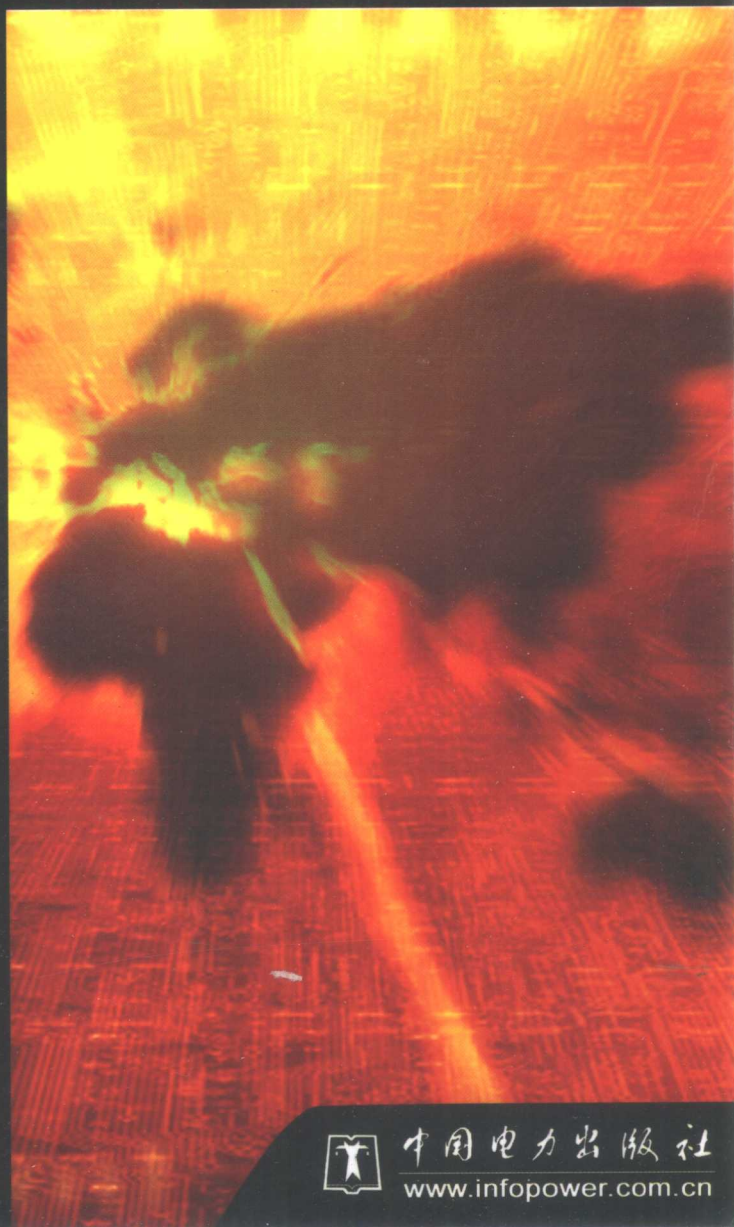


MICO

—— 开源 CORBA 的实现

Arno Puder Kay Römer 著
李炜 译



 本书附赠光盘一张



中国电力出版社
www.infopower.com.cn

MICO

——开源 CORBA 的实现

Arno Puder Kay Römer 著
李炜 译

中国电力出版社

内 容 提 要

本书从应用程序编程人员的角度来讲述 MICO，是一本集 MICO 安装指南、指导教程和参考手册为一体的不可多得的参考书籍。本书将介绍如何使用 MICO 来构建和管理专业的、大型的 CORBA 应用程序，如何通过使用 MICO 来循序渐进地把与操作系统无关的应用程序改变成分布式应用程序，如何做到与其他 CORBA 实现具有完全的互操作性等。本书共分九章，每一章针对一个核心问题展开讲述。

本书对计算机应用程序开发工程师及技术人员具有很高的参考价值。

图书在版编目 (CIP) 数据

MICO——开源 CORBA 的实现 / (美) 罗墨编著; 李炜译. —北京: 中国电力出版社, 2001

ISBN 7-5083-0609-0

I. M… II. ①罗…②李… III. 软件接口, MICO-程序设计 IV. TP311.52

中国版本图书馆 CIP 数据核字 (2001) 第 023501 号

著作权合同登记号 图字: 01-2001-1876

本书英文版原名: MICO: An Open Source CORBA Implementation

Copyright©2000 by Morgan Kaufmann Publishers.

Translation Copyright ©2001 by China Electric Power Press

All rights reserved.

