



清松电脑系列丛书



SCO OpenServer 5 实用大全系列丛书之六

SCO OpenServer 程序员技术精粹

李祥凯 魏 欧 程 波 等编译
孟庆昌 审校



清华大学出版社

SCO OpenServer 5 实用大全系列丛书之六

SCO OpenServer

程序员技术精粹

李祥凯 魏 欧 编译
程 波 吴 健
孟庆昌 审校

清华大学出版社

05

(京)新登字 158 号

内 容 简 介

本书主要讲述与程序员有关的一些技术资料。书中对浮点数操作、C 标准依从性、开发国际化软件、公共目标文件格式(COFF)、ELF 目标文件等内容进行了介绍,并且列举了许多实例来进一步说明。本书内容详实,对软件开发人员充分利用系统性能十分有益。

本书的读者对象为使用 SCO OpenServer 开发系统及其它 UNIX 系统开发应用软件的开发人员,以及计算机专业的大学高年级学生及研究生。

版权所有,翻印必究。本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

SCO OpenServer 程序员技术精粹/吴健等编译. —北京:清华大学出版社,1998. 12
ISBN 7-302-03220-3

I. S… II. 吴… III. 操作系统, SCO OpenServer-技术开发 IV. TP316

中国版本图书馆 CIP 数据核字(98)第 34510 号

出 版 者: 清华大学出版社(北京 清华大学校内, 邮政编码: 100084)

因特网址: <http://www.tup.tsinghua.edu.cn>

印 刷 者: 北京市清华园胶印厂

发 行 者: 新华书店总店北京科技发行所

开 本: 787×1092 1/16 印张: 13.75 字数: 326 千字

版 次: 1998 年 12 月第 1 版 1999 年 3 月第 2 次印刷

书 号: ISBN 7-302-03220-3/TP·1723

印 数: 2001—4000

定 价: 88.00 元

前 言

UNIX 系统自 1969 年踏入计算机世界以来已近 30 年。虽然目前市场上面临某种操作系统(如 Windows NT)强有力的竞争,但是它仍然是笔记本电脑、PC、PC 服务器、中小型机、工作站、大巨型机及群集、SMP 和 MPP 上通用的操作系统,至少到目前为止还没有哪一种操作系统可以担此重任。而且以其为基础形成的开放系统标准(如 POSIX)也是迄今为止唯一的操作系统标准,即使是其竞争对手或者目前还尚存的专用硬件系统(某些公司的中大型机或专用硬件)上运行的操作系统,其界面也是遵循 POSIX 或其它类 UNIX 标准的。从此意义上讲,UNIX 就不只是一种操作系统的专用名称,而成了当前开放系统的代名词。君不见许多公司“开放系统”的旗帜不是当 UNIX 在世界上流行并形成了开放的潮流之后,自愿或不自愿地举起来的么?就此一点 UNIX 就功不可没。我们在 80 年代初就说过下列的话:UNIX 这一名字可能会在电脑世界消失,但其精神永存。今日看起来正是这样。

在技术不断进步、产品日益丰富、市场激烈竞争的今天,我们每一个人都希望快速而有效地掌握最新技术并应用到实际工作中去。而科学技术知识浩如烟海,人们总是根据需要有所选择,学习 UNIX 系统也应如此。

除了一些特定的群集、SMP 和 MPP 上的大型系统之外,其余硬件平台上使用的 UNIX 系统目前主要分两大部分。一部分是各硬件厂家根据 UNIX 基本系统和相关标准开发并配备在不同平台上的系统,其代表有 IBM 的 AIX、HP 的 UX、NCR 的 RAS、SUN 的 Solaris 等;另一部分是专业软件厂家提供的系统,其代表有 SCO UNIX。SCO UNIX 是目前在 Intel 芯片构成的硬件平台上运行得最为普遍的 UNIX。AT&T 在 1989 年推出了 UNIX 系统 V 第 4 版(SVR4),之后 AT&T 把 UNIX 系统及其商标卖给了 Novell,Novell 以此为基础开发了 UnixWare。之后,Novell 又把 UnixWare 卖给了 SCO,SCO 以此为基础结合其以前的 XENIX 和 UNIX 版本于 1997 年开发了新的 UNIX 系统 V 第 5 版(SVR5)。

