**摘要:** 介绍了用 Java 语言实现的远程围棋对弈软件的设计, 该软件由服务方和客户方两部分组成, 任意两个 TCP/IP 连接的机器间通过分别运行服务方和客户方的 Java 应用程序进行对弈, 如将服务方在 Web 服务器上运行, 可实现浏览器与 Web 服务器间的对弈<sup>19</sup>。

**关键词:** Java; 围棋; 客户/服务器; Socket 通信; 多线程

**中图分类号:** TP 393.2      **文献标识码:** A

## 0 引言

Java 的出现带来了一场计算观念上的革命, 使得原来以“台式系统为中心”的计算转向“以网络为中心”的分布计算<sup>19</sup>。Java 是一种简单、面向对象、分布式、解释型、健壮、安全、体系结构中立、可移植、高性能、多线程、和动态等多种特性的语言<sup>19</sup>。Java 的跨平台、丰富的网络类库、Java 的多线程、与 WWW 的紧密结合、再加上 JDBC 对数据库的访问等支持使 Java 成为编写 CSCW 协作应用的理想工具<sup>19</sup>。以往两人异地对弈围棋只能靠传真棋谱来实现, 不能很好满足实时要求, 且费用开销高<sup>19</sup>。Internet 网将全世界连在一起, 编写一个基于 Internet 的围棋远程对弈程序有一定的实际意义<sup>19</sup>。

## 1 两人对弈的基本实现思路

在网络环境下进行两人对弈, 一方面要实现基于事件的下棋操作界面, 另一方面要实现双方的通信, 保证两端棋盘内容的一致性<sup>19</sup>。

具体实现上笔者采用了 Java 的多线程特性<sup>19</sup>。用 Socket 通信实现信息交换, 如图 1<sup>19</sup>。其中, 下棋线程实现与用户的交互, 完成下棋动作, 并将本方的下棋消息发送到对方; 通信线程负责接收对方的消息, 并将其反映在本方棋盘上<sup>19</sup>。按 Socket 通信的客户/服务器要求, 下棋双方分为服务方和客户方, 双方在能通

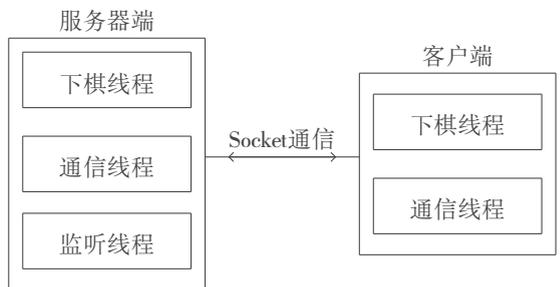


图1 对弈的基本思路

信前要建立 Socket 连接<sup>19</sup>为支持多用户同时与服务方对弈,在服务方另设有一个监听线程,对于每个客户连接申请,在服务方分别要建一个新的窗口、新的下棋线程及通信线程<sup>19</sup>。

## 2 类设计

Java 是一种纯面向对象的编程语言,所有 Java 程序代码都是通过类实现,通过类的继承实现软件重用,Java 提供了丰富的类供用户继承<sup>19</sup>其中,awt 包提供了大量的控件、容器及布局供用户进行跨平台的界面设计;net 包提供了强大的网络支持,为基于网络的应用设计提供了方便;现代流行的多线程编程在 Java 中直接支持,Java 语言有两种方式实现多线程:一种是直接继承 Thread 类;一种是实现 Runnable 接口<sup>19</sup>这两种方法实质上没有什么区别,但 Java 规定任何类只能继承一个类,当某类需要继承其它类同时又要作为线程时,只能使用 Runnable 方式,下面就本软件中编写的类作简要说明<sup>19</sup>。

### 2.1 服务器方

1) CHESS 类(棋盘面板):继承 Panel 类且通过实现 Runnable 接口实现多线程,该类实现下棋操作界面,含下棋光标的处理、下子、提子、让子、以及对方下子显示等方法<sup>19</sup>。

2) CHOICE 类(下棋辅助面板):继承 Panel 类,实现认输、放弃一手、与对方交谈等辅助功能<sup>19</sup>。

3) youchess 类(应用窗口):继承 Frame 类,实现服务方的下棋窗口,将 CHESS 类与 CHOICE 类的实例布局安排在此窗口中<sup>19</sup>。该类含 main() 方法<sup>19</sup>。