本书由美国 Morgan Kaufmann 公司授权出版

中国电力出版社出版、发行

(北京三里河路 6 号 100044 <http://www.infopower.com.cn>)

三河市实验小学印刷厂印刷

各地新华书店经售

*

2001 年 7 月第一版 2001 年 7 月北京第一次印刷

787 毫米×1092 毫米 16 开本 12.5 印张 200 千字

定价 28.00 元

版 权 所 有 翻 印 必 究

(本书如有印装质量问题, 我社发行部负责退换)

56-1103

关于作者

Arno Puder 在 Kaiserslautern 大学的计算机科学系取得了学士学位,并在 Frankfurt 大学取得博士学位。现在,他在 Deutsche Telecom AG 工作,致力于与 CORBA 相关的开发项目。他的专长领域包括分布式系统和普遍 (ubiquitous) 计算环境。

Kay Römer 在 Frankfurt/Main 大学计算机科学系取得了学士学位,现在,他是一名 ETH Zurich (瑞士) 的在读博士。他尤其专长于分布式系统和操作系统。除了他的学业之外,他还从事于操作系统、分布式系统、图形视窗系统、计算机图形方面的开发项目。

致 谢

完成像 MICO 这样的开发项目，没有很多人的帮助是不可能的。首先并且最重要的是，我们要感谢所有使用 MICO 的用户，通过他们的反馈使我们提高了软件的质量。一些 MICO 用户在软件开发过程中给予了积极的帮助。我们要感谢 Christian Becker、Christoph Best、Massimo Di Gioegio、Lars Doelle、Ben Eng、Wil Evers、Karel Gardas、Leif Jacobsmeier、Andrew Metcalfe、Marcus Müller、Martin Sander、Kai_Uwe Sattlev、Andreas Schultz、Owen Taylor、Jacques Tremblay Torben Weis 和 Carsten Zerbst 的帮助，尤其感谢他们帮助我们编写了代码。特别要感谢 Frank Pilhofer，他不仅实现了可移植的对象适配器，而且还编写了 4.4 节中的内容。

Frankfurt 大学的 Verteilte Systeme 和 Betriebssysteme (VSB) 研究组织、Berkeley 大学的国际计算机学院 (ICSI) 和 ETH Zurich 的分布式系统组织给与了我们必要的计算源代码和将 MICO 发布于因特网上的机会。我们感谢他们的支持。

最后，我们要感谢我们的出版商的大力支持，他们是来自 dpunkt.verlag 的 Christa Preisendanz 和来自 Morgan Kaufmann 出版社的 Tim Cox。没有他们的帮助，MICO 的发行是很困难的。

序 言

最初几年，在早期的对象管理组织（OMG）里，几个有远见的成员花了很多时间来定义“开放”这个术语。相比之下，对象技术是一个更容易的定义。

作为 OMG 的发起成员之一，我知道，对于解决在计算环境中到处出现的企业范围的系统集成问题，对象技术将成为一个好的选择。我们认为永远不会只存在一个单一的操作系统、单一的指令集体系结构、单一的网络协议或拓扑结构，当然也永远不会只有单一的编程语言，因为解决不同的问题需要有不同的方法。我们明白，一个封装有代码和数据的框架会更好地将企业体系结构中出现的混乱的设备和软件模块化。这种框架隐藏于多种接口后面，处于继承层次体系——一种面向对象的解决方案——之内。

关于对象技术的讨论比较少（即使在 1989 年也不是讨论的主流）。然而，对于术语“开放”却有持续的讨论。在 OMG 存在的最早两年里，大家逐渐形成的一致性就像“由一个中立组织通过民主的、保守的、书面的手续给出的一个规范”。这成为 OMG 的特点。这些年来，我们使用这些手续制定各种规范，从系统分析和设计以及中间件（middleware）/应用程序服务到纵向市场规范，从航空控制和制造到生命科学和健康保健系统。

在这些年里，另一种“开放”已经占领了计算机行业的想像力。这就是“开源（open source）”运动，它通过由开发者共享的可自由获得的程序实现来提供高质量、商业级别的软件。因为其他地方有关于开源运动的大量信息（比如本书），我不会涉及定义范围。如果你可以用免费提供的软件构件构造高品质、重要任务的系统，那就足够了。

尤其在过去的两年里，这两个“开放”的定义之间发生了冲突。

令人惊讶的是，开放获得的规范（比如 **OMG** 的）开始以开放源代码的形式出现（在许可下获得，都是免费的）。我相信“开源”和“开放规范”的相互影响必然会有其他发展方式，然而最终结果却与其他情况相同：开发者、集成者和用户都有机会对规范和程序实现施加影响。

