

Linux系统应用丛书

The New X
Window System
(美) Charles F. Bowman 著
李建森 译

新 X Window 系统

——完整的Internet体系结构



机械工业出版社
China Machine Press

Linux 系统应用丛书

新 X Window 系统

——完整的 Internet 体系结构

(美) Charles F. Bowman 著
李建森 译



机械工业出版社

本书对 X Window 系统的新体系结构——X11R6.4 进行了全面和权威的介绍。

首先简要介绍了分布式数据处理的概念和 X Window 的历史,概览并介绍了 Broadway 的技术和经济优势,详细讲述了通过 Internet 远程执行企业应用程序所需的管理任务,最后还介绍了其他相关的技术(Java 和微软的 Terminal Server)以及与 Broadway 的互操作性。并在最后附有 X 协议的应用程序组扩展、LBX 及 RX 文档。

本书适合大规模、分布式应用程序和环境的结构建构人员、设计开发人员和管理人员使用,阅读本书后他们将对 Broadway 的优势有一定的了解,并知道如何应用这种激动人心的新技术。本书对那些负责设计企业级部署策略和结构的 IT 管理人员也很有益处,他们将从中学到如何使传统的应用程序和新的体系结构结合在一起。

Translation copyright ©2002 by CHINA MACHINE PRESS

Original English language title: New X Window System, The: A Complete Internet Architecture, First Edition by Charles F. Bowman, Copyright © 2000, All Rights Reserved

Published by arrangement with the original publisher, ADDISON WESLEY LONGMAN, a Pearson Education Company.

本书中文简体字版由原著出版公司 Addison Wesley Longman, Inc. 授权机械工业出版社独家出版。未经出版者书面许可,不得以任何方式复制或抄袭本书内容。

本书版权登记号:图字 01-2001-2193 号

图书在版编目(CIP)数据

新 X Window 系统:完整的 Internet 体系结构/(美)鲍曼(Bowman, C. F.)著;李建森译. - 北京:机械工业出版社,2002.1

Linux 系统应用丛书

ISBN 7-111-09461-1

I. 新... II. ①鲍... ②李... III. 窗口软件, X Window IV. TP316.7

中国版本图书馆 CIP 数据核字(2001)第 072491 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策 划:胡毓坚 责任编辑:汪汉友 责任印制:付方敏

三河市宏达印刷有限公司印刷·新华书店北京发行所发行

2002 年 1 月第 1 版·第 1 次印刷

1000mm × 1400mm · B5 · 4.75 印张 · 176 千字

0001-5000 册

定价:19.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话(010)68993821、68326677 - 2527

译 者 序

按照客户机/服务器体系结构设计的 X Window 系统属于通用的、跨平台的视窗系统。它在局域网上久经考验,有广泛的应用基础。随着 Internet 的蓬勃发展,它又进行一次重大的改进,使它成为新的 Internet 解决方案。

Broadway(X11R6.4 的程序名)使 X Windows 系统具有通过 Web 协议访问 X 应用程序的能力。更容易理解的名称是 X Web,它表示同样的含义。X Web 允许通过 Web 协议在 Internet 上部署 X 应用程序。用户可以通过在 Web 浏览器中点击 HTML 页内的链接,启动远程的 X 应用程序,然后将运行结果在 Web 浏览器中显示。现在,可以通过 Web 浏览器独立于操作系统平台运行 Windows、Macintosh、或者 UNIX 类的应用程序;后者包括 UNIX、Linux、BSD 和 Solaris X86 等系统的应用程序。

使用 X Web 带来的好处是允许集中部署和管理 X 应用程序,在用户端可以使用真正的瘦客户机进行工作,而不必对大量的用户进行 X Window 系统的培训。同时,企业可以继续利用 X 编程技术编写新的应用程序,可以不加改变继续使用原有的应用程序,它还提供了强有力的安全机制。X Web 的这些特色使它成为企业利用 Web 协议部署应用程序的理想解决方案。这种方案比使用 Java 的方案有一定的优势。

