第 19 卷 第 12 期 小 型 微 型 计 算 机 系 统 Vol 19 No. 12
1998 年 12 月 MINI-MICRO SYSTEMS Dec 1998

JAVA 虚拟并行机的设计

王 博 王春森
(复旦大学计算机系 上海 200433)

摘 要 网络并行是高性能并行计算机领域的研究热点之一, JAVA 由于其自身的平台无关性、多线程、可移植等诸多特点正逐渐引起人们的注意, 本文在介绍上述两者的基础上, 充分利用它们的优势, 有机的把它们结合起来, 引入 JAVA 虚拟并行机的概念, 最后给出了简单实现。

关 键 词 JVPM JAVA JAVAOS 网络并行
分 类 号 TP338.8

随着计算机和网络通信技术的飞速发展, 网络并行应运而生。网络并行, 简单说, 就是通过高速信息网络, 利用网络上的计算机资源, 实现大型问题的并行计算。网络并行适合于多台微机通过高速局域网协作解决中等规模的问题, 它充分利用网络的高速弥补单个微机计算能力的不足。网络并行还适合没有大型机的微机用户通过快速网络与大型机互连, 借助大型机来解决问题。

JAVA 是近几年才推出的一种语言, 它具有平台无关性、可移植性、面向对象、多线程等特点, 正是这些特点使 JAVA 成为可移植的网络并行开发环境的最佳选择之一。本文在此想法的基础上, 对 JAVA 做了适当扩展, 并提出了 JAVA 虚拟并行机的概念以支持网络并行。

1 JAVA 平台简介

JAVA 是作为一种编程语言出现的, 但它发展迅速, 现在已提出独立的操作系统 JAVAOS。下面介绍两种 JAVA 平台的实现方案。首先是在宿主操作系统上的实现, 当在一个宿主操作系统上使用 JAVA 平台时, JAVA 虚拟机和基本类不仅嵌入到宿主操作系统, 还可以嵌入到应用程序中如 IE 和 Netscape 等浏览器。图 1 给出了在宿主操作系统上运行 JAVA 平台的软件结构。其中, JAVA 的每个应用程序都是平台无关的, 但 JAVA 平台的主要功能都需要宿主操作系统的某些支持。

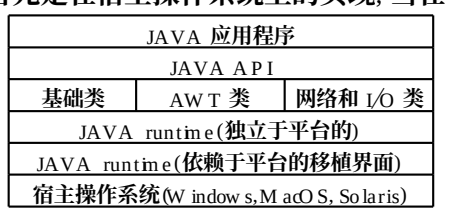

图 1 宿主操作系统上的 JAVA 平台

另一种是在非宿主操作系统上实现的 JAVA 平台 JAVAOS, 它是一个纯粹的独立的操作系

1998-05-21 收稿 王博, 24 岁, 硕士研究生, 研究方向为软件工程。王春森, 56 岁, 副教授, 研究方向为软件工程。

统,不仅可以运行在其上开发的应用程序,也可运行为宿主操作系统写的应用程序。图 2 显示了非宿主操作系统上运行 JAVA 平台的软件结构。

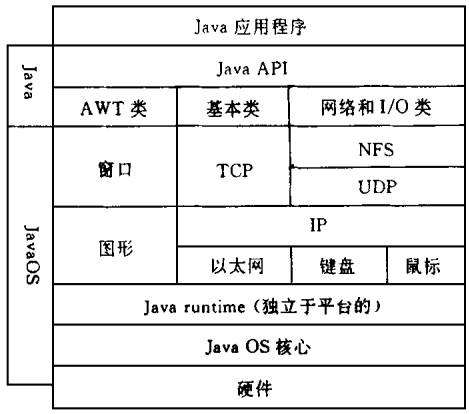

图 2 运行在 JAVAOS 上的 JAVA 平台

目前,还没有成熟的操作系统 JAVAOS 出现(据作者所知)。而更常见的 JAVA 平台是基于宿主操作系统的,实现了现有平台与 JAVA 程序的无缝连接,JAVA 程序可在所有平台上运行。具体讲,JAVA 语言开发环境包括编译环境和运行环境两部分。首先,在编译环境中,编译程序把 JAVA 源程序[*.java]编译为独立于任何特定平台的面向虚拟机的 JAVA 二进制码[*.class],它可以运行在 JAVA 平台上,这样就实现了与任何特定平台的分离,即平台无关性。在运行环境中,虚拟机通过执行虚拟机的指令,实现[*.class]文件到实际操作系统可以运行的程序格式的转换,并执行该程序。这就是"一次编程,到处使用"。

2 JAVA 虚拟并行机概念

在引入 JAVA 虚拟并行机 JVPM (JAVA Virtual parallel machine) 前,先介绍一下 JAVA 平台的扩展以及 JAVA 并行节点。正如前面所叙述,JAVA 平台有两种基本的实现方案,分别建立在宿主操作系统和 JAVAOS 基础上。(请参阅图 1,图 2) 下面以扩展 JAVAOS 为例讲解如何扩展 JAVA 平台,使其具有并行处理的机制。

