

<<UNIX到Linux的移植>>

图书基本信息

书名：<<UNIX到Linux的移植>>

13位ISBN编号：9787111209119

10位ISBN编号：7111209117

出版时间：2007-4

出版时间：机械工业

作者：门多萨

页数：450

译者：张昕

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<UNIX到Linux的移植>>

内容概要

本书讲述怎样把UNIX环境下的应用程序移植到Linux环境上运行，是一本综合的开发和解决问题的参考手册。

本书详细描述了当前IT行业中被广泛应用的各种UNIX操作系统间的区别，其内容包括项目移植思考、调查、分析、移植Solaris应用程序、移植AIX应用程序、移植HP-UX应用程序、测试和调试。

它的读者对象包括开发人员、应用程序架构师以及项目经理。

本书的内容包括： 理解Linux环境：GNU二进制工具、Java环境、shell、打包工具等。

 揭示并确定项目的未知因素、可变因素及其他风险。

 处理具体平台的区别：标准、编译器、链接器、版本化、系统/库调用、线程等。

 使用GNU调试器及Linux内存泄露和性能跟踪工具测试和调试移植后的应用程序。

 包含了UNIX和Linux上API、编译器和链接器选项的快速索引，以及IBM POWER结构特有的移植问题。

<<UNIX到Linux的移植>>

作者简介

本书的三位作者是IBM最有经验的Linux移植专家，他们将带领你走完整个项目移植过程：调查、分析、移植和测试。

他们给出了一个从开始到完成的移植方法，对关键任务的切实讨论，以及评估新项目时用到的问卷。按照本书的方法，你可以发现Linux在API、库函数、版本化、系统特性和

<<UNIX到Linux的移植>>

书籍目录

1 移植项目思考 191.1 软件程序商业过程 191.2 移植过程 201.2.1 调查 201.2.2 分析 211.2.3 移植
 221.2.5 支持 221.3 定义项目范围和目标 221.4 估计 241.4.1 技能水平和移植经验 241.4.2 编译
 软件和中间件的可用性 251.4.4 编译环境和工具 251.4.5 平台依赖的结构 261.4.6 平台/硬件依赖的代
 码 261.4.7 搭建测试环境 261.4.8 用户接口需求 271.5 创建项目移植进度表 281.6 从商业角度看移植
 带注解的技术调查问卷样例 291.7.1 平台相关的内容 291.7.2 应用程序相关的内容 291.7.3 数据库内容
 项目移植时间进度内容 321.7.5 测试相关的内容 321.7.6 项目移植的执行内容 331.8 小结 332 调查
 器 342.1.1 GNU gcc C编译器 342.1.2 GNU g++ (C++)编译器 352.1.3 GNU g77 FORTRAN编译器 362
 Java编译器 (注释8) 372.1.5 GNU COBOL 372.1.6 其他编译器 372.2 Java环境 382.3 编译环
 境 382.3.1 gmake 382.3.2 GNU二进制工具 (binutils) 392.3.2.1 GNU ld 392.3.3 GNU as 392.3.4
 环境 412.4 Linux Shell 412.5 打包 422.5.1 用RPM打包应用程序 422.5.2 RPM的基本用法 432.5.2.1 r
 -e 432.5.2.3 rpm -U 442.5.2.4 rpm -q 442.5.2.5 rpm -V 442.5.3 LSB (Linux Standard Base) 推荐的基准
 式 442.6 项目管理任务 442.7 小结 463 分析 473.1 Linux标准 473.2 GNU libc库 483.2.1 glibc
 科学库 503.4 共享库 513.5 库版本化 (注释8, 9) 513.5.1 外部库版本化 513.5.2 符号版本化 533
 (运行时链接器) 563.6.1 编程接口 573.6.2 延迟重定位 (Lazy Relocation) 573.6.3 初始化 (initializing)
 终止化 (finalizing) 函数 583.7 系统调用 583.8 大页面支持 593.9 Native POSIX线程库 (Native POSIX
 Library, 简写作NPTL) 603.9.1 最大线程数 613.10 国际化 (I18N) (注释18)和本地化 613.10.1 iconv
 支持 633.10.2 如何创建消息目录 (message catalog) (注释19) 633.11 大小端 (Big/Little-Endian, 也叫字节
 序) 环境 643.12 从32位移植到64位 653.12.1 常见的移植错误 663.12.2 最优方法 673.13 小结 674