4) CONNECTS 类(监听线程):继承 Thread 类,用于处理与客户端的连接<sup>19</sup>。

5) COMMS 类(通信线程):继承 Thread 类,负责接收来自客户方的下棋消息并对消息分析处理<sup>19</sup>。

6) WARNING 类(消息警告):继承 Dialog 类,用于显示下棋消息,提醒用户<sup>19</sup>。

7) TALK 类(谈话窗口):继承 Dialog 类,用于录入和发送谈话内容<sup>19</sup>。

### 2.2 客户方

客户方与服务方在下棋界面设计上与服务方有很多一致之处,以下各类在客户方与服务方完全一样,它们是 CHESS 类、WARNING 类、TALK 类、CHOICE 类,另外客户方还有如下一些类:

1) ENTER 类(对弈选择面板):有让子选择、黑白选择、更改颜色、开始下棋等控件<sup>19</sup>。

2) mychess 类(应用窗口):继承 Applet 类,同时实现 main() 方法,以支持在浏览器和命令窗口两种环境下使用<sup>19</sup>。

3) COMMC 类(通信线程):继承 Thread 类,用于接收服务器方的消息<sup>19</sup>。

## 3 应用界面设计

### 3.1 选择约定界面

只在客户方有此界面,用于规定正式对弈前双方的一些约定<sup>19</sup>。其中含如下项目:

1) 选择黑白棋选项组,两个 CheckBox 控件分别对应黑白选项,缺省是客户方执黑、服务

器方执白<sup>19</sup>.

- 2) 选择让子下拉框,有一个 Choice 控件可选让 0 至 9 个子,缺省是让 0 个子<sup>19</sup>.
- 3) 开始下棋按钮,按下此按钮将进入对弈操作画面,并将选择约定通知服务器方<sup>19</sup>.

### 3.2 对弈操作界面

此界面服务器方与客户方一致,在窗口中有两个面板,一块面板为棋盘面板,用于实现正常的下棋动作;另一个面板是辅助面板,实现认输、交谈、放弃一手等辅助动作<sup>19</sup>.

#### 3.2.1 棋盘面板

有一个红色小方块用于指示下棋位置, mouseMove 事件实现小方块的移动, mouseUp 事件用于在当前位置下棋子<sup>19</sup>.在下棋操作处理上有以下技术要点:

- 1) 避免闪烁<sup>19</sup>.

Java 对屏幕显示的更新是通过调用 repaint( ) 方法,它将调用 update(g) 方法,缺省的 update(g) 方法是先清屏再调用 paint(g) 方法,清屏会使屏幕显示有明显的闪烁<sup>19</sup>.笔者对 update(g) 方法重写,让它不清屏,分情况调用如下方法对屏幕更改: cursor(g) 方法实现重画光标; removechess(g) 进行提子处理;而 paint(g) 方法根据棋的内容对整个棋盘重画<sup>19</sup>.

- 2) 围棋规则的考虑<sup>19</sup>.

用一个二维数组表示棋盘上棋子,每个元素可能三个值:1 表示黑棋,2 表示白棋,0 表示无子<sup>19</sup>.用一个变量表示当前下棋方,每下一步棋要检查与该新下子相邻的棋块的气数,当气数为 0 时要进行提子处理<sup>19</sup>与新子邻接的最多有四个棋子(当处于棋盘边界位置时更少),为考虑这些相邻子是否要被提,实际要计算这些邻子所在棋块的气数是否为 0<sup>19</sup>.在计算棋块的气时引入了一些辅助数组来存放搜索的相关棋子以实现快速计算,具体算法限于篇幅略<sup>19</sup>.在提子个数为 1 时,要作打劫判定处理,程序中使用打劫位置登记和引入逻辑变量来解决打劫状态处理问题<sup>19</sup>.

- 3) 通过下子的合法性检查,确保双方棋盘内容的一致<sup>19</sup>.

每一步下棋要在本方的棋盘上下子,同时将下子的位置信息传给对方,在对方棋盘上要显示此棋子<sup>19</sup>.为保证双方内容一致,引入标记变量记录本方是下子方还是等待方,在 mouseUp 代码中要检测下子的合法性<sup>19</sup>. mouseMove 事件代码中将检测是否属本方下子,不属本方下子时,移动鼠标时小方块光标不显示,如此用户可知是否属本方下子<sup>19</sup>.另外,为使用户看清上一步棋的下子位置,光标会在刚下的棋子位置绘出<sup>19</sup>.