综合了以前的 OpenServer 和 UnixWare,SCO 推出了 OpenServer 5.0 版。所以 OpenServer 5.0 是目前为止在市场上最为流行的也是最新的商品系统。由于以 Intel 芯片为 CPU 的 PC、PC 服务器,笔记本电脑等在市场上极为普及,因此学习以这些电脑为载体的 SCO OpenServer 5.0 就成了掌握 UNIX 系统知识最便捷的途径。

本丛书的编译者为中国科学院软件研究所的有关专家学者。中国科学院软件研究所在 1979 年曾率先在国内引进了 UNIX 系统,近 20 年来基于 UNIX 作了大量的研究开发工作。

本丛书在 SCO 公司提供的技术资料的基础上,根据编译者的实践经验,增加了部分新内容。全套丛书共有六本:

丛书之一《SCO OpenServer 系统手册》重点在系统安装及配置、系统的基本操作(如启停系统)以及硬件设备。

丛书之二《SCO OpenServer 用户指南》向用户全面介绍系统的使用,特别是 shell 命令及其编程,系统命令使用及一般工作的使用。

丛书之三《SCO OpenServer 系统管理指南》侧重于系统管理,这只有在学习了丛书之一和之二后,对系统有一个完整的了解并懂得使用之后,才能有条件学习该书内容,也只有在掌握了这部分知识之后才有能力管理好系统。

丛书之四《SCO OpenServer 网络指南》侧重于 SCO 网络工具的讲述,目的是帮助读者访问各种网络服务。利用这些工作 SCO 用户可将 OpenServer 系统与各种相似或不同的硬件平台,如专用微机、工作站、运行 DOS 或 OS/2 的 PC 机以及软件平台,如其它 UNIX 系统连接。

丛书之五《SCO OpenServer 开发系统编程工具指南》则不像前面几本书只要求读者学会使用和管理系统,而是对读者提出了更高的要求:利用开发工具和系统来编程、调试和组装构成系统。利用 UNIX 系统的各种编程工具进行系统开发是 UNIX 系统开发人员必须掌握的知识,而且掌握越多就越“自由”,而本书介绍的这些工具是 UNIX 系统最重要、最有使用价值的,读者应不断深入掌握这些知识,使自己成为真正的 UNIX 行家里手。

丛书之六《SCO OpenServer 程序员技术精粹》主要是在丛书之五基础上对程序员提出的更高要求。书中对 SCO OpenServer 新增加的一些特征进行了说明,并辅以实例。对于程序员来说掌握这些知识对于其在 UNIX 系统上的开发会有更多的好处。

这套丛书涉及的内容很多,读者可以根据需要选读或通读全套丛书中的几本或一本中的几章、几节。希望这套丛书的出版对读者掌握 UNIX,特别是 SCO UNIX 有所帮助。

由于资料量大,参与编译的人员手头又有许多开发任务,编译出这套丛书工作量不小,为了尽快出版大家加班加点,急切之中不妥之处在所难免,敬请广大读者不吝赐教。

编 译 者

1998 年 4 月 29 日

序

UNIX 是当前世界上使用普遍、影响深远的主流操作系统。在 UNIX 世界中,SCO UNIX 一直以其植根于 Intel 平台,接近于大众的特色而名满天下。这其中最新的产品就是 SCO 的 OpenServer 系列,也就是摆在您面前的这一套丛书想要向您介绍的内容。根据国际数据集团(IDC)的统计,在 1996 年,SCO UNIX 在全球 UNIX 市场占有 35% 的市场。1997 年,SCO UNIX 的市场占有率提高到 40%。这其中大多数的产品是 SCO OpenServer。

