



深度阐释Linux操作系统原理的里程碑之作，由拥有超过10年研发经验的资深Linux专家撰写

以从零开始构建一个完整的Linux操作系统的过程为依托，宏观上全面厘清了构成Linux操作系统的各个组件以及它们之间的关系，微观上深入探讨了核心组件的基本原理以及相互间的协作关系，指引读者在富有趣味的实践中参透操作系统的本质




深度探索 Linux操作系统

系统构建和原理解析

Inside the Linux Operating System

王柏生 著



 机械工业出版社
China Machine Press

深度探索 Linux操作系统 系统构建和原理解析

Inside the Linux Operating System

王柏生 著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

深度探索 Linux 操作系统：系统构建和原理解析 / 王柏生著. —北京：机械工业出版社，2013.10
(原创精品系列)

ISBN 978-7-111-43901-1

I . 深… II . 王… III . Linux 操作系统 IV . TP316.89

中国版本图书馆 CIP 数据核字 (2013) 第 208809 号

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问 北京市展达律师事务所

本书是探索 Linux 操作系统原理的里程碑之作，在众多的同类书中独树一帜。它颠覆和摒弃了传统的从阅读 Linux 内核源代码着手学习 Linux 操作系统原理的方式，而是基于实践，以从零开始构建一个完整的 Linux 操作系统的过程为依托，指引读者在实践中去探索操作系统的本质。这种方式的妙处在于，让读者先从宏观上全面认清一个完整的操作系统中都包含哪些组件，各个组件的作用，以及各个组件间的关系，从微观上深入理解系统各个组件的原理，帮助读者达到事半功倍的学习效果，这是作者潜心研究 Linux 操作系统 10 几年的心得和经验，能避免后来者在学习中再走弯路。此外，本书还对编译链接技术（尤其是动态加载和链接技术）和图形系统进行了原理性的探讨，这部分内容非常珍贵。

全书一共 8 章：第 1 章介绍了如何准备工作环境。在第 2 章中构建了编译工具链，这是后面构建操作系统各个组件的基础。在这一章中，不仅详细讲解了工具链的构建过程，而且还通过对编译链接过程的探讨，深入讨论了工具链的组成及各个组件的作用，理解工具链的工作原理对理解操作系统至关重要。第 3~4 章，从零开始构建了一个具备用户字符界面的最小操作系统，详细讲解了构建的过程以及涉及的技术细节。第 5 章从理论的角度探讨了这一过程，从内核的加载、解压一直讨论到用户进程的加载，包括用户空间的动态链接器为加载程序所作的努力。第 6~7 章首先构建了操作系统的基础图形系统，然后在此基础上构建了桌面环境。第 8 章深入探讨了计算机图形的基础原理，包含 2D 和 3D 程序的渲染、软件渲染、硬件渲染等内容，同时也从操作系统的角度审视了 Pipeline。

机械工业出版社（北京市西城区百万庄大街 22 号 邮政编码 100037）

责任编辑：姜 影

北京市荣盛彩色印刷有限公司印刷

2013 年 10 月第 1 版第 1 次印刷

186mm × 240mm · 27.25 印张

标准书号：ISBN 978-7-111-43901-1

ISBN 978-7-89405-088-5（光盘）

定价：89.00 元（附光盘）

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

客服热线：(010) 88378991 88361066

投稿热线：(010) 88379604

购书热线：(010) 68326294 88379649 68995259

读者信箱：hzsj@hzbook.com

谨以此书献给恩师李明树先生。

为什么要写这本书

真正认真开始学习计算机是在 2000 年，当时书店里到处充斥着一系列如“21 天精通 xxx”、“7 天掌握 xxx”之类的图书，更有甚者宣称“24 小时学会 xxx”。既是高科技，又这么容易学，谁会拒绝呢？于是我走上了这一行。最初，确实如这些书所说，只要按照书中描述，将类似于 Visual Studio 等 IDE 安装到机器上，然后像搭积木一样，拖拽几个控件，再添加几行代码，一个程序就完成了。