#### 3.2.2 辅助面板

该面板上有“放弃一手”、“认输”、“更改棋盘颜色”、“谈话”共四个按钮,前两个按钮在下棋结束前可能用,“放弃一手”在对方将弹出消息,“认输”在双方分别要弹出不同消息<sup>19</sup>.用户可通过单击“更改棋盘颜色”来选取背景色,有五种棋盘颜色供选择,每击一次棋盘背景作一次更换<sup>19</sup>.“谈话”按钮是一个非常有用的按钮,下棋过程中让用户能交谈很有必要,点击该按钮后将弹出一个谈话录入窗口,只要在该窗口的文本域中输入内容,再点击该窗口中的“发送”按钮即可,对方收到谈话后将弹出消息框显示<sup>19</sup>.

## 4 Socket 通信及消息格式

## 4.1 通信方式

在 Java 的 `net` 包中提供了丰富的类支持网络连接,Java 应用程序可以借由 URL 访问网络上的对象,本软件中通过 Java 提供的 `Socket` 类与 `ServerSocket` 类实现基于 `Socket` 的通信连接,`Socket` 通信是一种可靠的基于连接的通信方式<sup>19</sup>。在这种通信中,服务器方处监听状态,必须先运行<sup>19</sup>客户方与服务器方连通后,双方按约定的发送和接收次序以及约定信息表示完成消息传递与处理<sup>19</sup>。Java 的所有输入和输出操作都是采用基于流的机制<sup>19</sup>。通过 `Socket` 对象的 `getInputStream()` 和 `getOutputStream()` 方法分别取得其输入流和输出流,对输入流可用 `DataInputStream` 流再加过滤处理,对输出流则常用 `PrintStream` 流和 `DataOutputStream` 流过滤,进而可用这些流的方法方便地进行数据的操作处理,如 `DataInputStream` 流的 `readLine()` 方法可从流中读取一行字符串<sup>19</sup>。`PrintStream` 流的 `println` 方法可往流写入一字符串<sup>19</sup>。值得一提的是不能用 Java 的上述流传送汉字字符,其原因在于 Java 的字符编码是采用国际通用的 16 位 Unicode 字符集,但 Java 的上述过滤流在处理字符时,忽略了 Unicode 码的高 8 位,与西文 ASCII 一致,这样西文传输不会有问题,但汉字不能正确传送<sup>19</sup>。笔者的解决办法是将汉字字符转化为整数,使用 `DataOutputStream` 传送<sup>19</sup>。在对方再用 `DataInputStream` 流读取整数再反转化为字符即可,从实际使用看速度还是很快<sup>19</sup>。

## 4.2 信息交换的消息格式

下棋双方在收发信息时根据约定规定对信息传递处理<sup>19</sup>。在具体处理上要力求在网上交换的信息要少<sup>19</sup>。为此,两边棋盘内容首先按双方约定(如让子数、黑白选择等)初始化后,以后传送的是每一步的下棋情况,其它如提子等不传输,由本地的 Java 代码处理<sup>19</sup>。以下列出目前的约定表示:

- 1) 让子:"LZ"打头,下一行为让子数;
- 2) 选白棋:"BQ",缺省是客户方为黑棋;
- 3) 谈话:"T"开头,下一行为谈话内容;
- 4) 放弃一手:"FQ";
- 5) 认输:"F";
- 6) 下子:传送该子的 X,Y 坐标值<sup>19</sup>。

## 5 应用方式

基于 Java 的应用有两种方式,一是 `Application`(应用程序)形式,要含 `main()` 方法;一是 `Applet`(小应用程序)形式,要继承 `Applet` 类<sup>19</sup>。本软件在服务器方采用 Java `Application` 的形式运行<sup>19</sup>。在客户方同时支持 Java `Application` 和 `Applet` 形式<sup>19</sup>。不管采用何种方式,作为服务器的一方要先运行<sup>19</sup>。