UNIX 从诞生到现在已经有了近 30 年的历史。在 30 年的风风雨雨中,UNIX 有过它的徘徊,也有过它的辉煌。到现在 20 世纪行将结束的时候,UNIX 再次面临市场的考验。来自于其他操作系统产品的竞争使人们产生了许多疑问和误解,我们用个时髦词称之为“误区”。在您开始阅读这套丛书之前,我认为有必要把这些误区澄清一下,以免在您学习的过程中还要考虑我为什么要学习 UNIX。

误区之一: UNIX 只适用于大中型核心系统

虽然在群集、SMP、MPP 领域 UNIX 是基本的甚至是唯一的操作系统,但是 UNIX 也同时是唯一可以运行在笔记本电脑、PC、PC 服务器、小型机乃至大型机上的操作系统。从以上 IDC 的统计数字可以看出,有四成的 UNIX 系统是 SCO UNIX,而 SCO UNIX 是只能运行在 Intel 平台上的。加上其他种类的 UNIX,可以说,世界上一大半 UNIX 系统是运行在 PC 上的。而今天的 SCO UNIX 也同样具备了支持群集和 SMP 的能力。

UNIX 可支持多用户核心系统,同时也支持分布式网络系统。对网络的支持是 UNIX 能够长期发展的关键所在。在那句“网络就是计算机”的名言背后,UNIX 功不可没。因为有了 UNIX,才有了 TCP/IP,才有了 Internet/Intranet。掌握了 UNIX,你才能更深刻地理解它们的精髓。

误区之二: UNIX 与我们的日常生活无关

UNIX 虽然不像 Windows,在每个办公室都可以见到。然而你的生活是离不开 UNIX 的。就我国而言,目前大部分关键性的业务是运行在 UNIX 平台上的。这其中包括银行、保险、税务、邮政和电信、铁道、海关、气象及政府机构等等。所有这些都在你的身边。相信许多了解 SCO UNIX 的人对此都深有体会。大部分关键应用部门的高层技术人员和管理人员都懂 UNIX 甚至是 UNIX 专家。如果您希望自己成为一位合格的信息主管(CIO),UNIX 恐怕是您的必修课程。

误区之三: UNIX 是阳春白雪,很难学且学而无用

UNIX 的技术含量很高,专家们说,当今计算机软件技术中的大部分技术都可以从 UNIX 中找到它们的来源和影子。正因为如此,UNIX 才有如此的高性能、可靠性、稳定性、

安全性、互操作性、可用性、可伸缩性等等。也正因为如此,UNIX 要学习的东西比其它的多。同时,也要承认由于历史的原因,UNIX 的用户界面不够友好。但是这种状况已经大大改观了。从这套丛书中您就可以看到,SCO 的 OpenServer 5 系统的界面已经变成非常友好了。学了 UNIX,你会感觉到你走入了一个新的境界,这种境界不是那些只满足于对软件技术浅尝辄止、一知半解的人所能够达到的。

还没有见到过学了 UNIX 而无用武之处的。见到的都是对 UNIX 感叹“书到用时方恨少”的。甚至没有见到买了 UNIX 系统而搁置不用的,虽然这种情况对于其它产品并不少见。如果你留心报上招聘软件工程师的广告,你会发现多少不需要懂 UNIX 的?

误区之四: UNIX 没有发展

UNIX 自从诞生那天起就没有停止过发展。我们只看最近几年的发展就能明了。

1995 年,新标准 UNIX'95 问世

1996 年,两大 UNIX 集团 X/Open 和 OSF 合并成为“开放集团”

1997 年,新的 UNIX 核心 SVR5(UNIX 系统 5,版本 5)在 SCO 完成

1998 年,基于 SVR5 的综合了 OpenServer 和 UnixWare 的 UnixWare 7 版本由 SCO 推出并计划推出 OpenServer 5.0.5 版本。

这些新标准的产品发展和更多新的技术融合,使得它们能为客户提供更强有力的功能和性能,对客户更友好,与 Internet/Intranet 结合的更紧密。这也是为什么 UNIX 仍然被 Intel 所钟爱,被广大软硬件厂商包括 IBM,HP,Compaq,SUN,NCR 等作为标准操作系统平台的原因。

