

# 目 录

第 1 章 关于本书的对话 .....	1	2.4 持久性 .....	9
第 2 章 操作系统介绍 .....	3	2.5 设计目标 .....	11
2.1 虚拟化 CPU .....	4	2.6 简单历史 .....	12
2.2 虚拟化内存 .....	6	2.7 小结 .....	15
2.3 并发 .....	7	参考资料 .....	15

## 第 1 部分 虚拟化

第 3 章 关于虚拟化的对话 .....	18	作业（测量） .....	47
第 4 章 抽象：进程 .....	19	第 7 章 进程调度：介绍 .....	48
4.1 抽象：进程 .....	20	7.1 工作负载假设 .....	48
4.2 进程 API .....	20	7.2 调度指标 .....	49
4.3 进程创建：更多细节 .....	21	7.3 先进先出（FIFO） .....	49
4.4 进程状态 .....	22	7.4 最短任务优先（SJF） .....	50
4.5 数据结构 .....	24	7.5 最短完成时间优先（STCF） .....	51
4.6 小结 .....	25	7.6 新度量指标：响应时间 .....	52
参考资料 .....	25	7.7 轮转 .....	52
作业 .....	26	7.8 结合 I/O .....	54
问题 .....	26	7.9 无法预知 .....	54
第 5 章 插叙：进程 API .....	28	7.10 小结 .....	55
5.1 fork() 系统调用 .....	28	参考资料 .....	55
5.2 wait() 系统调用 .....	29	作业 .....	56
5.3 最后是 exec() 系统调用 .....	30	问题 .....	56
5.4 为什么这样设计 API .....	32	第 8 章 调度：多级反馈队列 .....	57
5.5 其他 API .....	34	8.1 MLFQ：基本规则 .....	57
5.6 小结 .....	34	8.2 尝试 1：如何改变优先级 .....	58
参考资料 .....	34	8.3 尝试 2：提升优先级 .....	60
作业（编码） .....	35	8.4 尝试 3：更好的计时方式 .....	61
问题 .....	35	8.5 MLFQ 调优及其他问题 .....	61
第 6 章 机制：受限直接执行 .....	37	8.6 MLFQ：小结 .....	62
6.1 基本技巧：受限直接执行 .....	37	参考资料 .....	63
6.2 问题 1：受限制的操作 .....	38	作业 .....	64
6.3 问题 2：在进程之间切换 .....	40	问题 .....	64
6.4 担心并发吗 .....	44	第 9 章 调度：比例份额 .....	65
6.5 小结 .....	45	9.1 基本概念：彩票数表示份额 .....	65
参考资料 .....	45	9.2 彩票机制 .....	66

9.3 实现 .....	67	15.2 一个例子 .....	101
9.4 一个例子 .....	68	15.3 动态（基于硬件）重定位 .....	103
9.5 如何分配彩票 .....	68	15.4 硬件支持：总结 .....	105
9.6 为什么不是确定的 .....	69	15.5 操作系统的问题 .....	105
9.7 小结 .....	70	15.6 小结 .....	108
参考资料 .....	70	参考资料 .....	109
作业 .....	71	作业 .....	110
问题 .....	71	问题 .....	110
<b>第 10 章 多处理器调度（高级） .....</b>	<b>73</b>	<b>第 16 章 分段 .....</b>	<b>111</b>
10.1 背景：多处理器架构 .....	73	16.1 分段：泛化的基址/界限 .....	111
10.2 别忘了同步 .....	75	16.2 我们引用哪个段 .....	113
10.3 最后一个问题：缓存亲和度 .....	76	16.3 栈怎么办 .....	114
10.4 单队列调度 .....	76	16.4 支持共享 .....	114
10.5 多队列调度 .....	77	16.5 细粒度与粗粒度的分段 .....	115
10.6 Linux 多处理器调度 .....	79	16.6 操作系统支持 .....	115
10.7 小结 .....	79	16.7 小结 .....	117
参考资料 .....	79	参考资料 .....	117
<b>第 11 章 关于 CPU 虚拟化的总结对话 ..</b>	<b>81</b>	作业 .....	118
<b>第 12 章 关于内存虚拟化的对话 .....</b>	<b>83</b>	问题 .....	119
<b>第 13 章 抽象：地址空间 .....</b>	<b>85</b>	<b>第 17 章 空闲空间管理 .....</b>	<b>120</b>
13.1 早期系统 .....	85	17.1 假设 .....	120
13.2 多道程序和时分共享 .....	85	17.2 底层机制 .....	121
13.3 地址空间 .....	86	17.3 基本策略 .....	126
13.4 目标 .....	87	17.4 其他方式 .....	128
13.5 小结 .....	89	17.5 小结 .....	130
参考资料 .....	89	参考资料 .....	130
<b>第 14 章 插叙：内存操作 API .....</b>	<b>91</b>	作业 .....	131
14.1 内存类型 .....	91	问题 .....	131
14.2 malloc()调用 .....	92	<b>第 18 章 分页：介绍 .....</b>	<b>132</b>
14.3 free()调用 .....	93	18.1 一个简单例子 .....	132
14.4 常见错误 .....	93	18.2 页表存在哪里 .....	134
14.5 底层操作系统支持 .....	96	18.3 列表中究竟有什么 .....	135
14.6 其他调用 .....	97	18.4 分页：也很慢 .....	136
14.7 小结 .....	97	18.5 内存追踪 .....	137
参考资料 .....	97	18.6 小结 .....	139
作业（编码） .....	98	参考资料 .....	139
问题 .....	98	作业 .....	140
<b>第 15 章 机制：地址转换 .....</b>	<b>100</b>	问题 .....	140
15.1 假设 .....	101	<b>第 19 章 分页：快速地址转换（TLB） ..</b>	<b>142</b>
		19.1 TLB 的基本算法 .....	142

