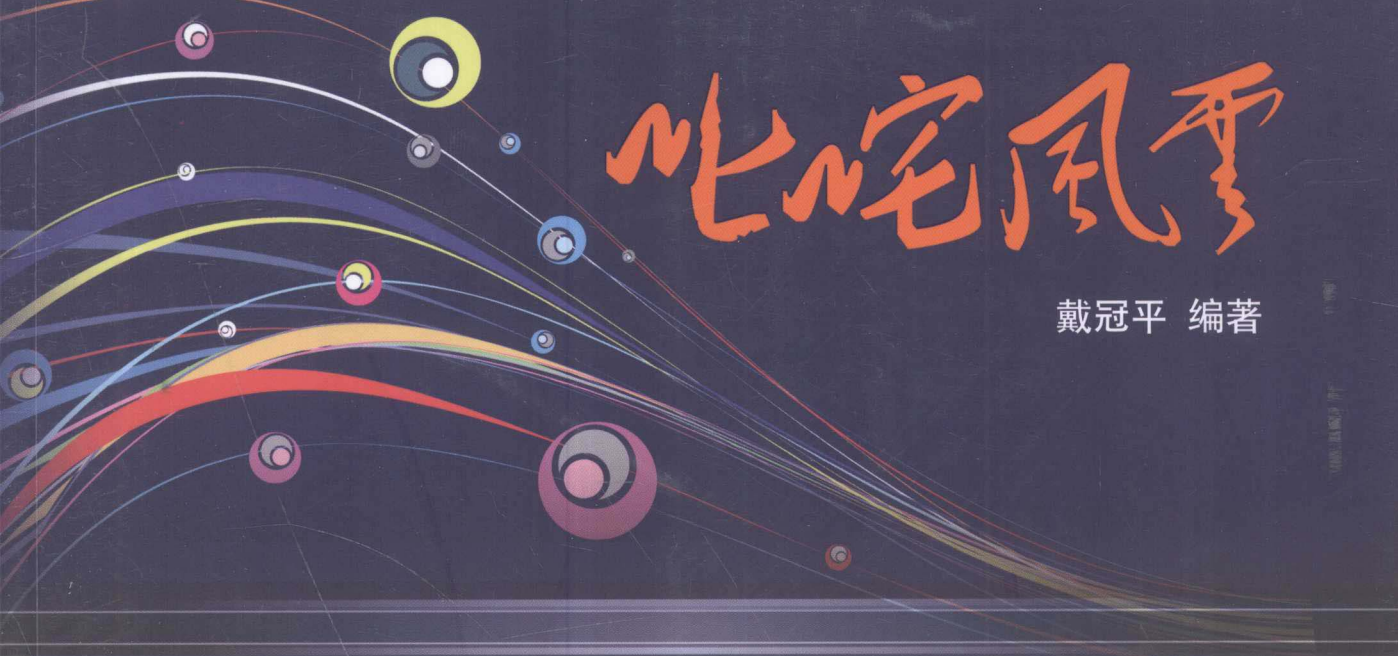




叱咤风云

戴冠平 编著

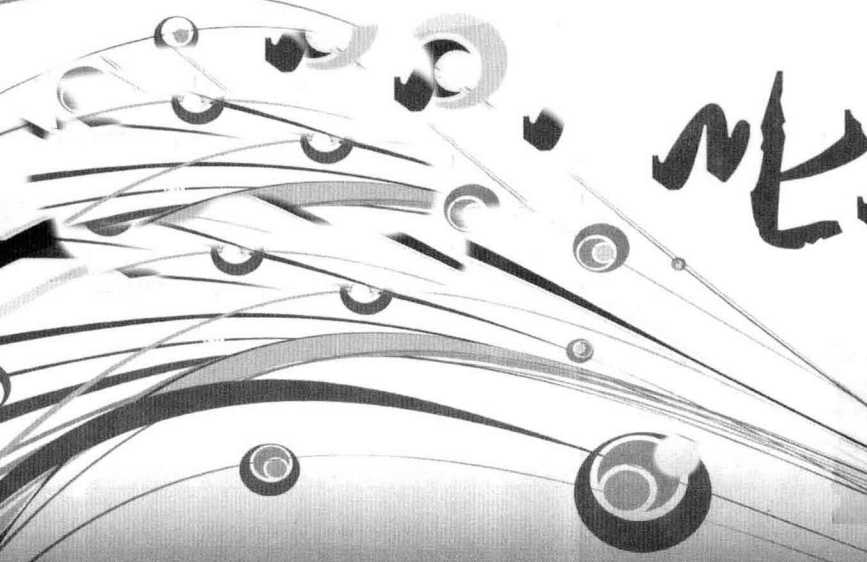
IXEDO® TUXEDO® TUXEDO® TU
EDO® TUXEDO® TUXEDO® TUXE
IO® TUXEDO® TUXEDO® TUXED

Tuxedo

企业级运维实战

清华大学出版社





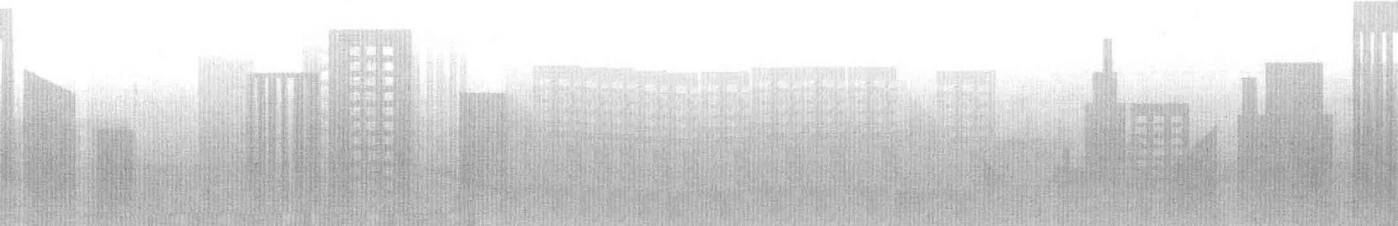
叱咤风云

戴冠平 编著



Tuxedo

企业级运维实战



清华大学出版社
北 京

内 容 简 介

Tuxedo 是一个成熟多年的联机事务处理产品,用于开发、集成、部署和管理大型分布式应用。本书由浅入深地论述了 Tuxedo 的体系和理念,结合作者多年业内专家的从业经验,充分地剖析了 Tuxedo 的核心技术。对于 Tuxedo 在实际生产中,客户系统累积出现的各种典型故障和错误,分门别类地进行了透彻讲解,给出了具体的诊断思路和解决方案,具有非常现实、非常重要的指导意义和实战价值。

本书适合作为 Tuxedo 运维技术人员的参考手册,也可以作为高校相关专业师生学习资料。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

叱咤风云: Tuxedo 企业级运维实战 / 戴冠平编著. —北京: 清华大学出版社, 2012. 1
ISBN 978-7-302-26762-1

I. ①叱… II. ①戴… III. ①互联网络—基本知识 IV. ①TP31

中国版本图书馆 CIP 数据核字(2011)第 185704 号

责任编辑: 夏兆彦

责任校对: 徐俊伟

责任印制: 何 芊

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62795954, jsjic@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 北京嘉实印刷有限公司

经 销: 全国新华书店

开 本: 190×260 印 张: 17.5 字 数: 437 千字

版 次: 2012 年 1 月第 1 版 印 次: 2012 年 1 月第 1 次印刷

印 数: 1~4000

定 价: 45.00 元

前 言

20 世纪 80 年代以来，伴随着网络和互联网的快速发展，中国各行各业的企业级信息系统应用从其体系结构、所采纳的技术和应用本身的复杂性都发生了深刻的变化。企业的 IT 系统日益成为其业务运营的支撑核心。主机、数据库、中间件、存储、网络等基础设施软硬件构成了企业 IT 系统的基石，同时这些不同层面的技术随着企业应用的规模扩张和复杂性增强，对企业的 IT 运维团队提出了越来越高的挑战，这也促使企业用户越来越深刻地认识到运维和服务所起的关键作用，及其带给自己的不菲价值。