误区之五: UNIX 将被其它操作系统所取代

如果我们谈一个产品或者一种技术是否将被其它的产品或技术取代,或者直截了当地说它是否会消亡,那么我想必须有两个前提:其一是这种技术或产品已经不能适应市场的需要,将被市场淘汰;其二是有一种比它更好、更先进的技术或产品问世。但是认真研究 UNIX,我们没有发现上述两个前提的存在。不但如此,UNIX 的技术到目前为止仍然是没有任何一种操作系统可以与之比拟的,而且基于 UNIX 的大量的应用是一笔不可估量的财富。UNIX 将会在相当长的时间里与其它操作系统共存。既然两年前谈 UNIX 会消亡的“预言家”面对 UNIX 目前的发展已经噤若寒蝉,那么现在预言 UNIX 在不久将被取代的惊世骇俗之言也同样会在历史面前成为茶余饭后的笑料。

现在您是不是对学好这套书很有信心了?希望如此。

美国 SCO 公司中国区总经理



一九九八年春

目 录

关于本书.....	1
符号约定.....	1
相关文档.....	2
其它参考资料.....	3
第 1 章 浮点运算	5
1.1 IEEE 算术运算	5
1.1.1 单精度.....	5
1.1.2 双精度.....	5
1.1.3 扩展精度.....	6
1.1.4 规格化数.....	6
1.1.5 非规格化数.....	7
1.1.6 最大和最小可表示的浮点数的值.....	7
1.1.7 特殊情况的取值.....	7
1.1.8 NaN 和无穷大	8
1.1.9 舍入控制.....	8
1.1.10 异常、粘着位(sticky bits)以及陷阱位(trap bits)	9
1.2 单精度浮点运算.....	10
1.3 扩展双精度类型.....	11
1.4 IEEE 要求	11
1.4.1 浮点格式与整型格式转换	11
1.4.2 平方根	12
1.4.3 比较和不可排序条件	12
1.4.4 在 input/output 中的 NaN 和无穷大	12
第 2 章 符合标准 C 语言	13
2.1 新旧风格函数的混合使用.....	13
2.1.1 编写新的代码	13
2.1.2 升级旧代码	14
2.1.3 综合考虑	14
2.1.4 几个例子	15
2.2 变实参函数.....	16

2.2.1	示例	16
2.3	类型转换: unsigned 及其值的保留	18
2.3.1	背景	18
2.3.2	编译操作	18
2.3.3	示例一: 使用常量	19
2.3.4	位字段	20
2.3.5	示例二: 结果相同	20
2.3.6	整型常量	20
2.3.7	示例三: 整型常量	21
2.4	单词化及预处理	21
2.4.1	ANSI C 编译阶段	21
2.4.2	旧版本中 C 的编译阶段	22
2.4.3	逻辑源行	23
2.4.4	宏替换	23
2.4.5	构成字符串	23
2.4.6	单词检查	24
2.5	使用 const 和 volatile 类型限定符	24
2.5.1	用于左值的类型	24
2.5.2	派生类型中的类型限定符	25
2.5.3	使用 const 类型读取字符的值	26
2.5.4	使用 volatile 类型	26
2.6	多字节字符和宽字符	27
2.6.1	“亚洲化”就意味着多字节字符	27
2.6.2	编码的特点	27
2.6.3	宽字符	28
2.6.4	转换函数	28
2.6.5	C 语言特性	28
2.7	标准前导文件和保留字	29
2.7.1	发展过程	29
2.7.2	标准前导文件	29
2.7.3	供应用程序使用的保留名	30
2.7.4	用于扩充而保留的名称	30
2.7.5	正确使用名称	31
2.8	国际化	31
2.8.1	本地环境	31
2.8.2	setlocale() 函数	31
2.8.3	被改变的函数	32
2.8.4	新函数	33