19.2 示例：访问数组 .....	143	21.7 小结 .....	170
19.3 谁来处理 TLB 未命中 .....	145	参考资料 .....	171
19.4 TLB 的内容 .....	146	<b>第 22 章 超越物理内存：策略</b> .....	172
19.5 上下文切换时对 TLB 的处理 .....	147	22.1 缓存管理 .....	172
19.6 TLB 替换策略 .....	149	22.2 最优替换策略 .....	173
19.7 实际系统的 TLB 表项 .....	149	22.3 简单策略：FIFO .....	175
19.8 小结 .....	150	22.4 另一简单策略：随机 .....	176
参考资料 .....	151	22.5 利用历史数据：LRU .....	177
作业（测量） .....	152	22.6 工作负载示例 .....	178
问题 .....	153	22.7 实现基于历史信息的算法 .....	180
<b>第 20 章 分页：较小的表</b> .....	154	22.8 近似 LRU .....	181
20.1 简单的解决方案：更大的页 .....	154	22.9 考虑脏页 .....	182
20.2 混合方法：分页和分段 .....	155	22.10 其他虚拟内存策略 .....	182
20.3 多级页表 .....	157	22.11 抖动 .....	183
20.4 反向页表 .....	162	22.12 小结 .....	183
20.5 将页表交换到磁盘 .....	163	参考资料 .....	183
20.6 小结 .....	163	作业 .....	185
参考资料 .....	163	问题 .....	185
作业 .....	164	<b>第 23 章 VAX/VMS 虚拟内存系统</b> .....	186
问题 .....	164	23.1 背景 .....	186
<b>第 21 章 超越物理内存：机制</b> .....	165	23.2 内存管理硬件 .....	186
21.1 交换空间 .....	165	23.3 一个真实的地址空间 .....	187
21.2 存在位 .....	166	23.4 页替换 .....	189
21.3 页错误 .....	167	23.5 其他漂亮的虚拟内存技巧 .....	190
21.4 内存满了怎么办 .....	168	23.6 小结 .....	191
21.5 页错误处理流程 .....	168	参考资料 .....	191
21.6 交换何时真正发生 .....	169	<b>第 24 章 内存虚拟化总结对话</b> .....	193

## 第 2 部分 并发

<b>第 25 章 关于并发的对话</b> .....	196	参考资料 .....	207
<b>第 26 章 并发：介绍</b> .....	198	作业 .....	208
26.1 实例：线程创建 .....	199	问题 .....	208
26.2 为什么更糟糕：共享数据 .....	201	<b>第 27 章 插叙：线程 API</b> .....	210
26.3 核心问题：不可控的调度 .....	203	27.1 线程创建 .....	210
26.4 原子性愿望 .....	205	27.2 线程完成 .....	211
26.5 还有一个问题：等待另一个 线程 .....	206	27.3 锁 .....	214
26.6 小结：为什么操作系统课要研究 并发 .....	207	27.4 条件变量 .....	215
		27.5 编译和运行 .....	217
		27.6 小结 .....	217