 程序 684.1 准备工作 684.1.1 搭建移植系统 684.1.2 搭建代码修改管理系统 694.1.3 搭建编译系统
 区别 694.2.1 C语言 704.2.2 C++语言 704.2.3 GNU C/C++语言 704.2.4 Java语言 714.2.5 线程
 之间的比较 714.3 编译器 724.3.1 C编译器 724.3.2 C编译器选项 724.3.3 C++编译器选项 824.3.4
 板实例化 924.3.5 编译32位或64位目标文件 924.3.6 编译器的优化和性能选项 924.3.7 Linux平台上的命
 令选择 934.4 链接器 944.4.1 输出映射 994.4.2 链接器工具 994.4.3 共享库 1004.4.4 库版本化 1
 (运行时链接器) 1014.5 移植shell脚本到Linux 1014.6 国际化和本地化 1024.7 Make 1024.7.1 为C-
 的Makefile变量 1044.7.2 库依赖 1044.7.3 空规则 1044.7.4 当前目标 1044.7.5 SCCS和RCS文件 105
 给宏赋值 1054.7.7 模式替换宏的引用 1054.7.8 VPATH 1054.7.9 命令执行 1064.7.10 特殊目标 106
 量 1074.7.12 退出状态 1074.8 调试器 1084.9 线程 1084.9.1 Solaris libthread与Linux libpthread 1084.9
 libpthread与Linux libpthread 1104.9.3 取消点 (Cancellation Points) 1134.9.4 C++异常和Linux线程 1144.1
 号 1154.10.1 信号动作 (Actions) 1164.10.2 简单信号 1164.10.3 Linux的信号支持 1174.10.4 POSIX
 数 1194.10.5 信号集 1204.10.6 信号信息 1204.10.7 kill() 1224.11 系统调
 用 1224.11.1 open() 1224.11.2 fcntl() 1244.11.3 dirent 1264.11.4 CPU亲和性 (Affinity) 1274.11.5
 库 1274.13 Solaris和Linux API 1284.13.1 内存管理 1284.13.2 进程间通信 (Interprocess Communication,
) 1294.13.3 内存布置优化 (Memory Placement Optimization, 简写作MPO
) 1344.13.4 vfstab 1344.13.5 posix_spawn()和posix_spawnp() 1344.13.6 plock() 1354.13.7 waitpid()
 库 1364.15 大页面支持 1364.16 Solaris上的一些不可移植的编程用法 1374.17 小结 1375 移植AIX应
 用 1395.1 准备工作 1395.1.1 搭建移植系统 1395.1.2 搭建代码修改管理系统 1405.1.3 搭建编译系统
 的区别 1405.2.1 C语言 1415.2.2 C++语言 1415.2.3 GNU C/C++语言 1415.2.4 Java语言 1415.2.5
 支持的标准之间的比较 1425.3 编译器 1435.3.1 调用编译器 1445.4 Make 1535.4.1 gmake条件分支语句
 器 1545.5.1 链接器的不兼容性 1545.5.2 链接器:从AIX移植到Linux参考手册 1565.6 AIX和Linux创建
 的比较 1575.7 动态装载 1575.8 AIX和Linux上模板实例化的区别 1585.8.1 使用#pragma
 或#include 1595.8.2 GNU中其他可用的模板选项 1605.9 C++filt工具 1605.10 系统API
 : AIX/Linux 1615.10.1 AIX/Linux I/O系统API 1615.11 部分AIX/Linux系统API之间的比较 1675.11.1 get
 getfsfile(), getfstype(), getfsspec() 1675.11.2 ioctl() 1685.11.3 read()
 , write() 1695.11.4 confstr() 1715.11.5 opendir() 1715.11.6 readdir() 1725.11.7 fcntl() 1725.11.8 llse
 lseek64() 1775.11.9 uname() 1785.11.10 syslog(), closelog(), openlog() 1795.11.11 swapoff(),