2.9	表达式的分组及求值	33
2.9.1	定义	34
2.9.2	Kernighan 和 Ritchie C 重新安排许可权	34
2.9.3	ANSI C 规则	34
2.9.4	括弧分组与求值	35
2.9.5	“as if”规则	35
2.10	不完整类型	35
2.10.1	概述	35
2.10.2	不完整类型的完整化	36
2.10.3	声明	36
2.10.4	表达式	36
2.10.5	判断	36
2.10.6	例子	37
2.11	兼容类型和混合类型	37
2.11.1	多重声明	38
2.11.2	不同编译系统的兼容性	38
2.11.3	同一编译系统的兼容性	38
2.11.4	兼容的指针类型	38
2.11.5	兼容的数组类型	39
2.11.6	兼容的函数类型	39
2.11.7	特殊情况	39
2.11.8	混合类型	39
第 3 章	建立和使用消息目录	40
3.1	建立消息目录	40
3.2	使用消息目录的源程序示例	41
3.3	消息目录与应用程序的结合	42
3.4	运行应用程序	43
第 4 章	使用受托系统设施	44
4.1	使用受保护子系统	44
4.2	受托系统使用指南	45
4.3	编程示例	46
4.4	subsys.c 示例	46
4.5	prwarn.c 示例	50
4.6	loge.c 示例	62

第 5 章 使用事件管理程序 API	76
5.1 操作及语义概述.....	76
5.2 事件设备.....	77
5.3 使用事件队列.....	77
5.4 事件管理程序调用.....	78
5.5 示例.....	78
第 6 章 ksh 下的作业控制	83
第 7 章 公共目标文件格式(COFF)	86
7.1 定义与规范.....	87
7.1.1 节	87
7.1.2 物理地址与虚拟地址	87
7.1.3 目标机	87
7.2 文件头.....	88
7.2.1 幻数	88
7.2.2 标志位	88
7.2.3 文件头说明	88
7.3 可选头信息.....	89
7.3.1 标准 UNIX 系统的 <i>a.out</i> 头	89
7.3.2 可选头说明	90
7.4 节头.....	90
7.4.1 标志	91
7.4.2 节头声明	91
7.4.3 <i>.bss</i> 节头	92
7.5 节.....	92
7.6 重定位信息.....	92
7.6.1 重定位信息项的声明	93
7.7 行编号.....	93
7.7.1 行编号声明	94
7.8 符号表.....	94
7.8.1 专用符号	95
7.8.2 内层程序块	96
7.8.3 符号和函数	97
7.8.4 符号表项	98
7.8.5 符号名	98
7.8.6 存储类	98
7.8.7 专用符号的存储类	99

7.8.8	符号值域.....	100
7.8.9	节号域.....	101
7.8.10	节号和存储类	102
7.8.11	类型字段	102
7.8.12	类型域和存储类	104
7.8.13	符号表项的结构	105
7.8.14	辅助表项	105
7.8.15	文件名	106
7.8.16	节	106
7.8.17	标记名	107
7.8.18	结构的结束符	107
7.8.19	函数	107
7.8.20	数组	108
7.8.21	程序块和函数的结束符	108
7.8.22	程序块和函数的起始符	108
7.8.23	与结构、联合和枚举相关的名称.....	109
7.8.24	辅助表项的声明	109
7.9	字符串表	110
7.10	访问例程.....	111
第 8 章	ELF 目标文件	112
8.1	文件格式	112
8.1.1	文件格式.....	112
8.1.2	数据表示.....	113
8.2	程序链接	114
8.2.1	ELF 头	114
8.2.2	节头.....	118
8.2.3	字符串表.....	124
8.2.4	符号表.....	125
8.2.5	重定位.....	128
8.3	程序执行	130
8.3.1	程序头.....	130
8.3.2	程序加载(与特定处理器相关).....	135
8.3.3	程序解释程序.....	137
8.3.4	动态链接程序.....	137
第 9 章	增强的汇编接口	144
9.1	术语定义	144