短暂的兴奋后，好奇心驱使我更深层次地探索这一切是如何发生的。于是我开始关注更多的书籍、更多的文章、更多的编程参考，国内的、国外的。但是，结果让我很沮丧，如果依然是用积木来举例子，我发现它们的区别就像一盒 10 块的积木和一盒 100 块的积木，只有量的变化，没有质的区别。有人说 Win32 编程更底层，于是我抛开 MFC，研究 Win32 编程。但是，结局一样让我失望。其实它们也没有本质区别，只不过如果把 MFC 比作大块积木，Win32 是小块积木而已。其间我又遍寻那些 Windows 内幕的书进行研读，也是铩羽而归，似乎前方已无路可走……

2003 年 4 月毕业后，我到了中科院软件所工作，开始从事与 Linux 相关的开发。经历了从 Windows 到 Linux 转型的阵痛后，我开始喜欢上了 Linux，因为它是开源的，我似乎看到了曙光。于是我开始疯狂地购买 Linux 方面各种各样的书籍，阅读各种权威资料，基本上网络上各种权威专家推荐的书籍在我的书桌上全部可以找到。其中，绝大部分是关于内核源码分析的书，于是我一头扎进讲解内核源码分析的书中。但是我很快淹没在庞大的内核代码中，几次都到了难以坚持的程度，但是我强迫自己坚持，强制自己接受作者的灌输。但是，最终的结果是：看的时候似乎明白，但是看完后感觉又什么也没有看。现在回头看，当初很像“盲人摸象”这个典故所描述的，在我还没有看清整个“大象”的时候，我就直接去研究“大象”的某些部分的构造了。

彷徨中，我又看到了另外一条路，低版本的内核。我就像一个在沙漠中饥渴难忍的人突然看到了绿洲，我甚至将低版本的内核打印出纸版，然后就像拿着伟人语录一样，只要觅得空隙，就虔诚地潜心研读。但是这条新路除了代码量小了点，与之前的相比并没有太多本质的区别，而且还有一个致命的缺点——早期版本的内核不能和工作中使用的 Linux 很好地结合。

2005年，我从软件所被派到了中科红旗。最初从事桌面操作系统的开发，使用的是基于Qt的KDE，因为比较成熟，所以当时做得更多的是一些维护工作。但是在我的探索过程中依然重复着上面的故事，没有任何的起色。转折大概出现在2007年，Intel因为一个低功耗平台项目开始和中科红旗合作，他们要在低功耗平台上开发一套Linux操作系统，我接手了这项工作。因为这个平台的处理器性能相对要低，所以对于操作系统的要求比较高。同时因为用于消费类电子产品，用户体验要求也与普通的PC环境完全不同。所以，基于已有的桌面系统几乎是不可能了。于是，我们开始从头开发和定制。

这个从零开始的过程，让我彻底认识了整个Linux操作系统，而不仅仅是Linux的内核。曾经对内核中很多做法和模块不明了，通过构建整个操作系统，我豁然开朗。比如，内核中的DRM模块，其全称是Direct Rendering Manager，从字面上看是直接渲染管理，这到底是什么意思？如果你仅仅从内核的角度来理解，相信我，你永远也不能正确理解它。恰恰是在构建系统时，亲手组装和调试图形环境，包括X、OpenGL、2D/3D图形驱动，让我明白了DRM的用途。这样的例子举不胜举。

经过这个过程中，我深刻认识到，学习操作系统，有三件最重要的事：第一是实践，第二依然是实践，第三还是实践。老祖宗说“纸上得来终觉浅”，唯物主义者说“实践是检验真理的唯一标准”，两句话中都蕴含着同一个道理——追求真理离不开实践。只是阅读、分析源码还远远不够，我们要动手实践，从实践中学习，实践反过来再促进思考。而且，实践也使学习不再是一个枯燥乏味的负担，而是一个乐趣。