在当今信息时代和人才时代，如何能够获得高质量的、快捷便利的 IT 服务是摆在所有企业 IT 运维团队面前的一个迫切需要思考的问题，而单纯求援于目前 IT 大厂商，其昂贵服务成本常常令大家觉得突兀和吃惊。一个可以作为参考的业内公开数据是，主流企业级软件公司的主要收入，尤其是利润来源，并不是软件销售本身，而是其技术和运维服务；而一般只够支付几年的服务运维费用，就已足够购买一套全新的软件产品。如果有机会能够以相对公开的方式，针对企业级 IT 的运维进行系统性的知识整理和经验共享，相信这种努力和转移应当是件相当有益处并有意义的工作。

就我个人来讲，毕竟做了这么多年的中间件，很久很久以前，就一直想写这么一本书和大家分享；毕竟曾经写了那么多零零散散的心得，如浩瀚夜空中的点点繁星，错落杂乱；终于有了一大段空闲的时光，能让这些点点滴滴串通起来，努力组成一个有形状有系统的星座。

感谢当年引导和伴随我成长的 BEA 老同事们，让我有机会在中间件的领域里遨游，越游越深，虽然现在应该改称 Oracle 的各位新朋旧友了。也特别感谢联动北方的技术团队，正是你们的敬业精神，才激励我最终坚持把这本书编著出来；也正是你们的辛勤帮助，才有了这本书如此专业详实的内容。

独坐明窗映斜阳，听沧海潮起潮落，观碧空云卷云舒；恰逢母校百年风范依然，学子拳拳赤心依旧，“自强不息，厚德载物”。此书的及时完稿，也算是一丝慰藉和心意了。

作者

2011 年于 Brisbane Manly 海滨

目 录

第 1 篇 入 门 篇

第 1 章 Tuxedo 概述	2
1.1 什么是 Tuxedo 系统	2
1.2 Tuxedo 的历史及发展	2
1.1.1 Tuxedo 的产生	3
1.2.2 Tuxedo 的发展	3
1.3 Tuxedo 支持的平台	5
1.4 Tuxedo 的技术架构	6
1.4.1 客户机/服务器模式	6
1.4.2 Tuxedo ATMI 体系结构	10
1.4.3 Tuxedo CORBA 体系结构	13
1.4.4 ATMI 与 CORBA 对比	17
1.5 Tuxedo 系统的关键特性	17
1.5.1 名字服务和位置透明性	17
1.5.2 强大的 C/S 通信能力	17
1.5.3 强大的联机交易性能	18
1.5.4 强大的分布式事务协调能力	18
1.5.5 完善的负载均衡机制	18
1.5.6 数据依赖路由	18
1.5.7 请求的优先级	19
1.5.8 容错和透明故障迁移	20
1.5.9 安全性	20
1.5.10 开放性和易用性	20
1.5.11 先进的组织架构	21
1.6 Tuxedo 与其他产品横向与纵向的比较	21
1.6.1 CICS 简介	21
1.6.2 Tuxedo 和 CICS 的对比	22
第 2 章 Tuxedo 的简单安装和运行	24
2.1 安装前准备	24
2.1.1 检查软件包	24
2.1.2 必备的硬件和软件	24
2.1.3 如何获得安装介质及文档	24
2.1.4 Tuxedo 许可证	24

2.2	快速安装	25
2.2.1	Tuxedo 环境要求	25
2.2.2	内核参数的调整	25
2.2.3	进行 Tuxedo 安装	25
2.2.4	兼顾需要 License 的版本	28
2.3	部署应用（simpapp 例子）	29
2.3.1	修改配置文件	29
2.3.2	加载配置文件	29
2.3.3	启动 Tuxedo	30
2.3.4	相关的日志文件	30
2.4	编译和运行	30
2.4.1	编译程序	30
2.4.2	运行程序	31
2.5	卸载 Tuxedo	31