X Web 的体系结构包括 5 个关键部分:

- 1) 远程执行——提供远程启动 X 应用程序的机制。
- 2) 一致性访问——通过集成 X11R6.4 和 HTTP 协议提供独立于平台的、一致性访问,使它具有广泛的可用性。
- 3) Web 浏览器插入——结合远程执行,提供在浏览器中显示应用程序执行结果的功能。
- 4) 安全机制——用区分可信(内部)和不可信(外部)应用程序的方法提供安全机制。它的实现包括 3 部分:扩充 PC 机的 X 服务器功能,增加防火墙的 X 代理,并为用户提供控制在可信和不可信应用程序之间交换数据的界面。
- 5) LBX(窄带 X)——用以改进 X 在广域网和低速拨号网上的性能,使用的技术包括高速缓冲、数据压缩和简化应答。

《新 X Window 系统》一书用简练的语言和简短的篇幅对 X Web 的体系结构进行了系统和全面的介绍,并对这种新的 Internet 解决方案所具有的优势进行了详细的分析。它有利于我们掌握和使用这一新的技术。在国内广泛接受 Linux 的今天,深入掌握 X Window 系统实际是用好 Linux 的关键。从当前的情况看,有关 X

IV

Window 系统的资料还是很不够的。我们将它翻译出来供大家参考。由于时间仓促和水平有限,不当之处,敬请指教!

本书由李建森翻译,童寿彬审校。参加本书翻译工作的还有:刘钰畴、李建新、郝强、杨传有、刘建新、杜源、李元、秦杰、许刚、刘英和江涛等。借本书出版之际,谨向他们表示由衷的感谢!

前 言

Internet 不仅改变了人们与计算机交换信息的方式,而且还彻底改变了设计人员和开发人员构造和部署软件系统的方法。旧的、整体式的设计规则已不再适用。为了充分利用今天的分布式技术,程序员和系统工程师必须从根本上改变他们的应用程序的设计方法。

当然,新的部署软件的体系结构所提供的优势不应该只适用于新开发的应用程序,信息技术(IT)专业人员还应该能够把已经存在的应用程序集成到新的设计模式中去。要实现这个目标,IT 专业人员需要新的工具,可以用它来将新设计的应用程序和传统的系统结合在一起。

本书把 X11R6.4——X Window 系统的新体系结构——作为完整的 Internet 解决方案来介绍。X11R6.4(它的程序名称为 Broadway)为 IT 管理人员和开发人员提供了许多重要的技术优势。首先,它允许已经存在的企业应用程序无缝地从局域网移到广域网和 Internet 上。这就实现了 X Window 最初的设计目标。特别是,Broadway 允许任何用户从 Internet 的任何位置访问任何 X 应用程序。其次,Broadway 明显地增强了已经存在的企业应用程序的显示性能。例如,Broadway 包括几项改进 X 的应用程序性能的特征(其中最引人注意的是窄带 X)。此外,Broadway 还为 X 应用程序提供了显示的一致性。作为结果,无论用户处于什么位置(在防火墙内还是在防火墙外),应用程序的屏幕显示效果和界面看上去都一样。

最重要的是开发人员不用重新编写(或重新编译)已经存在的程序就能获得这些优势。这一特征为 IT 公司节省数以千计的美元花费。

为什么这一特征很重要?因为所有 IT 公司的预算都是有限的。要在当今的市场环境下竞争,公司必须为员工提供每一种可能的技术优势。很显然,在将旧的应用程序移植到新的环境方面多花一元钱,在开发新工具方面的费用就会减少一元钱。Broadway 用最少的费用、甚至不需要费用就能够重新构造企业已经存在的系统,使企业有条件充分利用预算来扩展它的技术储备。另外,Linux(一种 UNIX 变种)正在成为许多 IT 机构所选择的平台。因此,对最新开发的、分布式的、基于 X 的应用程序进行自然、有效的支持,已经成为许多开发公司迫切需要完成的任务。