通过这个过程，我也体会到，即使只为了学习内核，也不能将目光全部放在内核上。从整个操作系统的角度，从各个组件间关系的角度理解内核，效果反而更好。当对整个系统有了深入的理解后，再去理解组成操作系统的各个组件，会事半功倍。一旦从总体上理解了系统，你就会“艺高人胆大”，就可以尽情地“折腾”Linux系统了，因为每一个组件尽在你的掌握之中。而恰恰在这不断的“折腾”中，理论又得到不断的提高，从此进入一个良性循环。

很早我就想把这种方法整理成书，和更多的读者分享，希望帮助所有有志于操作系统、又尚在门外徘徊的年轻人少走些弯路。但是因为忙于生计，只能在有限的业余时间写作，所以直到2013年中期，才基本把整个书稿写完。

对于计算机而言，操作系统的重要性不言而喻，但它也是我们心中的痛，我将为此求索一生。如果有生之年没能成功，请将我埋在后来者脚下。

读者对象

对于如同笔者一样怀揣操作系统梦的爱好者，希望本书能帮他们顺利地迈进操作系统这扇门；对于正在或者准备学习操作系统理论的大学生，本书将帮助他们感性触摸那些“高居庙堂之上”的抽象理论；对于高级读者，本书中的很多内容对他们也很有用处，比如动态

链接部分的讨论、Linux 图形原理部分的讨论等。

除了以上的读者外，本书适合以下相关从业人员阅读：

- 系统程序员。要想成为一个合格的系统程序员，操作系统和编译链接技术是必不可少的技能，本书对此有较深入的讨论。
- 嵌入式 Linux 工程师。作为一名嵌入式 Linux 工程师，应该知道如何使用交叉编译工具链、配置编译内核、裁剪系统、搭建图形系统，甚至定制桌面环境，这些相关知识读者在本书中都可以找到。
- Linux 发行版工程师。作为制作发行版的工程师，更需要彻底熟悉操作系统的每个组件以及组件间的关系，本书可以满足他们这方面的需求。
- Linux 应用开发工程师。对于应用开发程序员，也推荐阅读本书，因为越深入地理解操作系统和编译链接原理，就越能写出高效而简洁的程序。

如何阅读本书

本书围绕着构建一个完整的 Linux 操作系统这一主线展开，除了第 1 章外，其余各章环环相扣，所以请读者严格按照章节顺序阅读。

工欲善其事，必先利其器。尤其是对于这样一本实践丰富的书来说，工作环境是后续内容的基础。因此，第 1 章介绍了如何准备工作环境。但是类似安装 Linux 发行版这样的内容，相关参考随处可见，因此书中并没有浪费篇幅去一一介绍，而是仅仅指出其中需要特别注意之处。

工具链是后面进行构建的基础，因此，接下来在第 2 章中构建了工具链。工具链是整个操作系统中非常重要的一部分，理解工具链的工作原理，对理解操作系统至关重要，所以第 2 章中并没有仅仅停留在构建的层次，还通过探讨编译链接过程，讨论了工具链的组成以及各个组件的作用。

在第 3 章和第 4 章，我们从零开始，构建了一个具备用户字符界面的最小操作系统。同时，在第 5 章，我们从更深层次的角度探讨了这一切是如何发生的。我们从内核的加载、解压一直讨论到用户进程的加载，包括用户空间的动态链接器为加载程序所做的努力。

在第 6 章和第 7 章，我们首先构建了系统的基础图形系统，然后在其上构建了桌面环境。在第 8 章，我们深入探讨了计算机图形的基础原理，讨论了 2D 和 3D 程序的渲染、软件渲染、硬件渲染，我们也从操作系统的角度审视了 Pipeline。

笔者强烈建议读者在真实的计算机上安装一个 Linux 操作系统，让它成为你日常的工作机。然后将书中的，尤其是与实践相关的所有命令实际运行一遍。之后再尝试脱离本书，自己争取从头再构建一遍，相信你一定会在这个过程中受益匪浅的。