第 2 篇 基 础 篇

第 3 章	OLTP 基本知识	34
3.1	三层或多层 C/S 架构	34
3.2	事务的概念	35
3.2.1	什么是事务	35
3.2.2	什么是全局事务	35
3.2.3	XA 规范	36
3.3	IPC 机制简介	37
3.3.1	命名管道	37
3.3.2	消息队列	37
3.3.3	信号量	38
3.3.4	共享内存	38
3.3.5	IPC 资源相关的操作系统内核参数	38
第 4 章	Tuxedo 的基本概念	39
4.1	域 Domain	39
4.1.1	域的概念和范围	39
4.1.2	为什么要使用域	39
4.2	逻辑机器 Machine	39
4.2.1	Machine 的概念和范围	39
4.2.2	为什么使用 Machine	40
4.3	服务器组 Group	40
4.3.1	组的概念和范围	40
4.3.2	为什么要使用组	40
4.4	服务进程 Server 和服务 Service	40

4.4.1	什么是 Server 和 Service	40
4.4.2	Server 和 Service 的关系	40
4.4.3	服务进程中的主要函数	41
4.5	Tuxedo 通信方式综述	41
4.5.1	请求/应答式通信	41
4.5.2	会话方式 tpsend()/tprecv()	45
4.5.3	通知广播 tpnotify()/tpbroadcast()	47
4.5.4	事件代理 tppost()/tpsubscribe()	49
4.5.5	队列存储 tpenqueue()/tpdequeue()	50
4.6	Tuxedo 多机部署	52
4.6.1	Tuxedo 集群	52
4.6.2	及多套 Tuxedo 应用之间的通信	52
4.7	Tuxedo 远程客户端	52
4.7.1	什么是远程客户端	52
4.7.2	WSL/WSH 配置与工作机理	53
4.7.3	Java 远程客户端接入 Jolt	54
第 5 章	Tuxedo 主要的目录结构	57
5.1	总体目录结构分布	57
5.2	可执行文件说明	57
5.3	提要系统目录 udataobj	58
5.4	C 语言头文件和库	58
第 6 章	Tuxedo 配置相关文件	59
6.1	几个关键环境变量	59
6.2	系统配置文件 UBB 及其内容	60
6.2.1	*RESOURCES 段的配置	61
6.2.2	*MACHINES 段的配置	62
6.2.3	*GROUPS 段的配置	63
6.2.4	*NETWORK 段的配置	63
6.2.5	*SERVERS 段的配置	63
6.2.6	*SERVICES 段的配置	64
6.2.7	*ROUTING 段的配置	64
6.3	域配置文件 DMCONFIG 及其内容	65
6.3.1	域 (Domain) 简介	65
6.3.2	Tuxedo 域划分原则	65
6.3.3	域 (Domain) 的功能	66
6.3.4	Tuxedo Domain 的配置	66
6.4	日志文件 ULOG	67