参考资料 .....	218	30.3 覆盖条件 .....	260
<b>第 28 章 锁</b> .....	219	30.4 小结 .....	261
28.1 锁的基本思想 .....	219	参考资料 .....	261
28.2 Pthread 锁 .....	220	<b>第 31 章 信号量</b> .....	263
28.3 实现一个锁 .....	220	31.1 信号量的定义 .....	263
28.4 评价锁 .....	220	31.2 二值信号量（锁） .....	264
28.5 控制中断 .....	221	31.3 信号量用作条件变量 .....	266
28.6 测试并设置指令（原子交换） .....	222	31.4 生产者/消费者（有界缓冲区） 问题 .....	268
28.7 实现可用的自旋锁 .....	223	31.5 读者—写者锁 .....	271
28.8 评价自旋锁 .....	225	31.6 哲学家就餐问题 .....	273
28.9 比较并交换 .....	225	31.7 如何实现信号量 .....	275
28.10 链接的加载和条件式存储指令 .....	226	31.8 小结 .....	276
28.11 获取并增加 .....	228	参考资料 .....	276
28.12 自旋过多：怎么办 .....	229	<b>第 32 章 常见并发问题</b> .....	279
28.13 简单方法：让出来吧，宝贝 .....	229	32.1 有哪些类型的缺陷 .....	279
28.14 使用队列：休眠替代自旋 .....	230	32.2 非死锁缺陷 .....	280
28.15 不同操作系统，不同实现 .....	232	32.3 死锁缺陷 .....	282
28.16 两阶段锁 .....	233	32.4 小结 .....	288
28.17 小结 .....	233	参考资料 .....	289
参考资料 .....	233	<b>第 33 章 基于事件的并发（进阶）</b> .....	291
作业 .....	235	33.1 基本想法：事件循环 .....	291
问题 .....	235	33.2 重要 API：select()（或 poll()） .....	292
<b>第 29 章 基于锁的并发数据结构</b> .....	237	33.3 使用 select() .....	293
29.1 并发计数器 .....	237	33.4 为何更简单？无须锁 .....	294
29.2 并发链表 .....	241	33.5 一个问题：阻塞系统调用 .....	294
29.3 并发队列 .....	244	33.6 解决方案：异步 I/O .....	294
29.4 并发散列表 .....	245	33.7 另一个问题：状态管理 .....	296
29.5 小结 .....	246	33.8 什么事情仍然很难 .....	297
参考资料 .....	247	33.9 小结 .....	298
<b>第 30 章 条件变量</b> .....	249	参考资料 .....	298
30.1 定义和程序 .....	250	<b>第 34 章 并发的总结对话</b> .....	300
30.2 生产者/消费者（有界缓冲区） 问题 .....	252		

## 第 3 部分 持久性

<b>第 35 章 关于持久性的对话</b> .....	302	36.3 标准协议 .....	304
<b>第 36 章 I/O 设备</b> .....	303	36.4 利用中断减少 CPU 开销 .....	305
36.1 系统架构 .....	303	36.5 利用 DMA 进行更高效的数据 传送 .....	306
36.2 标准设备 .....	304		

36.6	设备交互的方法	307	39.8	获取文件信息	348
36.7	纳入操作系统：设备驱动程序	307	39.9	删除文件	349
36.8	案例研究：简单的 IDE 磁盘驱动程序	309	39.10	创建目录	349
36.9	历史记录	311	39.11	读取目录	350
36.10	小结	311	39.12	删除目录	351
	参考资料	312	39.13	硬链接	351
<b>第 37 章</b>	<b>磁盘驱动器</b>	314	39.14	符号链接	353
37.1	接口	314	39.15	创建并挂载文件系统	354
37.2	基本几何形状	314	39.16	小结	355
37.3	简单的磁盘驱动器	315		参考资料	355
37.4	I/O 时间：用数学	318		作业	356
37.5	磁盘调度	320		问题	356
37.6	小结	323	<b>第 40 章</b>	<b>文件系统实现</b>	357
	参考资料	323	40.1	思考方式	357
	作业	324	40.2	整体组织	358
	问题	324	40.3	文件组织：inode	359
<b>第 38 章</b>	<b>廉价冗余磁盘阵列 (RAID)</b>	326	40.4	目录组织	363
38.1	接口和 RAID 内部	327	40.5	空闲空间管理	364
38.2	故障模型	327	40.6	访问路径：读取和写入	364
38.3	如何评估 RAID	328	40.7	缓存和缓冲	367
38.4	RAID 0 级：条带化	328	40.8	小结	369
38.5	RAID 1 级：镜像	331		参考资料	369
38.6	RAID 4 级：通过奇偶校验节省空间	333		作业	370
38.7	RAID 5 级：旋转奇偶校验	336		问题	371
38.8	RAID 比较：总结	337	<b>第 41 章</b>	<b>局部性和快速文件系统</b>	372
38.9	其他有趣的 RAID 问题	338	41.1	问题：性能不佳	372
38.10	小结	338	41.2	FFS：磁盘意识是解决方案	373
	参考资料	339	41.3	组织结构：柱面组	373
	作业	340	41.4	策略：如何分配文件和目录	374
	问题	340	41.5	测量文件的局部性	375
<b>第 39 章</b>	<b>插叙：文件和目录</b>	342	41.6	大文件例外	376
39.1	文件和目录	342	41.7	关于 FFS 的其他几件事	377
39.2	文件系统接口	343	41.8	小结	378
39.3	创建文件	343		参考资料	378
39.4	读写文件	344	<b>第 42 章</b>	<b>崩溃一致性：FSCK 和日志</b>	380
39.5	读取和写入，但不按顺序	346	42.1	一个详细的例子	380
39.6	用 fsync()立即写入	346	42.2	解决方案 1：文件系统检查程序	383
39.7	文件重命名	347	42.3	解决方案 2：日志（或预写日志）	384