勘误和支持

由于作者水平有限，加之编写时间仓促，书中难免会出现一些错误或者不准确的地方，恳请读者提出宝贵意见，批评指正。来信请发送至邮箱 baisheng_wang@163.com，笔者会尽自己最大的努力给出回复。

致谢

首先感谢恩师李明树先生，是他将我带进了操作系统这扇大门。

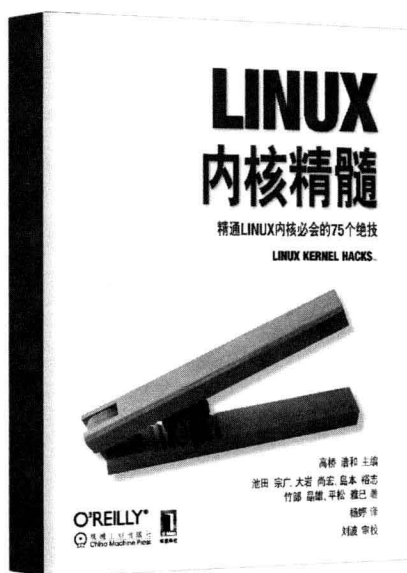
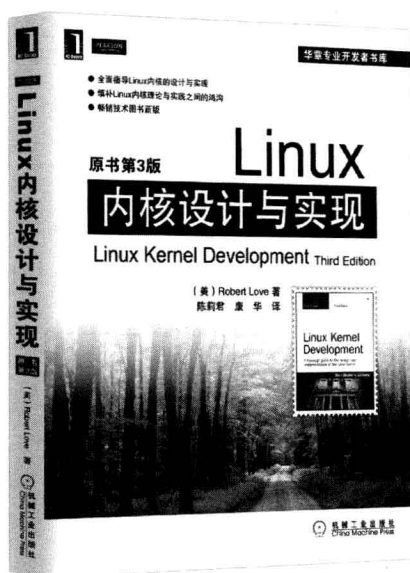
感谢机械工业出版社华章公司的策划杨福川，在他身上我看到了专业精神，这也是我在与几个出版团队沟通后，毫不犹豫地决定请他们出版的原因。

感谢机械工业出版社华章公司的姜影编辑，她清晰的思路让我深深折服。每每在遇到困难不知如何表达时，她都能通过简单的几句话点醒梦中人。

感谢我的父母，感谢他们的养育之恩。感谢我的哥哥，为了让我受到更好的教育，在他刚刚毕业不久，就顶着生活的压力，将我从农村接到了城里接受教育，为我的学业奔波操劳。感谢我的嫂子在生活上给予我的无微不至的照顾。把最后一份感谢留给我的妻子，是她在我工作这些年，承担了照顾父母、操持家务的重任，是她的无私付出让我能全身心地投入到工作和学习中。

王柏生
北京

推荐阅读



Linux内核设计与实现 (原书第3版)