第 3 篇 实 施 篇

第 7 章	Tuxedo 应用的部署模式	70
7.1	单机 SHM 模式	70

7.2	多机 MP 模式	72
7.3	多域模式	77
7.4	各种模式的比较	81
7.5	Tuxedo 与多种平台连通	83
7.5.1	与其他系统的互联概要	83
7.5.2	经典的 WTC	83
7.5.3	JCA Adapter 新特性	88
第 8 章	Tuxedo 常用的管理操作	90
8.1	启停 Tuxedo 应用	90
8.1.1	相关应用环境	90
8.1.2	启动 Tuxedo 应用	92
8.1.3	停止 Tuxedo 应用	92
8.2	管理和监控	93
8.2.1	一般管理监控 tadmin	93
8.2.2	域管理监控 dadmin	96
8.2.3	队列管理监控 qadmin	97
8.3	动态配置 tmconfig	105
8.3.1	概述	105
8.3.2	配置 tmconfig 运行环境	105
8.3.3	tmconfig 常用操作	106
8.4	TSAM	107
8.4.1	TSAM 简介	107
8.4.2	TSAM 安装	107
8.4.3	TSAM 配置	112
8.4.4	TSAM 监控	113
8.4.5	TSAM 监测预警	114
8.5	高可用性	115
8.5.1	高可用性概述	115
8.5.2	高可用性详细分析	116
8.6	Tuxedo 如何打补丁	118
8.6.1	备份	118
8.6.2	补丁升级	118
8.6.3	重启应用	119
第 9 章	如何用好全局事务	120
9.1	什么是全局事务	120
9.2	本地事务的优缺点	120
9.3	Tuxedo 对事务的控制与管理	120
9.4	常用事务相关的函数	121
9.5	数据库连接	122

9.5.1	TMS 介绍	122
9.5.2	XA 模式与 NO-XA 模式	123
9.5.3	Tuxedo 与各种数据库的连接	123
9.6	全局事务的使用规则	124
9.6.1	谁发起谁结束	124
9.6.2	不允许嵌套	125
9.6.3	处理好超时	125
9.7	事务挂起的问题	125
第 10 章	Tuxedo 性能调优	127
10.1	目标描述	127
10.2	调优独立的 Tuxedo 服务	127
10.3	将相似的 Tuxedo 服务分组到一个 SERVER	129
10.4	调整 SERVER 数量	131
10.5	FML 性能	134
10.6	额外的性能参数	135
10.6.1	多个 WSH 连接	135
10.6.2	关闭 WSL / WSH 加密	136
10.6.3	打开 WSL / WSH 压缩	136
10.6.4	机器类型	136
10.6.5	SPINCOUNT	136
10.6.6	去掉授权和审计安全	137
10.6.7	关闭多线程处理	137
10.6.8	关闭 XA 事务	137

第 4 篇 诊 断 篇

第 11 章	Tuxedo 监控	140
11.1	监控 Tuxedo 应用的方法	140
11.2	可以监控的系统和应用数据	141
11.3	使用管理控制台监控应用	141
11.4	使用命令行方式监控	141
11.5	使用 EventBroker 监视应用程序	141
11.5.1	相关 API 介绍	142
11.5.2	相关例子参考	143
11.6	使用 MIB 监视应用程序	144
11.7	使用日志文件来监控	147
11.7.1	Tuxedo 日志的分类	147
11.7.2	Tuxedo 事务日志	147
11.7.3	Tuxedo 用户日志	149

第 12 章 服务 core dump 分析	151
12.1 什么是服务 core dump 文件	151
12.2 什么情况可以导致 core dump 文件生成	151
12.3 服务器 core dump 文件探查	151
12.3.1 检查系统环境以保证 core dump 生成	152
12.3.2 保存 core 文件	152
12.3.3 找到 core 文件并使用其探测错误成因	153
12.3.4 探查错误的根源	154
12.4 core dump 成因案例	157
12.4.1 为 strings 分配太少的内存	157
12.4.2 使用已经释放的内存	158
12.4.3 在 scanf 调用丢掉&	158
12.4.4 用非法的参数调用函数	158
12.4.5 没有分配内存给指针	158
12.4.6 没有初始化变量	158
12.5 错误信息的含义	158
12.5.1 总线错误	159
12.5.2 内存错误	159
12.5.3 I/O 陷阱	159
12.5.4 跟踪/BPT 陷阱	159
12.5.5 浮点异常	160
12.5.6 分段错误	160
12.5.7 非法命令	160
第 13 章 异常高 CPU 占用率故障	161
13.1 异常高 CPU 占用率	161
13.2 异常高 CPU 占用率的伴随症状	161
13.3 异常高 CPU 占用率探查	161
13.3.1 探查概述	161
13.3.2 初步探查	162
13.3.3 进一步跟踪	162
13.4 异常高 CPU 占用率故障排除策略	165
第 14 章 常规服务器阻塞故障	166
14.1 确认是服务器阻塞	166
14.2 服务器阻塞的可能原因	166
14.3 服务器阻塞的探查	166
14.3.1 Solaris	167
14.3.2 Linux	170
14.3.3 AIX	171
14.3.4 HP-UX	172