42.4 解决方案 3: 其他方法 .....	392	参考资料 .....	429
42.5 小结 .....	393	<b>第 48 章 Sun 的网络文件系统 (NFS)</b> ..	430
参考资料 .....	393	48.1 基本分布式文件系统 .....	430
<b>第 43 章 日志结构文件系统</b> .....	395	48.2 交出 NFS .....	431
43.1 按顺序写入磁盘 .....	396	48.3 关注点: 简单快速的服务器崩溃 恢复 .....	431
43.2 顺序而高效地写入 .....	396	48.4 快速崩溃恢复的关键: 无状态 ..	432
43.3 要缓冲多少 .....	397	48.5 NFSv2 协议 .....	433
43.4 问题: 查找 inode .....	398	48.6 从协议到分布式文件系统 .....	434
43.5 通过间接解决方案: inode 映射 ..	398	48.7 利用幂等操作处理服务器故障 ..	435
43.6 检查点区域 .....	399	48.8 提高性能: 客户端缓存 .....	437
43.7 从磁盘读取文件: 回顾 .....	400	48.9 缓存一致性问题 .....	437
43.8 目录如何 .....	400	48.10 评估 NFS 的缓存一致性 .....	439
43.9 一个新问题: 垃圾收集 .....	401	48.11 服务器端写缓冲的隐含意义 .....	439
43.10 确定块的死活 .....	402	48.12 小结 .....	440
43.11 策略问题: 要清理哪些块, 何时清理 .....	403	参考资料 .....	440
43.12 崩溃恢复和日志 .....	403	<b>第 49 章 Andrew 文件系统 (AFS)</b> ..	442
43.13 小结 .....	404	49.1 AFS 版本 1 .....	442
参考资料 .....	404	49.2 版本 1 的问题 .....	443
<b>第 44 章 数据完整性和保护</b> .....	407	49.3 改进协议 .....	444
44.1 磁盘故障模式 .....	407	49.4 AFS 版本 2 .....	444
44.2 处理潜在的扇区错误 .....	409	49.5 缓存一致性 .....	446
44.3 检测讹误: 校验和 .....	409	49.6 崩溃恢复 .....	447
44.4 使用校验和 .....	412	49.7 AFSv2 的扩展性和性能 .....	448
44.5 一个新问题: 错误的写入 .....	412	49.8 AFS: 其他改进 .....	450
44.6 最后一个问题: 丢失的写入 .....	413	49.9 小结 .....	450
44.7 擦净 .....	413	参考资料 .....	451
44.8 校验和的开销 .....	414	作业 .....	452
44.9 小结 .....	414	问题 .....	452
参考资料 .....	414	<b>第 50 章 关于分布式的总结对话</b> .....	453
<b>第 45 章 关于持久的总结对话</b> .....	417	<b>附录 A 关于虚拟机监视器的对话</b> .....	454
<b>第 46 章 关于分布式的对话</b> .....	418	<b>附录 B 虚拟机监视器</b> .....	455
<b>第 47 章 分布式系统</b> .....	419	<b>附录 C 关于监视器的对话</b> .....	466
47.1 通信基础 .....	420	<b>附录 D 关于实验室的对话</b> .....	467
47.2 不可靠的通信层 .....	420	<b>附录 E 实验室: 指南</b> .....	468
47.3 可靠的通信层 .....	422	<b>附录 F 实验室: 系统项目</b> .....	478
47.4 通信抽象 .....	424	<b>附录 G 实验室: xv6 项目</b> .....	480
47.5 远程过程调用 (RPC) .....	425		
47.6 小结 .....	428		