新的开发结构之所以重要的另一个原因是:开发公司将不再需要受制于一个厂商、一种平台的解决方案。现代的 IT 解决方案是由一组硬件和软件组件所集成,在这些硬件和软件组件中有些是购买的、有些是租借的、有些是开发的。确实,开发人员面对这种环境的新问题之一就是如何面对丰富的资源做好选择。

本书的读者

本书将 X11R6.4 作为完整的 Internet 解决方案来介绍。讨论了它的体系结构和主要组件,特别是,说明了它在开发和部署分布式应用程序方面所具有的各种优势。本书还包括一些讨论与管理移植有关的章节,并对 X11R6.4 与其他的 Internet 技术(如 Java)进行了比较。此外,还提到了 X11R6.4 提供的许多工具,这些有助于保护企业在传统的系统和应用程序上的投资。

本书的读者包括开发大规模、分布式应用程序和环境的体系结构建构人员、设计人员、开发人员和管理人员。每种类型的读者都将通过阅读本书理解 Broadway 的优势所在,洞察如何利用这项激动人心的新技术以取得他们自身的优势。

本书对那些负责构思企业级部署策略和体系结构的 IT 管理人员是很有益处的,他们将学到如何延长传统的应用程序的生命周期,并使它们与新的部署方式和技术协调一致。这样既能节省时间、金钱和精力,又使管理人员能将精力集中在开发新系统和新应用程序上。

本书的内容

本书开始说明开发分布式、基于 Internet 的应用程序的要求和所涉及的历史背景。第 1 章说明分布式处理的要求;第 2 章说明 X Window 的发展情况以及产生 Broadway 的根源;第 3 章综述 Broadway,并讨论它的技术和经济优势;第 4 章详细介绍通过 Internet 远程执行企业应用程序所需的管理任务;在第 5 章中,本书以介绍 Broadway 与其他的技术(Java 和微软的 Terminal Server)的互操作性作为结束。这样,不仅为读者提供了各种方法的比较,并使读者对 Broadway 在新技术中所扮演角色有正确的理解。

最后的 3 个附录分别介绍 X11R6.4 的一些重要方面:应用程序组扩展、窄带 X 扩展和 RX 文档。

致谢

虽然在本书的封面上只有一位作者的名字,但是,出版一本书决不是一个人所能完成的任务。它需要许多尽职的专业人员的帮助。有感于此,我衷心感谢 Addison Wesley 所有给予我帮助和建议的人们。另外,我还要感谢下面这些给予我忠告和指导的评论家。他们的努力在本书的字里行间都能看到:

Rawn Shah

Hewlett - Packard 公司的 Jeff Walls

Paul Anderson

Linda Cain

康柏计算机有限公司的 Brian Brobryk

最后,当然是很重要的,我要衷心感谢我的家庭:感谢我的孩子们 Charles、

Michael 和 Nicole 对他们“讨厌的”爸爸的无限耐心；感谢我的妻子 Florence 对她的“讨厌的”丈夫的无限耐心。他们是我的快乐、我的灵感和我存在的目的。我深深地爱着他们。