世界范围内公认的Linux内核经典著作，畅销全球多个国家

Linux内核精髓

畅销书，一线内核技术专家经验和智慧结晶，深刻解读Linux内核的资源管理、文件系统、网络、虚拟化、省电技术、调试、性能调优、分析与追踪等核心主题

推荐阅读



AIX UNIX系统：管理、维护与高可用集群建设

作者：文平 ISBN：978-7-111-35951-7 定价：79.00元

资深UNIX系统专家、数据库专家、

系统架构师近20年一线工作经验结晶，系统、全面、实用

紧扣实际生产环境，从系统架构、工程部署、管理维护、性能优化、高可用环境建设等多角度详细探讨AIX系统的管理和运维的方法与最佳实践

构建高可用Linux服务器

作者：余洪春 ISBN：978-7-111-36055-1 定价：79.00元

基于实际生产环境，从Linux虚拟化、集群、服务器故障诊断与排除、系统安全性等多角度阐述构建高可用Linux服务器的最佳实践

资深Linux/Unix系统管理专家兼架构师多年一线工作经验结晶，51CTO和ChinaUnix等知名社区联袂推荐

目 录

前 言

第 1 章 准备基本环境 1

- 1.1 安装 VirtualBox 1
- 1.2 创建虚拟计算机 2
- 1.3 安装 Linux 系统 2
- 1.4 使用 root 用户 5
- 1.5 启用自动登录 5
- 1.6 挂载实验分区 6
- 1.7 安装 ssh 服务器 6
- 1.8 更改网络模式 7
- 1.9 安装增强模式 8
- 1.10 使用 Xephyr 8

第 2 章 工具链 10

- 2.1 编译过程 10
 - 2.1.1 预编译 12
 - 2.1.2 编译 14
 - 2.1.3 汇编 17
 - 2.1.4 链接 31
- 2.2 构建工具链 39
 - 2.2.1 GNU 工具链组成 40
 - 2.2.2 构建工具链的过程 40
 - 2.2.3 准备工作 43
 - 2.2.4 构建二进制工具 45
 - 2.2.5 编译 freestanding 的交叉编译器 46
 - 2.2.6 安装内核头文件 49
 - 2.2.7 编译目标系统的 C 库 50

- 2.2.8 构建完整的交叉编译器 52
- 2.2.9 定义工具链相关的环境变量 54
- 2.2.10 封装“交叉” pkg-config 54
- 2.2.11 关于使用 libtool 链接库的讨论 56
- 2.2.12 启动代码 57

第 3 章 构建内核 62

- 3.1 内核映像的组成 62
 - 3.1.1 一级推进系统—— setup.bin 63
 - 3.1.2 二级推进系统——内核非压缩部分 65
 - 3.1.3 有效载荷——vmlinux 65
 - 3.1.4 映像的格式 66
- 3.2 内核映像的构建过程 68
 - 3.2.1 kbuild 简介 68
 - 3.2.2 构建过程概述 71
 - 3.2.3 vmlinux 的构建过程 71
 - 3.2.4 vmlinux.bin 的构建过程 75
 - 3.2.5 setup.bin 的构建过程 80
 - 3.2.6 bzImage 的组合过程 81
 - 3.2.7 内核映像构建过程总结 82
- 3.3 配置内核 86
 - 3.3.1 交叉编译内核设置 86
 - 3.3.2 基本内核配置 87
 - 3.3.3 配置处理器 88
 - 3.3.4 配置内核支持模块 90

3.3.5	配置硬盘控制器驱动	91	文件	144
3.3.6	配置文件系统	96	4.6.8 启动 udevd 和模拟热插拔	146
3.3.7	配置内核支持 ELF 文件格式	97	4.7 挂载并切换到根文件系统	147
3.4	构建基本根文件系统	99	4.7.1 挂载根文件系统	147
3.4.1	根文件系统的基本目录结构	99	4.7.2 切换到根文件系统	149
3.4.2	安装 C 库	100	第 5 章 从内核空间到用户空间	154
3.4.3	安装 shell	101	5.1 Linux 操作系统加载	154
3.4.4	安装根文件系统到目标系统	102	5.1.1 GRUB 映像构成	155
第 4 章 构建 initramfs		104	5.1.2 安装 GRUB	160
4.1	为什么需要 initramfs	104	5.1.3 GRUB 启动过程	165
4.2	initramfs 原理探讨	105	5.1.4 加载内核和 initramfs	170
4.2.1	挂载 rootfs	106	5.2 解压内核	181
4.2.2	解压 initramfs 到 rootfs	110	5.2.1 移动内核映像	182
4.2.3	挂载并切换到真正的根目录	116	5.2.2 解压	186
4.3	配置内核支持 initramfs	117	5.2.3 重定位	187
4.4	构建基本的 initramfs	118	5.3 内核初始化	190
4.5	将硬盘驱动编译为模块	121	5.3.1 初始化虚拟内存	190
4.5.1	配置 devtmpfs	121	5.3.2 初始化进程 0	201
4.5.2	将硬盘控制器驱动配置为模块	126	5.3.3 创建进程 1	206
4.6	自动加载硬盘控制器驱动	130	5.4 进程加载	209
4.6.1	内核向用户空间发送事件	131	5.4.1 加载可执行程序	211
4.6.2	udev 加载驱动和建立设备节点	136	5.4.2 进程的投入运行	223
4.6.3	处理冷插拔设备	139	5.4.3 按需载入指令和数据	234
4.6.4	编译安装 udev	141	5.4.4 加载动态链接器	243
4.6.5	配置内核支持 NETLINK	142	5.4.5 加载动态库	246
4.6.6	配置内核支持 inotify	143	5.4.6 重定位动态库	250
4.6.7	安装 modules.alias.bin		5.4.7 重定位可执行程序	268
			5.4.8 重定位动态链接器	271
			5.4.9 段 RELRO	274
			第 6 章 构建根文件系统	278
			6.1 初始根文件系统	278
			6.2 以读写模式重新挂载文件系统	280
			6.3 配置内核支持网络	282