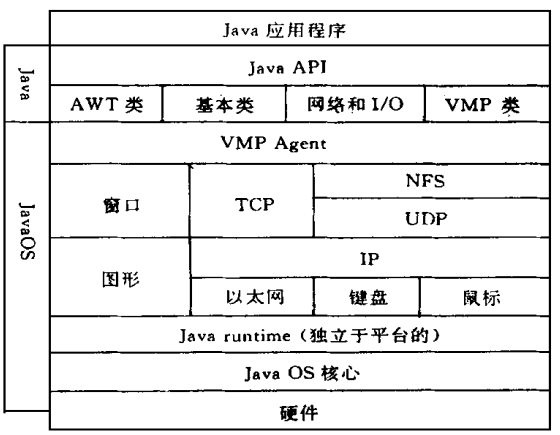

图 3 运行在 JAVAOS 的 JAVA 并行平台

(1) VPM Agent (Virtual Parallel Machine Agent) 的实现。

VPM Agent 作为 JAVA 的系统进程存在,它接受本机的并行处理请求,创建一个 JVPM 的进程,按照用户的参数配置它,使本节点加入 JVPM,利用 JAVA 的多线程机制创建一组从属于该 JVPM 进程的线程,在本机并发执行,它可以根据需要和要求,通过网络与属于本 JVPM 的其它节点机上的 VPM Agent 打交道,把任务分配到其它节点机上,由各自节点机的 VPM Agent 创建本地线程,实现网络并行。VPM Agent 还负责一个并行处理进程在本节点的不同线程之间以及不同节点上的线程通讯。同节点上的线程之间通过共享进程的数据区的方式通信。由此引起的同步问题,利用 JAVA 提供的 Syschronized 做标记的办法解决。不同节点机间的通信采用基于消息传递的 MPI(Message Passing Interface),把通信信息封装成消息,以消息的方式相互通信。这对通信者来讲是透明的,感受不到网络的存在。低层采用面向连接的 TCP/IP 协议,由于建立在 JA-

VA 平台上, 不用再考虑不同网络协议之间的转化问题。

(2) 提供 V PM 类编程接口, 用户可通过编程动态配置、运行、管理 JV PM , 实现中大规模问题的网络并行处理。图 3 给出了基于 JAVAOS 平台的实现示意图(引入并行处理功能后称为 JAVAOS '), 基于宿主机的 JAVA 平台的扩展与此类似。

按照上述方式扩展的 JAVA 平台称为 JAVA 并行节点。由于 JAVA 本身的平台无关性, 因此, 它支持从微机到巨型机, 从 W indow s 到 U nix 以及将来可能出现的任何机种、操作系统, 极大的便利了异种机间的网络并行。在此基础上相对 JAVA 虚拟机概念, 引入 JAVA 虚拟并行机(JV PM)的概念, 以支持网络并行计算。JAVA 虚拟并行机, 就是 JAVA 并行节点通过网络连接在一起, 从而构成具有中大规模运算功能的虚拟并行机。图 4 显示了 JAVA 虚拟并行机的设计原理。

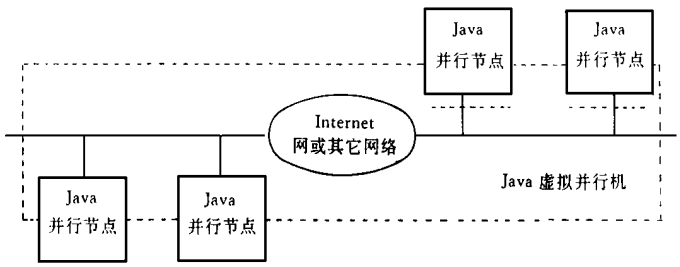

图 4　JAVA 虚拟并行机原理图

与传统意义上的并行机相比, 联结 JV PM 各节点机的网络从某种意义上讲是传统并行机的内联网络的外化, 这是随着网络速率的高速发展的结果, 当然速率方面还存在着不小的差别, 同时由于网络的特有现象(如路由问题, 堵塞现象, 网络流通量不可预测等) 也增加了实现的难度, 但也增加了并行机配置的灵活度, 大大减少了投资。JV PM 的体系结构依赖于具体的网络连接方式, 一般是较简单的总线结构或星形结构, 难以实现 M esh 结构, 超立方体等结构, 这是 JV PM 的局限性之一, 但这并不是设计 JV PM 的目标, 换句话说, 这也是它降低成本的方式之一。JV PM 是分布式存储方式的, 每个 JAVA 并行节点内部的存储器组成了整个并行机的存储器部分, 由 V PM A gent 管理。与其它的网络并行实现方式相比, JV PM 是建立在统一的 JAVA 平台上, 有一个统一的内部核心, 再加上 JAVA 语言本身的优势, 一方面减少了工作量同时又增加了系统的稳定性。