Charles F. Bowman

目 录

译者序		第3章 Broadway	19
前言		3.1 全球性访问	20
第1章 使用计算机的未来趋势	1	3.2 关于 Broadway	21
1.1 Internet	2	3.2.1 标准	21
1.2 新的开发目标	2	3.2.2 设计目标	22
1.2.1 增强事务处理能力	2	3.2.3 分布式体系结构	23
1.2.2 可移动的客户机和服务器	3	3.2.4 Broadway 的体系结构	23
1.2.3 24×7 的可使用性	3	3.2.5 程序下载	23
1.2.4 增强的数据集	3	3.3 Broadway 的组件	24
1.3 分布式计算机的使用	3	3.3.1 窄带 X(LBX)	24
1.4 新的计算浪潮	4	3.3.2 嵌入浏览器	25
1.4.1 分布式体系结构	4	3.3.3 Broadway 的安全性	26
1.4.2 面向对象	7	3.3.4 其他的特性	26
1.4.3 商业对象	7	3.4 为什么使用 Broadway	27
1.4.4 分布式对象	7	3.4.1 Broadway 的优点	27
1.4.5 组件	8	3.4.2 市场上的 Broadway	27
1.5 新的设计目标	8	3.5 总结	28
1.5.1 瘦客户机	8	第4章 X Web	31
1.5.2 Internet 应用程序	9	4.1 程序下载	32
1.6 总结	10	4.2 HTML 概述	33
第2章 X 的故事	11	4.3 Broadway 和浏览器	35
2.1 X Window 系统	12	4.4 实现的要求	37
2.2 X 的体系结构	13	4.4.1 Broadway 的 HTTP/HTML 扩展	37
2.2.1 设计目标	13	4.4.2 浏览器的要求	38
2.2.2 真正的客户机/服务器	14	4.4.3 应用程序服务器的要求	38
2.2.3 X 协议	14	4.5 应用程序样例	39
2.2.4 控件的出现	15	4.5.1 Web 服务器的管理	39
2.2.5 X 的成功之处	15	4.5.2 浏览器的管理	39
2.2.6 X 成功的原因	16	4.5.3 应用程序文件	40
2.3 X:展望未来	16	4.6 带宽和性能	46
2.4 总结	17		

4.6.1 传统 X 的性能	46	5.5.2 Broadway 上的 TS	59
4.6.2 窄带 X	47	5.6 总结	60
4.7 安全问题	48	附录 A X 协议的应用程序组扩展 ..	61
4.7.1 安全威胁	48	A.1 目的和目标	62
4.7.2 传统 X 的安全	48	A.2 协议概述	62
4.7.3 Broadway 的安全	49	A.3 请求	63
4.8 总结	50	A.3.1 AppGroupQueryVersion	63
第 5 章 Broadway 的互操作性	51	A.3.2 AppGroupCreate	63
5.1 Java 的诞生	52	A.3.3 AppGroupDestroy	64
5.2 Java 的特性	52	A.3.4 AppGroupGetAttr	65
5.2.1 简单性	52	A.3.5 AppGroupQuery	65
5.2.2 面向对象	53	A.3.6 AppGroupCreateAssociation ...	65
5.2.3 编译与解释	53	A.3.7 AppGroupDestroyAssociation ...	65
5.2.4 垃圾收集	53	A.4 对已有请求的修改	65
5.2.5 可移植性	53	A.4.1 MapWindow	65
5.2.6 结构的中立性	53	A.4.2 ConfigureWindow	66
5.2.7 性能	54	A.4.3 CreateWindow	66
5.2.8 安全问题	54	A.4.4 ChangeWindowAttributes	66
5.2.9 多线程	54	A.5 对已有事件的修改	66
5.2.10 Java 虚拟机	54	A.5.1 MapRequest	67
5.3 使用 Java	55	A.5.2 ConfigureRequest	67
5.3.1 在现实世界中的可移植性 ...	55	A.6 错误	67
5.3.2 GUI 变种	55	A.6.1 AppGroupQueryVersion	67
5.3.3 安全	55	A.6.2 AppGroupCreate	67
5.3.4 重新编码的代价	56	A.6.3 AppGroupDestroy	67
5.3.5 缺乏控制	56	A.6.4 AppGroupGetAttr	68
5.4 关于 Broadway	57	A.6.5 AppGroupQuery	68
5.4.1 为什么要使用 Broadway	57	A.6.6 AppGroupCreateAssociation ...	68
5.4.2 Broadway 安全吗	57	A.6.7 AppGroupDestroyAssociation ...	68
5.4.3 Broadway 快吗	57	A.7 编码	68
5.4.4 X 和 Internet	58	A.7.1 AppGroupQueryVersion	68
5.4.5 Broadway 和 Intranet	58	A.7.2 AppGroupCreate	69
5.5 微软的 Windows NT Terminal		A.7.3 AppGroupDestroy	69
Server 版本	58	A.7.4 AppGroupGetAttr	70
5.5.1 TS 概述	59	A.7.5 AppGroupQuery	70