6.3.1	配置内核支持 TCP/IP 协议	282	7.1.3	主要数据结构	328
6.3.2	配置内核支持网卡	283	7.1.4	初始化	331
6.4	启动 udev	285	7.1.5	为窗口“落户”	334
6.5	安装网络配置工具并配置网络	285	7.1.6	构建窗口装饰	337
6.6	安装并配置 ssh 服务	287	7.1.7	绘制装饰窗口	341
6.7	安装 procps	291	7.1.8	配置窗口	343
6.8	安装 X 窗口系统	291	7.1.9	移动窗口	345
6.8.1	安装 M4 宏定义	292	7.1.10	改变窗口大小	348
6.8.2	安装 X 协议和扩展	292	7.1.11	切换窗口	348
6.8.3	安装 X 相关库和工具	294	7.1.12	最大化 / 最小化 / 关闭窗口	351
6.8.4	安装 X 服务器	296	7.1.13	管理已存在的窗口	354
6.8.5	安装 GPU 的 2D 驱动	297	7.2	任务条和桌面	356
6.8.6	安装 X 的输入设备驱动	297	7.2.1	标识任务条的身份	357
6.8.7	运行 X 服务器	300	7.2.2	更新任务条上的任务项	358
6.8.8	一个简单的 X 程序	302	7.2.3	激活任务	359
6.8.9	配置内核支持 DRM	303	7.2.4	高亮显示当前活动任务	360
6.9	安装图形库	307	7.2.5	显示桌面	361
6.9.1	安装 GLib 和 libffi	307	7.2.6	桌面	362
6.9.2	安装 ATK	307	第 8 章	Linux 图形原理探讨	364
6.9.3	安装 libpng	308	8.1	渲染和显示	364
6.9.4	安装 GdkPixbuf	308	8.1.1	渲染	365
6.9.5	安装 Fontconfig	308	8.1.2	显示	365
6.9.6	安装 Cairo	311	8.2	显存	366
6.9.7	安装 Pango	311	8.2.1	动态显存技术	367
6.9.8	安装 libXi	311	8.2.2	Buffer Object	370
6.9.9	安装 GTK	312	8.3	2D 渲染	375
6.9.10	安装 GTK 图形库的善后工作	312	8.3.1	创建前缓冲	377
6.9.11	一个简单的 GTK 程序	313	8.3.2	GPU 渲染	381
6.10	安装字体	315	8.3.3	CPU 渲染	386
第 7 章	构建桌面环境	317	8.4	3D 渲染	388
7.1	窗口管理器	317	8.4.1	创建帧缓冲	390
7.1.1	基本原理	318	8.4.2	渲染 Pipeline	399
7.1.2	创建编译脚本	325	8.4.3	交换前缓冲和后缓冲	414
			8.5	Wayland	421