两个“开放”市场的汇合促使我们写这本书。虽然至少有一打 **OMG** 的 **CORBA** 规范的开放源代码的程序实现，**MICO** 是我知道的惟一作为指导性工具的程序实现。**Puder** 和 **Römer** 致力于构造 **MICO** 是为了写一本关于分布式系统的书，他们的努力已经为开放规范和开放源代码系统的价值取得了里程碑式的证明。基于 **MICO CORBA** 实现的商业产品的出现证明了他们工作的价值；而这本书则证明了他们的学识。在构造大型、分布式、异构型系统时，你将发现这些巨大的价值。

Richard Mark Soley 博士

主席 兼 **CEO**

对象管理组织公司

1999/12

再版序言

你将开始 CORBA 领域的愉快之旅。不，并不只是普通的字面旅行，这次你可以开始一个真正的接力远征！

你将驾驶的用于远征的工具是 MICO。虽然它比较新，却已经得到了很多关注和掌声。你不必担心：虽然也许有时地形有些崎岖，但这本指导书和书内的软件会帮助你安全通过那些有趣的分布主题。

非常幸运，你的导游 Arno Puder 和 Kay Römer 是设计和建造 MICO 的设计师。我与他们交往很久了，相信我，你完全可以相信他们。我不知道还有其他更好指导 CORBA 远征的书了。你将发现在你的旅行袋里有 MICO 工具的完整说明和蓝图。在有些情况下，这些也许会十分方便。

当然，在很多方面这次旅行是很有好处的。当你跟随旅行时，你将自己领悟 CORBA 的各个方面：

CORBA 如何帮助你构造分布式应用程序，

CORBA 内部如何运作，并且

CORBA 如何做到互操作性。

这次旅行进入了一个至今人们并不完全了解的领域。你会发现未解决的问题和挑战等着你。请与其他通过 MICO 讨论论坛的旅行者分享你的观点和经验。

现在，请系紧你的安全带。祝旅行愉快！

Kurt Geihs
Frankfurt 大学

前言

MICO 历史简介

创作 MICO 的灵感起源于想写一本分布式系统的书。但不是仅描写透明度问题和失败语义的“另一本分布式系统的书（yet another distributed system textbook YADST）”。我们想写一本书，书的主旨应该像 Andrew Tanenbaum 那本著名的操作系统（参见[12]）一样。Tanenbaum 选择了一个实用的方法，首先他实现了一个操作系统（名字是 MINIX，代表 Mini Unix），然后他围绕系统实现写了那本书。

我们的要求是对分布式系统做一些相似的事情。这样做的好处是：不仅可以解释分布式系统的抽象概念，还可以显示如何将概念变换成“代码行”。所以我们首先必须决定要实现什么。很快，我们选择了实现基于流行的 CORBA 标准的中间件（middleware）（参见[6]）。作为对 Tanenbaum 的感谢，我们决定把它命名为 MICO（最初代表 Mini CORBA）。

我们已走过了很长的路。有两种东西是 Tanenbaum 在写 MINIX 时所没有的：因特网和轰轰烈烈的开源软件运动。我们认为在因特网上公布 MICO 源代码是无害的。毕竟，我们只关心书，而不关心 MICO 本身的商业利润。对我们来说在 GNU 的通用公共许可（General Public License）下发布源代码也是不错的，虽然我们没人花费大量时间去了解它的真正主旨。

第一个开放 MICO 版本发布于 1997 年 4 月。我们始料不及的是，不断增加的用户群迫使我们不停地忙于修改错误和增加新特性。最后 MICO 不再是“小型”的了，而成为一个完全符合 CORBA

规范的版本。它超出了我们最初的设想。结果我们决定重命名 MICO，即它现在代表 MICO Is CORBA（这种循环定义的灵感来自 GNU，GNU 代表 GNU's Not Unix）。

现在，MICO 被应用于学术和商业上。学生们使用它去理解中间件平台的体系结构。公司使用它则常常免除了使用商业 CORBA 实现的昂贵的许可费用。开放组织认证 MICO 符合 CORBA 标准，这又一次表明开放源代码可以生产出有商业价值的软件。

这本书是我们实现 MICO 后的一个延续。本书通过从应用程序编程人员的角度来讲述 MICO，从而实现了一本关于分布式操作系统的书。MICO 软件现在及以后将一直可以在因特网上自由获得。这本书和附带的有预编译二进制代码的 CD 增加了可自由获得的软件的价值，它们从应用程序编程人员的角度给了你一个全面的 MICO 文档。