A.7.6	AppGroupCreateAssoc	71	B.1.6	事件压缩	81
A.7.7	AppGroupDestroyAssoc	71	B.1.7	主客户机	81
A.8	库应用程序编程接口	71	B.1.8	客户机的多路复用	81
A.8.1	Status Xag Query Version(	71	B.1.9	交换	81
A.8.2	Status Xag Create Embedded Application Group(	72	B.1.10	差值缓存	82
A.8.3	Status Xag Create Nonembedded Application Group(	72	B.1.11	流压缩	82
A.8.4	Status Xag Destroy Application Group(	72	B.1.12	验证协议	82
A.8.5	Status Xag Get Application Group Attributes(	73	B.2	C 库接口	83
A.8.6	Status Xag Query Application Group(	74	B.2.1	应用程序库接口	83
A.8.7	Status Xag Create Association(	74	B.2.2	代理库接口	83
A.8.8	Status Xag Destroy Asscciation(	75	B.3	协议	84
A.9	系统窗口编码	75	B.3.1	语法约定和常用类型	84
A.9.1	App Group Create Assoc [X11]	75	B.3.2	错误	85
A.9.2	App Group Create Assoc [Macintosh]	75	B.3.3	请求	86
A.9.3	App Group Create Assoc [Win32]	76	B.3.4	事件	104
A.9.4	App Group Create Assoc [Win16]	76	B.3.5	响应	106
附录 B	窄带 X 扩展	77	B.4	算法命名	106
B.1	说明	78	B.5	编码	107
B.1.1	数据流	78	B.5.1	事件	126
B.1.2	标记	79	B.5.2	X 事件的重新编码	128
B.1.3	简化应答	80	B.5.3	响应	129
B.1.4	图形的重新编码	80	附录 C	RX 文档	131
B.1.5	移动事件	80	C.1	表示法约定和通用语法	132
			C.2	RX MIME 类型	133
			C.2.1	一般形式	133
			C.2.2	参数	133
			C.2.3	返回的参数	135
			C.3	在 X Window 系统中如何 使用 RX 文档	136
			C.3.1	参数	136
			C.3.2	返回的参数	137
			C.3.3	样例	137
			参考文献	139	

第1章

使用计算机的未来趋势

本章要点:

- Internet(互联网)的影响
- 瘦客户机
- 分布式计算
- Intranet(内联网)的强大功能
- 新的体系结构模型

本章着重介绍使用计算机的当前状况和影响用户和软件专业人员的未来趋势。

开发应用程序的领域正在迅速发生变化。单独的、基于大型机的应用程序的时代已经成为过去,计算机产业现在已经进入了分布式计算的时代。分布式计算是一种新的体系结构,它允许把应用程序分成多块,每块可以在不同的地点执行。不管它的形式如何,这种趋势既简化了计算又加强了用户处理计算的能力。

1.1 Internet

促进分布式计算发展的最强大的动力之一是(现在是,将来也是)Internet。Internet 正在改变现代计算的面貌,也正在改变我们的生活,这决不是夸张。想一想,现在,我们可以在家中不受干扰地进行许多活动(从购物、投票到研究和辩论)就足以说明这一点。

正如人们所预料的,在商业社会里,Internet 的力量和潜能也是巨大的。通过 Internet 来接触消费者和经营业务已经成为人们经常讨论的话题。软件公司很快认识到 Internet 的威力。营销机构发现,把 Internet 作为为消费者服务的媒介可以大大减少产品的营销费用。系统设计人员还发现,Internet 可以使用户能够方便、经济地访问提高个人效率的工具和关键性应用程序。