14.3.5	Windows	173
14.4	故障排查清单	175
14.5	进程挂起例子分析	175
14.5.1	进程挂起在 sleep 循环中	175
14.5.2	进程一直等待数据库查询大数据	175
14.5.3	死锁：不同 SERVER 中的服务相互调用	175
第 15 章	内存不足和内存泄漏故障	177
15.1	问题描述	177
15.2	问题诊断	177
15.2.1	进程地址空间及物理内存的区别	177
15.2.2	为什么这个问题会发生	177
15.3	问题研究	178
15.4	分析与检测内存泄漏	179
15.4.1	监控进程虚拟内存大小	179
15.4.2	隔离应用程序来跟踪内存泄漏	183
15.4.3	隔离应用服务	184
15.4.4	隔离应用的组织机构的库/代码	184
15.5	内存分析工具	184
15.5.1	memwatch	184
15.5.2	Purify	185
15.5.3	Valgrind	186
15.5.4	Insure++	186
15.6	常见的内存泄漏的原因	187
15.6.1	非成对使用 tpalloc()/malloc()与 tpfree()/free()	187
15.6.2	重写指针	187
15.6.3	C 库函数的 bug	188
第 16 章	与全局事务 XA 相关的故障	189
16.1	问题描述	189
16.2	通过配置让 Tuxedo 支持事务	189
16.2.1	配置 Tuxedo XA	189
16.2.2	创建事务管理器和 XA 服务器	190
16.2.3	XA-OPENINFO 字符串	191
16.2.4	TMS 服务器	192
16.3	运行时问题	193
16.3.1	调用 tx_open() 或 tlopen() 失败	193
16.3.2	启发式失败	193
16.3.3	xa_start() 返回 XAER_RMERR	193
16.3.4	xa_start()=-9 问题	194
16.3.5	Oracle TMS 挂起错误	195

16.4	XA 跟踪	195
16.4.1	TMTRACE	195
16.4.2	DbgFl	196
第 17 章	IPC 相关故障	198
17.1	Tuxedo 使用的 IPC	198
17.1.1	信号量 (Semaphore)	198
17.1.2	消息队列 (Message Queue)	198
17.1.3	共享内存 (Shared Memory)	199
17.1.4	Tuxedo 使用的 IPC 资源	199
17.1.5	定义 IPC 限制	199
17.2	IPC 设置	200
17.3	IPC 命令	201
17.3.1	ipcs	202
17.3.2	ipcrm	202
17.3.3	tmipcrm	203
17.3.4	IPC 清除脚本	204
17.3.5	bbsread	204
17.4	IPC 常见疑难问题	205
第 18 章	一般网络故障	207
18.1	防火墙及防火墙相关故障	207
18.2	网络状态查询 netstat	207
18.3	网络报文追踪	208
18.4	其他网络工具	210
18.4.1	ping 命令	210
18.4.2	telnet 命令	211
18.4.3	ifconfig	212
18.4.4	ipconfig	212
18.4.5	网络压缩	213
18.5	Tuxedo 多机架构 (MP)	213
18.5.1	Tuxedo MP 应用的注意事项	213
18.5.2	负载均衡网络应用程序	213
18.5.3	常见问题	214
18.5.4	用 tmdadmin 监控	215
18.6	Tuxedo 的多域架构 (Domain)	217
18.6.1	DMCONFIG 常见配置问题	218
18.6.2	使用 dmdadmin 监控 domain	218
18.7	故障分类排除	219
18.7.1	Tuxedo MP 应用	219
18.7.2	Tuxedo Domain 应用	220