参加一个开放源代码开发项目是十分有益的。看到 MICO 被用于各处，并且收到陌生人士的支持真是一件让人兴奋的事。世界各地的无数的编程者都做出了或大或小的贡献，从修改小错误到编写主要构件。长久以来，MICO 已经成为了一个社会工程。开放源代码在商业软件发展中找到了合适的位置。毫无疑问，MICO 将会不断成长。

CORBA 标准提供了一种快速开发树状层次体系结构的实现路径。MICO 是一个实现 CORBA 标准的最好的 ORB，因此降低了对专有初始状态的依赖性。当选择一种 ORB 来实现树状层次体系结构初始状态时，上述特性和极低的花费（可自由获得）使得 MICO 成为我们的首选。

Michael J .Clark

软件开发经理

国家访问控制、国家数字电视中心

AT&T 宽带& Internet 服务

我们使用 MICO 作为一个基础，来为掌上引导（Palm Pilot）体系结构构造小型 CORBA。MICO 强大的模块化方法给了我们很大的帮助，而另一个重要的附带好处是它提供用户群体支持。

Andreas Paepcke

高级研究科学家

对于开放组织（Open Group），MICO 已经证明了自己是一个高品质的 CORBA 标准的实现。应该对 MICO 的现在以至将来能提供的如此好的品质，以及对这个技术所做的简明解释进行表扬。

Howard Greenwell

商业开发指导

开放组织

目 录

关于作者

致 谢

序 言

再版序言

前 言

第 1 章	简介	1
1.1	什么是 MICO?	1
1.2	支撑平台	3
1.3	困难和问题?	4
1.4	本书概述	4
第 2 章	安装	6
2.1	先决条件	6
2.2	从 CD 上运行	9
2.3	从 CD 上安装	10
2.4	从源代码安装 (Unix)	12
2.5	从源代码运行 (Windows)	15
2.6	路径映射	18
第 3 章	使用 MICO 学习 CORBA	20
3.1	分布式系统中的对象	20
3.2	CORBA 概述	22
3.3	实例程序	23

第 4 章	程序实现概述	44
4.1	ORB.....	44
4.2	接口存储库 (IR)	50
4.3	BOA	51
4.4	POA	76
4.5	IDL 编译器.....	93
4.6	编译和链接包.....	99
第 5 章	C++映射	102
5.1	使用字符串	102
5.2	无类型值	104
5.3	数组	114
5.4	联合	117
5.5	接口继承	122
5.6	向下定型 Downcasting	125
5.7	模块	126
5.8	异常	130
5.9	操作参数映射	136
第 6 章	互操作性	139
6.1	GIOP 和 IIOP	139
6.2	Lona 的 Orbix	140
6.3	Inprise 的 VisiBroker	143
第 7 章	命名服务	150
7.1	概述	150
7.2	名字服务器新进程	151
7.3	管理	152
7.4	例子	153

第 8 章	接口存储库浏览器	160
8.1	概念图	160
8.2	动态调用接口	161
8.3	解剖一个操作声明	163
8.4	一个一般 DII 接口	163
8.5	运行这个例子	164
8.6	使用 CG 编辑器	166
第 9 章	许可	167
9.1	GNU 通用公共许可证有关复制、发布和修改的 条款和条件	168
9.2	没有担保	172
9.3	最后的条款和条件如何将这些条款用到你的新程序	172
附录 A	范例程序	174
A.1	Siegel 的例子	174
A.2	Redlich 的例子	174
A.3	Henning 和 Vinoski 的例子	175
附录 B	常见问题	176
术语表		180
参考文献		184

第 1 章 简 介

1.1 什么是 MICO?

Mico 是 Mico Is COrba 的缩写。要得到更精确的解释，首先我们要解释术语 CORBA。CORBA 代表 Common Object Request Broker Architecture（公共对象请求代理程序体系结构）。它描述了一个支持分布式和异构型环境应用程序实现的中间件（middleware）平台体系结构（参见参考文献[6]）。CORBA 标准由对象管理组织（OMG）制定。OMG 是一个其成员包括超过 800 个信息软件厂商、软件开发者和用户的国际性组织。OMG 的目标是建立产业方针和对象管理规范来提供应用程序开发的通用框架。

CORBA 重要的一点是：它是一个规范（specification）而不是一个程序实现（implementation）。CORBA 提供了一个允许在分布式和异构型环境中的应用程序之间进行互操作的框架，但是关于如何实现 CORBA 标准的特定技术，它并没有给出任何规定。这个标准可以在因特网网址 www.omg.org 上自由获得。当前，存在很多用于市场不同部门的 CORBA 程序实现。