具有讽刺意义的是:方便、经济的系统访问也是麻烦的来源。大多数机构对安全问题特别敏感,它们正在尽力保护它们的知识财富。正如人们所预料的,一些 IT 管理人员对 Internet 的全球性访问带有偏见。尽管如此,就是这些 IT 管理人员也不能忽视由新的计算模式所带来的技术优势。因此,为了充分利用这种新的应用程序的体系结构并使风险降到最低限度,就形成了一种折中方案:Intranet。概括地讲,Intranet 是封闭的、专用的 Internet。从本质上讲,它是与 Internet 具有相同的基本体系结构的企业内部网络,但却大大降低了安全风险。

1.2 新的开发目标

由于计算环境的变化,软件专业人员必须从两个方面改变他们开发应用程序的方法,这两个方面是体系结构和人的效率的充分发挥。在体系结构方面,系统设计人员必须利用新技术,使开发人员可以创建更容易实现和维护的应用程序。特别是,必须使这些应用程序达到在下面几个小节中设定的目标。

1.2.1 增强事务处理能力

当我们进入分布式应用程序的领域时,一项简单的事务(换句话说,一个工作单位)可能跨越多个主机和多个进程。然而,这个应用程序仍然必须保证每个事务

的固有完整性,也就是说,要么把这个事务全部做完(没有被中断过),要么把系统恢复到原始状态。由于这种体系结构本身的原因,定义和实现分布式环境中的基本事务是必须的。特别是,随着对分布式体系结构的使用的增加,完成一项单独的商业事务所需的多个、独立的组件服务的数量很可能也需要增加。如果其中一个或多个组件没有完成它所应该完成的任务,那么,很可能所有与之相关的进程(在其他的组件中的进程可能已经完成了)都必须恢复到原始状态(undone)。

1.2.2 可移动的客户机和服务器

在分布式环境中,用户和应用程序都是可移动的。也就是说,用户可以通过 Internet 从世界上的任何地点访问应用程序。因此,不管在什么地点,应用程序都必须能够“找到它们的服务器”。而且,这种设计还有一个附加的优点,即系统管理员可以根据诸如资源要求、硬件失败和 CPU 性能等等因素,在可使用的计算机之间移动应用程序。这样也就提高了硬件的使用效率,在这种系统中,系统管理员还可以适当地调整硬件组件的大小并确定这些硬件组件。

1.2.3 24×7 的可用性

Internet 要求大部分商业应用程序具有每周 7 天和每天 24 小时的可用性。在这方面,分布式环境允许管理人员拥有复制应用程序的服务。这样,尽管有间歇的失败和定期的系统维护,还是大大提高了整个应用程序的可用性。

1.2.4 增强的数据集

应用程序不再只处理简单的数据类型。当今的技术允许系统设计人员甚至把增强的对象(如视频、音频和多媒体)加入到大多数基本应用程序中。

第二个方面的变化涉及应用程序的人机工程学问题。设计人员必须开发和使用这样的新技术,这些技术使用户能够以直觉方式访问关键性任务和提高个人工作效率的应用程序。虽然应用程序被移植到分布式体系结构中,但是,从用户看来,这些应用程序及它们的服务应该是无缝的和一致的。也就是说:不管底层的体系结构如何变化,用户所看到的应用程序都应该是一个简单而且一致的界面。否则,所有的技术优势都没有意义。

1.3 分布式计算的使用

虽然早就开始有了分布式计算的概念,但是直到现在,我们才实现分布式计算。许多与分布式处理有关的概念并不是什么新概念。事实上,其中的一些概念的出现甚至比计算机工业本身还要早。因此,我们最好以提问的方式来开始讨论:为什么我们现在应该使用分布式计算?第一个答案是:因为我们现在能够使用分布式计算。