第 19 章 WTC 和 JOLT 支持模式	221
19.1 重温什么是 WTC 和 JOLT	221
19.1.1 概述	221
19.1.2 WebLogic Txuedo 连接器介绍	221
19.1.3 JOLT 介绍	221
19.2 什么引发 WTC 和 JOLT 故障	222
19.2.1 JOLT 和 WTC 问题主要的两种形式	222
19.2.2 选择适当技术: JOLT VS WTC	222
19.2.3 引发 JOLT 和 WTC 错误的主要原因	222
19.3 WTC 和 JOLT 相关故障的症状及解决方法	223
19.3.1 JOLT 常见问题及解决方法	223
19.3.2 WTC 常见问题及解决方法	225
19.4 WTC 和 JOLT 故障排查清单	226
19.4.1 WTC 故障排除步骤	226
19.4.2 JOLT 故障排除步骤	226

第 5 篇 高 阶 篇

第 20 章 Tuxedo 的 COBOL 编程	228
20.1 运行环境配置	228
20.1.1 Tuxedo COBOL 数据记录类型	228
20.1.2 如何使用 FML 数据类型	233
20.1.3 Tuxedo COBOL 客户端编程	236
20.1.4 Tuxedo COBOL 服务器端编程	238
20.2 Tuxedo 下使用 COBOL 编程与 C 语言编程的异同	245
20.3 使用 COBOL 编写 Tuxedo 程序的局限性	245
20.3.1 FML 支持的局限性	245
20.3.2 COBOL 语言编译的局限性	245
20.3.3 开发人员要求比较高	246
20.3.4 错误处理开销	246
20.3.5 数据类型的使用相对有限	246
20.4 Tuxedo 下 COBOL 与 C 语言的混合编程及模块集成	246
20.4.1 混合编程规则	246
20.4.2 COBOL 调用 C	247
20.4.3 C 调用 COBOL	249
第 21 章 基于 Tuxedo 对大机应用的迁移——ART	251
21.1 ART 简介	251
21.2 Application Rehosting Workbench 作业运行环境	251
21.2.1 关键特性	251
21.2.2 优点	251

- 21.2.3 流程简介.....252
 - 21.2.4 详细流程.....252
 - 21.3 ART for CICS 作业运行环境.....255
 - 21.3.1 关键特性.....255
 - 21.3.2 优点.....255
 - 21.3.3 流程简介.....256
 - 21.3.4 详细流程.....256
 - 21.4 ART for Batch 作业运行环境259
 - 21.4.1 流程简介.....259
 - 21.4.2 详细流程.....260
- 后记.....264

第 1 篇

入 门 篇

第 1 章 Tuxedo 概述

Tuxedo 是一个成熟多年的联机事务处理产品，用于开发、集成、部署和管理大型分布式应用，拥有处理关键业务应用系统问题所需的性能、安全性、可扩展性和高可用性，同时又易于安装、部署和管理。

1.1 什么是 Tuxedo 系统

Tuxedo (Transactions for UNIX, Extended for Distributed Operations) 是在企业分布式计算环境中，开发和管理三层“客户机-服务器”(C/S) 关键业务系统的平台软件。它具有空前的交易处理性能、高度的可靠性和无限的伸缩性，能为企业建立、运行和管理大规模、高性能、分布式的关键业务系统提供强大的支撑平台。该平台具有很好的开放性，它支持各种各样的客户端、数据库、网络、通信方式和主机遗留系统。开发人员能够用它建立跨多个硬件平台、数据库和操作系统的系统。