CORBA 得到了人们的关注并被用于很多不同的开发项目中，从它的文献数目的持续增长中可见一斑。然而，这些书都有一个主要缺陷：它们假定你可以得到了 CORBA 的程序实现并自己做实验。有很多商业提供的 CORBA 产品，比如，Iona 的 Orbix 或者 Inprise 的 VisiBroker。但若想买一个许可权，你就必须要花很多钱。

这样 MICO 就出现了。MICO 可以自由获得，而且它是完全符合 CORBA 标准的实现。使用 CORBA，不必事先付费，你手头现有的就足够了。本书附带的 CD 有完整的 CORBA 源代码，还有对

于不同平台的预编译的二进制代码。MICO 作为一种开放源代码项目已经十分流行，并被广泛用于各种目的。作为一个重要的里程碑，MICO 已经被开放组织认证为符合 CORBA 标准，这表明开放源代码的确可以产生有商业价值的软件。我们的目标是使 MICO 符合最新的 CORBA 标准。MICO 源代码被放在 GNU 版权通告下（参见第九章）。下面是指导 MICO 程序实现的设计原则：

MICO 的设计原则

- (1) 从小处做起：只使用标准的 Unix API 提供的函数，不需要依赖于专有的或特定的函数库。
- (2) 使用 C++ 来实现程序。
- (3) 只使用广泛提供的、非专有的工具。
- (4) 删除额外功能：只实现必需的符合 CORBA 标准的实现。
- (5) 清除那些即使是为了保证程序扩展性的内部代码得以实现的设计。

MICO 是一个完全符合 CORBA 标准的实现（有必要查阅一下 CORBA 标准中第[0.5]节的“符合 CORBA 标准”的定义；一些 CORBA 程序实现，包括一些商业软件，对这个定义并不关心）。下面是一个在本次发布中附带在 CD 上的，可在 MICO 当前版本找到的（不全面的）特性表：

MICO 实现的特性

- ☐ IDL 到 C++ 映射
- ☐ 对动态编程的全面支持：动态调用接口（Dynamic Invocation Interface, DII）、动态框架接口（Dynamic Skeleton Interface, DSI）、接口存储库（Interface Repository, IR）、动态 Any（Dynamic Any）
- ☐ 可移植的对象适配器（Portable Object Adapter, POA）
- ☐ 值类型语义（Value Type Semantics）
- ☐ 因特网 ORB 之间的协议（Internet Inter-ORB Protocol, IIOP）作为基本协议
- ☐ 在 SSL 之上的 IIOP
- ☐ 模块化的 ORB 设计：即使在运行时使用加载的模块，新的传输协议和对象适配器也可以很容易地链接到 ORB
- ☐ 支持嵌套调用
- ☐ 监听

- ❑ Any 提供接口来插入和取出在编译时程序不认识的构造类型
- ❑ Any 和 TypeCode 支持 RM-ODP 中定义的递归子类型
- ❑ 完全的 BOA 实现，包括所有的激活模式，支持对象转移、对象持久性，以及存储库的实现
- ❑ 在运行时，BOA 可以使用加载模块在客户端加载对象实现
- ❑ 支持使用 X11 应用程序 (Xt、Gtk、Tcl 和 Qt) 内部的 MICO
- ❑ 符合公用对象服务规范 Common Object Services Specification (COSS) 的可互操作的命名、事件、属性、关系、时间和交易服务
- ❑ 图形界面的接口存储库 (Interface Repository) 浏览器允许你在任意接口调用任意方法

大部分特性对于符合 CORBA 标准（类似于 DII 和 DSI）的实现是必需的。令人吃惊的是很多商业上符合 CORBA 标准的实现失去了其中一些特性！当前的 MICO 版本只支持 IDL 到 C++ 的映射。但通过使用 MICO 的 IIOP 实现，已经给了你一个使用 Java 语言实现的 CORBA 对象的机会（这在第六章中有详细的描述）。

1.2 支撑平台

可以用 GNU 工具实现 MICO，并且应该可以将 MICO 置于提供 GNU 工具的平台。本书附带的 CD 包含以下平台的二进制代码：

MICO 支持
GNU

- ❑ Sun SPARC 上的 Solaris2.6
- ❑ IBM RS/600 上的 AIX 4.2
- ❑ Intel x86 上的 Linux2.2.x
- ❑ DECAlpha 上的 Digital Unix 4.x（毫无例外）
- ❑ PA-RISC 上的 HP-UX10.20
- ❑ Windows NT 4.0 和 Windows 95