过去我们不能够利用分布式计算的优势,因为我们缺乏一些必要的基础资源。现在我们可以实现分布式计算的优点是因为许多重要的技术进步了。这些技术包括:

- 低成本、宽带网络的迅速扩张。
- 高性能、低成本的 CPU 的普遍使用。
- 操作系统平台提供了广泛的特征。
- 改进了的开发工具和方法。

注意,这些优势之间有相互影响。例如, CPU 费用的减少将允许系统开发人员提供具有扩展特征的操作系统,这种操作系统又允许开发人员实现功能更强大的工具,这些工具反过来又可以支持和提高应用程序设计人员的能力。

我们应该开始提供分布式计算的另一个原因是:因为我们必须使用分布式计算。商业(所有的商业)的性质正在发生变化:雇员更频繁地旅行、家庭办公已是很平常的事,而且消费者在地理的分布和在文化方面的差异都很大。采用分布式体系结构不再是一种奢侈,它已经变成了一种实际需要。

1.4 新的计算浪潮

深远影响现代计算体系结构的基础有两个重要的概念:面向对象和客户机/服务器体系结构。然而,要充分利用这些开发模式所提供的优势,我们必须理解分布式计算的性质。

1.4.1 分布式体系结构

简单地讲,我们可以将应用程序划分为若干块,每一块在合适的地点执行,以此来实现分布式体系结构。这种概念虽然简单,但是实际有效地实现这种体系结构是困难的。此外,通常有若干相互冲突的术语来描述不同形式的分布式处理。为此,我们从几个定义开始进行讨论。

(1) 分布式计算

术语分布式计算描述这样的情况:在这种情况下,我们把一个应用程序划分成多个单独的任务(其中的每个任务都独立地执行,通常把这种任务称作进程)。这是一般性的定义,它既不要求任务之间相互进行通信,也不要求这些任务必须驻留在单独的处理单元中。

(2) 分布式处理

真正的分布式处理要求至少有两个应用程序的组件,它们必须在独立的处理器(不同的平台)上被执行。这些组件必须是应用程序的一部分,也就是说,它们不是操作环境的一部分。所有的任务可以同步执行,也可以异步执行,而且它们可以通过某种协议进行通信。注意,在多处理器系统上运行一个应用程序不能称作真

正的分布式处理。

(3) 客户机/服务器体系结构

从本质上讲,客户机/服务器体系结构是一种分布式计算和分布式处理的混合体。在这种模型下,我们把应用程序分成两个单独的组件,但是,我们现在给予它们特殊的名称(并扮演不同的角色):客户机和服务器。这些组件独立地执行,而且完成特定的任务:服务器提供服务;客户机请求这些服务。客户机和服务器使用一种(可能是惟一的)协议来交换信息。图 1-1 提供了一个样例。

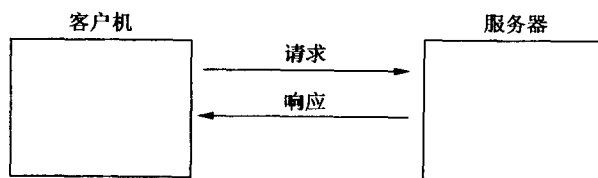


图 1-1 客户机/服务器体系结构

注意,这种模型的内在特征是:服务器可以为许多客户机提供服务,而客户机可以从许多服务器那里得到服务。正如我们所看见的,客户机/服务器体系结构是 X Window 系统结构设计的基础。而且, Broadway 利用了这种体系结构的优点,形成了完整的 Internet 解决方案的基础。

(4) 客户机/服务器的优点

与传统的结构相比,客户机/服务器模型包含了许多固有的优点。这些优点如下:

1) 分布式处理。在客户机/服务器模型下,开发人员可以把处理负载分布到多台机器上(假定某些组件能够并将在独立的处理器上执行)。因此,我们可以获得真正的并行处理(即同时执行多条机器指令)。更重要的是,应用程序不再依赖单独的 CPU 来处理。也就是说,管理员可以在若干机器上复制服务。如果一台主机失败了,可以使用另一台主机来承担处理任务。

2) 服务的独立性。除安全问题外,服务的独立性意味着服务器可以处理来自多个客户机的请求。而且,客户机不需要与服务器或其他的客户机共享通用的平台、语言、开发模式等等。进行协作的组件只需要理解请求/响应协议的格式即可。另外,服务的独立性还有利于客户机,它们可以选取多个服务器的多个服务来完成一个给定的应用程序任务。

3) 模块化设计。我们可以构造满足特殊需要的、定制的客户机。例如,考虑经典的数据库服务器,我们可以开发一些专用的客户机,每个客户机在特定的执行环境(例如,MS Windows、X Window、ASCII 终端等等)产生显示。

4) 负载均衡。如前所述,我们可以跨处理器来分布服务。作为附加的优点,我们可以适当地选择客户机或服务器来分担负载。例如,如果由于对打印服务器

的请求太多,使处理器 A 负担过重,那么,我们可以启动机器 B 上的另一台打印服务器。这个新的服务器可以分担处理器 A 的一些负担。注意,可以使用手动和自动过程来处理负载平衡。

5) 冗余。我们可以通过在系统中设计组件冗余来增加系统的可用性。参见前面的例子。如果处理器 A 失败了,那么我们可以自动地把所有的打印请求发送给处理器 B。当然,B 的性能将降低,然而,作为整体应用程序仍然保持正常运行。

6) 组件的可扩展性。只要我们保持对现有协议的向后兼容性,我们可以在需要时升级单独的组件。因此,我们可以扩展和维护每个组件而不影响系统的其他部分。注意,这时可能需要在协同工作的组件之间进行协商(negotiation),以确定合适的协议版本。

7) “选择合适的规模”。购买硬件已经成为日常的购买活动。我们可以混合和配备不同厂商生产的产品和平台。例如,为多个站点提供分布式打印服务。如果一个站点需要最少数量的打印服务,我们可以配置一台整修过的 386 来运行打印服务器;然而,如果另一个站点需要大量的打印服务支持,我们可以为它安装高端的 Linux 服务器。由此可知,我们不再需要依赖于一个厂商、一种平台的解决方案。事实上,在这种环境下开发人员所面对的新问题是要知道如何从丰富的资源中作出合适的选择。

8) 资源的集成。我们可以在单独的实体(服务器)中集中管理与安全和经营规则有关的组件,而不是把这些组件分散。这样做,在维护和扩展方面有明显的优势。特别是,当经营规则改变时,我们只需要修改一个程序,就能确保完全的一致。

9) 互操作性。客户机和服务器不需要使用相同的语言、数据库结构、操作系统和硬件平台等等。它们只需要使用相同的协议。

10) 对传统系统的支持。没有人要求你的服务器必须是用 C++ 编写的、在 Linux 平台上运行的、最近开发的程序。它也可以是一个过时的 COBOL 应用程序。

(5) 客户机/服务器的缺点

正如谚语所言:世上没有免费的午餐。这个谚语多少也反映了客户机/服务器结构的真实情况。在这种结构中,还有许多问题是(信息技术)专业人员所必须要面对的。这些问题如下:

1) 对网络的依赖性。当客户机在远离服务器的平台上执行时,应用程序对底层的基础结构和网络的依赖性就显现出来了。

2) 延迟。如在第一个缺点中所暗示的,当请求和响应在网络中穿行时,应用程序运行的延迟就增加了。这是一个关键的设计问题,但常常被开发人员(新手)所忽略。网络异常确实发生过。因此,为了确保这样的问题不会对应用程序造成较大的破坏,IT 专业人员必须设计容错结构。

3) 增加了的管理任务。分布式结构的分散和相互依赖之间的矛盾增加了管