准备基本环境

在开始 Linux 操作系统的探索旅程之前，我们首先需要准备一下环境，读者最好在真实的计算机上安装一个 Linux 操作系统作为工作机。毫无疑问，使用是最好的学习方法，如果日常工作系统也是 Linux，那么这无疑有助于更好地理解 Linux 操作系统。但是这并非必须的，也可以安装一台虚拟机作为工作机。鉴于现在的 Linux 发行版的安装过程非常友好和自动化，本章无意浪费版面介绍其安装过程。

另外，在构建操作系统时，需要频繁重启系统，因此强烈建议读者不要使用工作机作为实验机，而是另外安装一台虚拟机作为实验机。本章将介绍如何创建一个虚拟的裸机以及如何在其上安装 Linux 操作系统，并且介绍为了后面的开发和调试，在虚拟机上需要进行的一些必要的准备。

因为桌面环境可以利用一个模拟的小 X 服务器 Xephyr 来调试，所以我们可以先在宿主机的 Xephyr 上进行开发和调试，然后再到构建的真实系统上调试。因此，本章的最后部分介绍了如何使用 Xephyr。

1.1 安装 VirtualBox

笔者建议在真实的计算机上安装一个 Linux 操作系统，这个系统作为工作机，主要进行编译、构建和开发，另外辅助提供做一些实验及阅读源代码等。理论上使用哪家的发行版或者哪个版本都可以，但是为了避免意外的麻烦，建议使用和笔者相同的环境。在写作这本书的最后，笔者使用 Ubuntu12.10 将构建过程全部验证了一遍，所以建议读者也使用这个版本。

另外，我们当然不希望使用工作机调试我们构建的操作系统，因为这样需要频繁的启动。所以我们需要一个虚拟机，笔者使用的虚拟机是 VirtualBox。在 Ubuntu12.10 下，使用如下命令安装 VirtualBox：

```
root@baisheng:~# apt-get install virtualbox
```

因为我们是从零开始构建系统，因此虚拟机上还需要一个额外的 Linux 系统作为

桥梁。鉴于其只是一个桥梁，所以使用什么版本没有关系，比如笔者虚拟机上使用的是 Ubuntu11.10。

1.2 创建虚拟计算机

在安装 Linux 操作系统之前，我们需要从硬件层面创建一个虚拟的计算机。VirtualBox 启动后，主界面如图 1-1 所示。

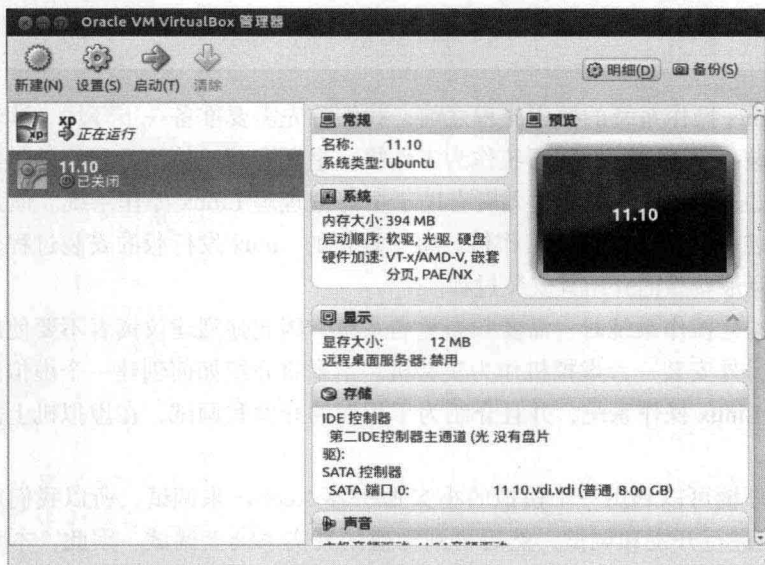


图 1-1 VirtualBox 主界面

单击图 1-1 中 VirtualBox 主界面工具条中的“新建”按钮，新建虚拟机的向导将启动。这个过程非常简单，读者按照新建向导一路执行下去就好。读者只需要注意在安装过程中要选择安装 Linux 操作系统，其他全部默认即可。