9.2	示例	145
9.2.1	定义	145
9.2.2	使用	145
9.3	使用汇编宏	146
9.3.1	宏定义	146
9.4	编写 asm 宏	148
第 10 章	映射文件选项	149
10.1	使用映射文件选项	149
10.1.1	映射文件结构和语法	150
10.1.2	段声明	150
10.1.3	映射指令	152
10.1.4	尺寸符号声明	153
10.2	映射示例	153
10.3	映射文件默认设置	154
10.4	内部映射结构	155
10.5	出错消息	157
10.5.1	警告	157
10.5.2	致命错误	158
第 11 章	Desktop 集成	159
11.1	背景知识	159
11.2	图形接口标准和库	160
11.3	向下兼容	160
11.4	将图标放在 Desktop 上	161
11.4.1	创建图标像素文件	161
11.4.2	定义图标触发器	163
11.4.3	安装图标文件	166
11.4.4	动画图标	168
11.4.5	本地化图标标签和消息	169
11.4.6	为用户数据文件创建图标	171
11.5	运行时调整显示分辨率	173
11.6	支持 Desktop 颜色选择程序	175
11.6.1	定义与显示器有关的颜色	177
11.7	与会话管理程序进行通信	179
第 12 章	客户软件安装工具箱	182
12.1	创建可用 custom 安装的软件版本	182

12.1.1	关于软件存储对象(SSO)	183
12.1.2	确定产品的结构	184
12.1.3	确定 SSO 文件属性	185
12.1.4	建立版本树	186
12.1.5	建立 CDMT 的环境	186
12.1.6	创建 CDMT 输入文件	187
12.1.7	创建和构造 SSO 树	190
12.1.8	创建可用 custom 安装的文档	190
12.2	高级的软件安装	192
12.2.1	关于组件脚本	192
12.2.2	软件管理阶段	193
12.2.3	创建软件升级	201
12.2.4	SSO 路径映射 API	203
12.2.5	使用 SSO 路径映射 API	204

关 于 本 书

本书主要讲述与程序员有关的技术资料的一些专题。书中对 SCO 新增加的一些特征进行了说明,并且列举了许多实例来说明这些特征。

本书包括以下内容:

第 1 章,“浮点运算”讨论与硬件以及与软件版本有关的浮点运算。

第 2 章,“符合标准 C 语言”讨论与能产生符合 ANSI C 的代码有关的合适的编码方法

第 3 章,“建立和使用消息目录”讨论怎样利用消息目录进行软件国际化。

第 4 章,“采用受托系统设施”提供了三个运行于该方式下的程序实例。

第 5 章,“使用事件管理程序 API”讨论如何利用事件管理器 API 处理诸如鼠标(含基于光标的应用程序)的设备。

第 6 章,“ksh 下进行任务控制”讨论在 Korn Shell 下如何实现任务控制。

第 7 章,“公共目标文件格式(COFF)”定义了公共目标文件格式,并且重点讲述了与二进制 COFF 相联系的版本。

第 8 章,“ELF 目标文件”定义了扩充连接器格式(ELF)并且重点讲述了与二进制 ELF 相联系的版本。

第 9 章,“增强的汇编接口”讨论 SCO OpenServer™开发系统中可供使用的增强的汇编接口。

第 10 章,“映射文件选项”讨论用于连接器的映象文件控制存储器分配。

第 11 章,“Desktop 集成”描述如何集成应用程序进入开放的桌面环境。

第 12 章,“客户软件安装工具箱”讨论如何使用 Toolkit(CDMT)进行软件的分发以及如何创建安装的软件。

符号约定

这里列出了书中的一些命令、文件名、键和其它特殊符号:

例 子	含 义
lp 或 lp(C)	命令,设备驱动程序,程序和实用程序(名字、图标或窗口);圆括号中的字母表示介绍命令、驱动程序、程序或实用程序的参考手册。
<i>/new/client.list</i>	文件、目录和桌面(名字图标或窗口)
<i>root</i>	系统,网络或用户名
<i>filename</i>	用合适的名字或值替换
<Esc>	键盘上的键
Exit program?	系统输出(提示,信息)

yes 或 yes	用户输入
"Description"	字段名或列标题(在屏幕上或在数据库中)
Cancel	按钮名
Edit	菜单名
Copy	菜单项
File ◊ Find ◊ Text	菜单或菜单项序列
Open 或 open(S)	库程序、系统调用、核心函数、C 关键字;括号中的字母表示该文件所在的参考手册的章节。
\$ HOME	环境或 shell 变量
SIGHUP	命名的常量或信号
"adm30"	数据值
employees	数据库名
orders	数据库表
buf	C 程序结构
b-b.errno	结构成员