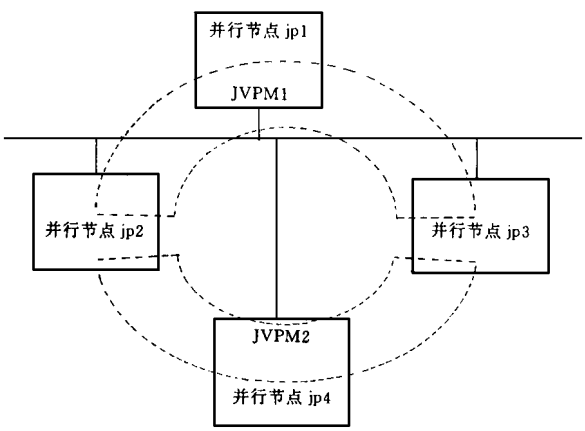

图 5　JV PM 例子

一个并行节点上可运行同一个 JV PM 的多个不同的线程, 也可加入几个不同的 JV PM , 分别同时运行几个 JV PM 的线程。JV PM 是可以动态配置的, 并行节点的加入与退出可动态实现。

在图 5 的例子中, 并行节点 jp 1, jp 2 和 jp 3 构成了 JV PM 1, jp 2, jp 3 和 jp 4 构成了 JV PM 2,

jp1, jp4 分别只加入了一个 JVPM, 而 JP2, JP3 则同时分别加入两个 JVPM, 也就是分别同时运行不同 JVPM 的线程。

3 JAVA 虚拟并行机的编程接口设计实现

在 JAVA 现有包的基础上新增包 JAVA. JVPM, 基本思路是: 主要包含两个类, 一个是类 VPM, 一个是类 PTHREAD。类 VPM 的一个实例对应一个虚拟并行机, 可以配置该并行机所包含的主机及有关参数, 测试该并行机中有关主机的信息等。

3.1 类 VPM

```
class java. jvpm. vpm {
  //constructors
  public boolean vpm (string name);
  //methods
  public string getvpminfo();
  public int addhost(inetaddress host);
  public int delhost(inetaddress host);
  public void destroy();
  public boolean isactive(inetaddress host);
  public void setopt(option opt, int val);
  public int getopt(option opt);
}
```

方法 addhost, delhost 实现节点机的加入、退出 JVPM, 达到动态配置虚拟并行机的目的。方法 getvpminfo 返回一个包含本 JVPM 的主机配置信息的字符串, 方法 isactive 判断相应主机是否活跃。方法 setopt, getopt 配置网络参数, option 包括路由策略、调试掩码、自动错误报告、消息片长度等有关网络性能的参数设置。

3.2 类 PTHREAD

```
class java. jvpm. pthread{
  //constructors
  public pthread(string name);
  //methods
  public int intovpm (string vpmname);
  public int exitvpm (string vpmname);
  public int getpriority();
  public int setpriority(int newpriority);
  public boolean isactive();
  public void start();
  public void stop():
  public void sleep();
  public void suspend();
  public void resume();
  public int read(string name, int datetype, string info);
  public int write(string name, int datetype, string info);
}
```

类 pthread 实际上是特殊的线程, 一方面具有普通线程的诸如优先级, 线程组的特征, 另一方面有一些特性, 达到不同机器上的线程在虚拟并行机上运行, 如同在同一台机器多线程运

行一样。

方法 intovpm, exitvpm 允许线程动态加入、退出虚拟并行机。方法 start(), stop(), sleep(), suspend(), resume() 执行常规的线程操作, 方法 read(), write() 提供了线程之间发送、接受消息的方式, 实现MPI消息传递, 由系统把网络封装起来, 使线程感觉不到是否是网络的存在, 达到与存取本地数据一样的目的。

4 小　结

JAVA 的出现, 从某种方面看是一件具有重大意义的事情, 它将我们逐渐引向以网络为中心的新阶段。在介绍JAVA 平台的有关内容基础上, 把JAVA 平台与网络并行结合起来, 充分利用JAVA 的平台无关性, 提出了 JAVA 虚拟并行机的概念和设计原理, 最后给出了 JAVA 编程接口的简单实现。

参　考　文　献

〔1〕 孙家旭等. 网络并行计算与分布式编程环境. 科学出版社. 北京. 1996
〔2〕 Peter Madany, JAVAOS: 一个独立的JAVA 环境. 微电脑世界. 1997. (2)
〔3〕 VJ++ 1.0 随机文档资料. 微软公司

DESIGN OF JAVA VIRTUAL PARALLEL MACHINE

WANG Bo　WANG Chunsen
(*Department of Computer, Fudan University　Shanghai* 200433)

Abstract　Network Parallel is one of the hotspot in parallel field and JAVA is being attention because of its Architecture Neutral, Multithreaded, Portable and etc. In this paper, after presenting Network Parallel and JAVA, we introduced JVPM (JAVA Virtual Parallel Machine) that take advantage of them to make a better parallel machine.

Key words　JVPM　JAVA　JAVAOS　Network parallel