### 5.1 双主都为 Java 应用程序之间的对弈

要求在两个机器上都要装有 Java 解释器,一台机器上运行服务器端 Java 程序,另一台机器运行客户端 Java 程序<sup>19</sup>。基于 Java 应用程序的方案的特点是在任何两台用 TCP/IP 连接的机器上进行对弈,任何一台机器都可充当服务方<sup>19</sup>。客户方在命令行中要输入代表服务器方地址的参数,供客户方与服务方建立 `Socket` 连接用<sup>19</sup>。

## 5.2 Web 服务器上 Java 应用程序与浏览器上 Java Applet 间的对弈

当从浏览器访问网页时,Applet 便被下载到客户机上执行<sup>19</sup>。由于浏览器对 Java 的安全性限制,Applet 只能与 Web 服务器上的应用程序建立网络连接<sup>19</sup>。所以服务器方的 Java 应用程序只能在 WWW 服务器上运行<sup>19</sup>。客户方只要求有一个支持 Java 的浏览器,地点灵活,但要求下棋的服务方是处于 Web 服务器上,这限制了任意地点的两人对弈<sup>19</sup>。

## 6 结束语

该软件可在校园网上使用,它对于增进职工的娱乐生活有一定的意义,尤其是可以在 Internet 网上进行远程对弈<sup>19</sup>。该软件在具体实现上有以下特点:

1) 整个软件是采用面向对象的方法设计,面向对象的开发方法可有效实现软件重用,提高软件开发效率<sup>19</sup>。Java 是一种纯面向对象的开发语言,Java 所有代码都封装在类中,通过类的继承实现软件重用<sup>19</sup>。

2) 应用灵活,既可以在命令窗口中以应用程序的形式使用,又可在 Web 环境下使用<sup>19</sup>。

3) 充分用 Java 的多线程特性将应用的功能进行合理的划分,提高应用的性能和可扩展性<sup>19</sup>。服务器方可同时与多个客户对弈<sup>19</sup>。

4) 使用 Socket 通信和内定的对弈约定实现客户方与服务器方的对话,通信量少<sup>19</sup>。

5) 操作界面友好,在操作时,用户可方便地通过移动鼠标将红色小方块移动到某位置下子;棋盘的大小、颜色在下棋时可随意更改;支持用户在下棋过程中的一些常用选择,并可在下棋过程中与对手进行交谈,交谈内容支持汉字<sup>19</sup>。双方下棋消息以弹出对话框显示提示信息<sup>19</sup>。

笔者打算作如下改进:

1) 加入多媒体支持,在下棋过程中用播入声音提示结合现在的对话框提示<sup>19</sup>。

2) 目前下棋结束时,是人为数子判胜负,以后将改为自动判胜负<sup>19</sup>。

3) 可在两个浏览器间对弈<sup>19</sup>。如图 2,使两个访问

WWW 服务器的客户端能对弈,将大大扩展应用的范围,由于 Applet 的安全性限制,Java Applet 只能与 WWW 服务器上的应用程序建立 Socket 连接,要实现两客户浏览器间的对弈必须通过 Web 服务器<sup>19</sup>。所以,运行于 Web 服务器上的 Java 应用程序在功能是要进行改造,使之成为两个浏览器 Applet 程序的信息转接站<sup>19</sup>。

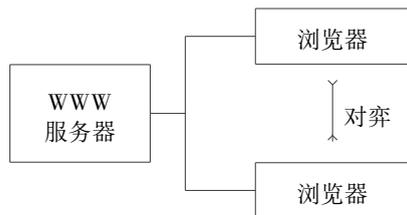


图2 浏览器间的对弈

## [ 参 考 文 献 ]

- [1] 美 Jamie Jaworski 著,曹康等译<sup>19</sup>.Java 开发指南[M]<sup>19</sup>.北京:中国水利水电出版社,1996
- [2] 张晓宁等<sup>19</sup>.基于 Java 的 CSCW 模型及其实现技术[C]<sup>19</sup>.CERNET 的研究与发展(第二卷)<sup>19</sup>.西安:西安交通大学出版社,1997,166~170

## The Design of Remote I-Go Play Software Based on Jave

DING Zhen-fan

(Modern Education and Technology center, Esat China Jiaotong University, Nanchang 330013, China)

**Abstract:** This paper presents the design of remote I-go play software written by Java. It consists of two parts, the client side and the server side. by running the server application and the client application separately, any two machines connected with TCP/IP can play I-go. If the server application is running on Web server, it can play between Browser and Web Server.

**Key words:** Java; I-Go; client/server; sockt communication; multiple thread