相关文档

SCO OpenServer™ 开发系统带有众多的文档,这些书通过联机方式和/或印刷方式提供。双击 Desktop 上 Help 图标就可在联机方式下阅读这些书。这些书还有相应的印刷版本。

SCO OpenServer 开发系统文档集的内容如下所示。除这些书之外,还有一本书叫《开发系统版本和安装注释》也随着开发系统软件一道提供。

	参 考 手 册	指 南
UNIX 应用	• C++ 库手册 • C++ 产品参考手册 • 程序员参考手册(2 卷)	• 选择阅读 C++ • 字符用户接口指南 • 开发人员专题 • 编程工具指南
网络	• 网络程序员指南和参考手册	
图形	• X 窗口系统程序员参考手册 • OSF/Motif 程序员参考手册	
任选	• 设备驱动程序编写者指南	

开发系统包括的书有:

《C++ 库手册》

介绍 C++ 类库和它们的使用。

《C++ 产品参考手册》

提供 SCO C++ 语言系统的 Release 3.1 所支持的 C++ 语言的完整定义。

《选择阅读 C++》

介绍 C++ 语言的情况。

《字符用户接口指南》

介绍开发用户接口可用的机制,并详细介绍了如何使用 **curses**(S)和扩展终端接口(ETI-Extended Terminal Interface),如何编制能够在 ASCII 终端上运行的用户界面。这本书包含大量 **curses** 程序示例。

《程序员技术精粹》

这是开发系统用户感兴趣的专题方面的技术论文集。这里面的许多论文都用例子来说明 SCO 软件提供的扩展性能的特点。

《网络程序员指南和参考手册》

本书介绍应用面向网络的通信协议(TCP/IP、IPX/SPX 和 NetBIOS)、开发应用程序时可用的库和应用程序编程接口(API)(Internet 域 socket, TLI, XTI, ONC, RPC 和 XDR)。书中还介绍了如何使用 UNIX 域 socket 进行进程间通信,如何使用 SMUX API 为 SNMP 代理程序创建同级 SMUX。本书还包括有关(NC), (NET), (NS), (SAPD), (SLIB)和(SSC)的参考指南。

《OSF/Motif 程序员参考手册》

本书由 OSF/Motif 工具箱,窗口管理程序,用户界面语言命令和函数组成。

《程序员参考手册》

本书分两卷来介绍“开发系统”,书中包括(CDMT), (CP), (FP), (S)等章节。

《编程工具指南》

书中介绍编程工具和在设计软件中的使用,包括开发系统提供的符合 ANSI 标准的 C 编译程序专用实现的详细介绍。书中还介绍了调试工具及如何使用它们跟踪、消除 C 和汇编语言程序中的问题,如何使用 **dbxtra**TM调试基于窗口环境的 OSF/Motif 中的 C 和 C++ 程序。

《版本和安装注释》

着重介绍此版本开发系统的新特点、产品变化、文档变化、已知的限制。书中也包含安装注释。

《X 窗口程序员参考手册》

书中提供 X Window SystemTM编程命令和函数。本书分为以下几个内容:X 库(XS)、X 工具箱(xt)、各种实用程序(Xmu)和 X 扩展(Xext)。

此外,如果需要下面这本书也可作为参考:

《设备驱动程序编写指南》

本书为编写 SCO 系统设备驱动程序提供参考和指导。该书可单独使用。

其它参考资料

有许多书和文章讨论如何开发 UNIX 操作系统上的软件。

我们特别推荐开发人员购买下列标准教科书:

《C 编程语言》,第 2 版,作者:Brian Kernighan 和 Dennis Ritchie

下列的书籍较全面地介绍 UNIX 操作系统编程:

- Donald Lewine 的《POSIX Programmer's Guide》