创建好虚拟机后，在 VirtualBox 主界面中将出现新建的虚拟裸机，如图 1-2 所示，其中，ubuntu11.10 就是笔者新创建的虚拟机。

1.3 安装 Linux 系统

本节我们将在 1.2 节创建的裸机上安装 Linux 操作系统。

在图 1-2 所示的工具栏上单击“设置”按钮，当然要确保在左侧的列表中选中的是刚刚创建的裸机，出现如图 1-3 所示的界面。

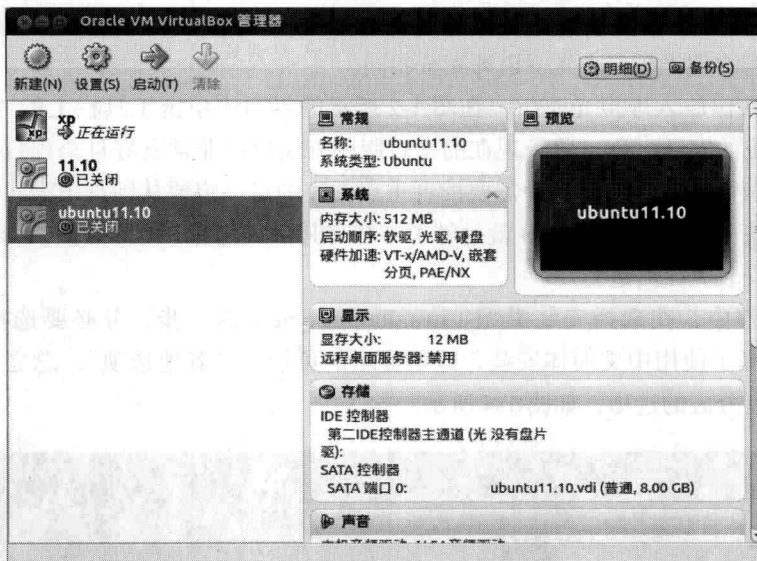


图 1-2 新建的虚拟裸机

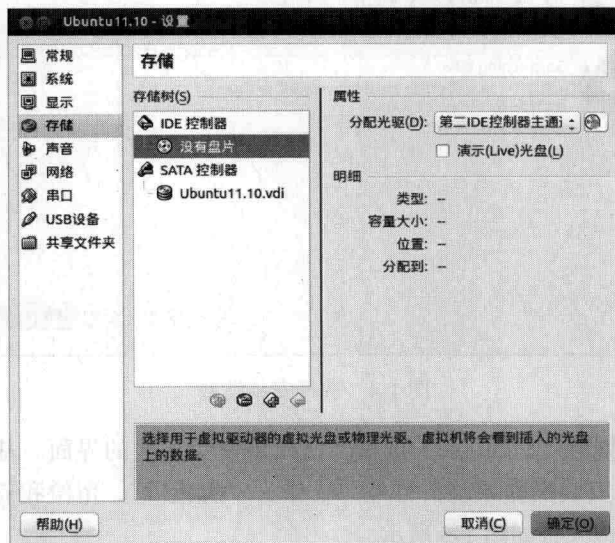


图 1-3 载入虚拟光盘映像

在图 1-3 中，首先在左侧的列表中选择“存储”。在默认情况下，我们会看到虚拟机已经添加了一个空的虚拟光驱。如果 VirtualBox 没有自动添加，读者手动添加即可。至于 SATA 接口还是 IDE 接口，是没有关系的，毕竟是虚拟的。然后，选中虚拟光驱，即图 1-3 所示的 IDE 控制器下的“没有盘片”，然后单击“分配光驱”文本框旁的带有光盘图片的按