图 1-1 展示了企业级 Tuxedo 系统的体系结构。

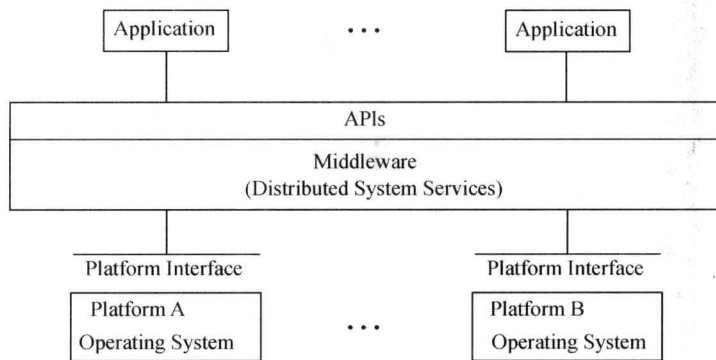


图 1-1

Tuxedo 具有全面而健壮的功能。在企业分布式联机交易系统中，Tuxedo 常作为一个事务监控器 (TP Monitor, TM) 来协调分布式事务；在构建多层 C/S 应用系统中，Tuxedo 常作为中间件的角色部署在客户机和服务器之间，为应用提供服务；在构建企业级应用系统中，Tuxedo 常以应用服务器平台的角色出现，为企业应用提供部署环境和运行环境。

Tuxedo 支持广泛的操作系统平台，包括 64 位/32 位的 Solaris, Linux, IBM 的 AIX、System i, HP 的 HP-UX、OpenVMS, 以及 Microsoft 的 Windows。

1.2 Tuxedo 的历史及发展

Tuxedo 是一个久经考验的成熟的系统，在 20 多年的历史中不断地发展和增强。

1.1.1 Tuxedo 的产生

Tuxedo 系统于 1983 年由美国贝尔实验室的 AT&T 分部开发，最初被命名为 UNITS (UNIX Transaction System)。开发 UNITS 的目的是便于 AT&T 内部构建基于 UNIX 的业务支撑系统。

在 1989 年 UNITS 项目转移到 AT&T 的 UNIX 实验室 (USL) 时，这个 C/S 框架结构已经以“Tuxedo 系统”的名称销售了。1993 年 Tuxedo 系统被转到 Novell 公司。

在 1996 年，BEA 和 Novell 公司达成了排他协议来继续研发和出售不同平台下的 Tuxedo 系统，包括 Windows 和 UNIX 系统。2008 年 Oracle 公司收购了 BEA，Tuxedo 也转归 Oracle 旗下。

1.2.2 Tuxedo 的发展

1. 从版本 1.0 到版本 7.1

从 1983 年的 1.0 版本到 2000 年的 7.1 版本，Tuxedo 系统经过无数次的改进和扩展，目的就是为了使客户端和服务端通信模式更加多样化。Tuxedo 系统作为事实上的标准，演变为开放式 (open standard) 的在线交易处理 (OLTP) 解决方案。Tuxedo 的 4.1 版本增加了 ATMI 接口以及对事务的支持。Tuxedo 对事务的支持直接导致了 XA 接口规范的产生。在 Tuxedo 的 5.1 版本中出现了域 (Domain) 组件，它能够实现 Tuxedo 系统中多个应用程序之间的动态链。Tuxedo 7.1 版本推出了安全插件架构，这为集成第三方安全系统提供了接口。

2. 版本 8.0

Tuxedo 系统的 8.0 版本发布于 2001 年，它的总体性能比其他版本有所加强，Tuxedo 8.0 的最大特点是引进了对 CORBA 的支持。在 CORBA 域中实现多线程、统一编程和负载均衡机制。Tuxedo 8.0 可以通过 WTC (WebLogic Tuxedo Connector) 部件实现与 WebLogic 的互联。

3. 版本 8.1

Tuxedo 8.1 版本发布于 2003 年，该版本对 WTC 做了进一步的加强，集成了 XML C++ 解析器，以便更好地支持 XML 数据。Tuxedo 可以和 WebLogic 7.1 或者更高版本的 Domain 集成，进行单点安全管理。还在本地化方面做了提高，除支持英语外，还支持了日语。网络通信方面，在没有改变任何接口的同时提高了域网关的性能。

4. 版本 9.0

Tuxedo 的 9.0 版本发布于 2004 年，该版本主要是在 Web Service 方面做了进一步的加强，提供了 XML schema 和 FML 之间的双向转换功能，同时还提供了一个用于保存 Tuxedo