<<UNIX到Linux的移植>>

swapon() 1795.11.12 acct() 1805.11.13 mmap(), mmap64()(注释17),
 munmap() 1815.11.14 pread(), pwrite() 1845.11.15 stat(), lstat(), fstat() 1855.11.16 ptrace() 1855.11.17 set
 setregid() 1875.11.18 sync() 1885.11.19 wait3(), wait4() 1895.11.20 getcwd() 1905.11.21 mount(), umoun
 vmount() 1915.11.22 readv(), writev() 1945.11.23 select() 1955.11.24 reboot() 1965.11.25 chroot() 198
 , statfs() 1995.11.27 poll() 2005.11.28 quotactl() 2015.12 AIX上的一些非标准的编程用法 2045.13 信
 信号动作 2055.13.2 简单信号 2055.14 网络编程 2105.15 进程间通信 (Interprocess Communication, 简
 作IPC) 2115.15.1 信号量 2115.15.2 系统V信号量集 2135.15.3 消息队列 (mqqueue.h) 2135.15.4 共
 存 2155.16 POSIX线程 2175.16.1 Linux和AIX的线程模型比较 2175.16.2 内核空间线程 2175.16.3 用
 程 2185.16.4 AIX和Linux POSIX线程交叉对比 2185.17 大页面支持 2215.18 通用API 2215.19 小结
 植HP-UX应用程序 2246.1 准备工作 2246.1.1 搭建移植系统 2246.1.2 搭建代码修改管理系统 2256.1.1
 环境 2256.2 常见的区别 2256.2.1 C语言 2266.2.2 C++语言 2266.2.3 GNU C/C++语言 2266.2.4 Ja
 程 2266.2.6 所支持标准之间的比较 2276.3 编译器 2286.3.1 C编译器选项 2296.3.2 C++编译器选项
 g++的模板实例化 2356.3.4 编译32/64位二进制程序 2366.3.5 编译器优化和性能选项 2366.3.6 使用GN
 移植和编译 2386.3.7 Linux平台上的命名空间选择 2386.4 链接器 (注释21) 2396.4.1 输出映射 (exp
) 2406.4.2 链接器工具 2406.5 库版本化 2416.5.1 HP-UX上的库版本化 2416.5.2 Linux上的库版本化
) 2426.6 动态链接和共享库 2426.6.1 GNU libtool 2446.7 移植shell脚本到Linux 2446.8 国际化 (118
 释26) 2456.8.1 iconv支持和消息目录 (message catalog) 2476.9 软件开发工具 2476.9.1 Makefile语法
 调试器 2506.10 线程 2506.10.1 HP-UX和Linux上的线程支持 2516.10.2 HP-UX DCE和POSIX线程的区
 别 2546.10.3 调试Linux线程 2566.10.4 C++异常和Linux线程 2586.10.5 链接线程安全的库 2606.11 作
 信号动作 (Acitions) 2606.11.2 简单信号 2616.11.3 信号发送 2626.11.4 Linux的信号支持 2626.11.5
 捕获函数 2636.11.6 信号信息 2636.11.7 信号缺省动作 2656.12 HP-UX上的系统调用和Linux上的对应
 用 2676.12.1 检查POSIX API支持 2676.12.2 检查Linux内核版本 2716.13 系统库 2726.14 GNU科学
 库 2736.15 HP-UX和Linux API 2746.15.1 glibc版本 2746.15.2 glibc遵循的标准 2746.15.3 HP-UX上的
 植的编程用法 2756.16 进程间通信 2756.16.1 管道 2756.16.2 FIFO 2766.16.3 POSIX消息队列 2766.
 号量 2776.16.5 POSIX共享内存 2786.16.6 系统V共享内存 2786.16.7 系统V消息 2796.16.8 系统V信
 小结 2807 测试和调试 2817.1 使用GNU调试器 2827.1.1 编译可以使用GDB的程序 2827.1.2 调
 用GDB 2827.1.3 GDB基本命令 2837.1.4 调试多进程 2837.1.5 调试多线程应用程序 2847.1.6 停止多
 序 2857.1.7 GDB初始化文件 2867.1.8 GDB和信号 2867.1.9 GDB的图形化前端 (front end) 2887.2
 和ltrace 2887.3 cscope工具 2887.3.1 Vim和cscope 2897.4 glibc调试支持 2907.5 内存调试工具 2917.
 Fence 2917.5.2 GNU/Linux mtrace, mcheck,
 MALLOC_CHECK 2927.5.3 Mpatrol 2927.5.4 dmalloc 2927.5.5 valgrind 2937.5.6 Rational
 Purefy 2937.5.7 Insure++ 2947.6 静态程序检查器 2947.6.1 Splint 2947.6.2 CodeSonar 2947.7 线程
 能监测工具 2957.8.1 内部视角 2957.8.2 外部视角 2997.8.3 其他工具 3027.9 小结 303F.3 编译和链
 》670 控制和优化栈增长的编译器选项 415F.3.2 汇编代码 415F.3.3 栈框架结构和链接 415F.3.4 预先定义的
 符号 416F.3.5 调试 416F.4 字节序 416F.5 其他区别 416F.5.1 ASCII和EBCDIC问题 416F.5.2 cpuinfo的格
 式 417F.6 变参列表 417F.7 从哪儿可以获取更多信息 417F.8 致谢 417G.1 打包任务 418G.2 安装和升级任
 务 418G.3 用户管理任务 420G.4 网络管理和配置 420G.5 NFS管理和配置 421G.6 管理系统资源 422G.7
 和文件系统管理 422G.8 交换分区管理 423G.9 逻辑卷管理 423G.10 常用的解决错误的方法 425G.11 网络
 题解决方法 427G.12 配置文件及其他文件 427G.13 可比较的命令 428

<<UNIX到Linux的移植>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>