

115

352.8

CORBA 技术系列丛书

CORBA 服务

CORBA Services

[美] OMG 编著

信息产业部电信研究院

韦乐平

朗讯科技贝尔实验室

薛君教

主持编译

北京邮电大学

孟洛明



A1014974

电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书是“CORBA 技术系列丛书”的第三册,根据 OMG 的《CORBA Services》编译而成。内容包括命名服务、事件服务、通知服务、生命周期服务、持久性对象服务、事务服务、并发服务、关系服务、具体化服务、查询服务、许可服务、属性服务、时间服务、安全服务、交易对象服务和对象集合服务。本书的编译工作是由朗讯科技(中国)贝尔实验室、朗讯科技(中国)、上海国信朗讯,及深圳朗讯科技光网络(中国)共同完成。

本书可作为专业软件工程师的实用参考书,也可作为高等院校通信与计算机相关专业的教学参考书。



本书中文专有翻译出版权由美国 OMG (Object Management Group) 授予信息产业部电信研究院。信息产业部电信研究院授予电子工业出版社本书的中文专有出版权。该专有出版权受法律保护。

图书在版编目(CIP)数据

CORBA 服务/美国对象管理组织编;韦乐平,薛君敖,孟洛明译.—北京:电子工业出版社,2002.7

(CORBA 技术系列丛书)

书名原文: CORBA Services

ISBN 7-5053-7727-2

I. C… II. ①美… ②韦… ③薛… ④孟… III. 分布式处理系统—应用软件, CORBA IV. TP316.4

中国版本图书馆 CIP 数据核字(2002)第 041968 号

责任编辑:张 毅 zhangyi@phei.com.cn

特约编辑:毛联霞

印 刷:北京兴华印刷厂

出版发行:电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编:100036

经 销:各地新华书店

开 本:787×1092 1/16 印张:62 字数:1584 千字

版 次:2002 年 7 月第 1 版 2002 年 7 月第 1 次印刷

印 数:3 000 册 定价:128.00 元

版权贸易合同登记号 图字:01-2000-2759 号

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系电话:(010) 68279077。

序

当前，网络的多样化使得网络规划的任务十分具有挑战性。分布式应用通常包含由不同编程语言撰写的操作系统，在开发时，网络规划人员必须全面考虑这些因素。CORBA 定义了开发面向对象分布式应用的框架。该框架使得开发不同分布式应用的人员感觉这些应用是用同一语言在同一机器上实现的，因而使得开发工作大为简化。CORBA 在分布式环境中引入了面向对象技术的优势，使得设计人员可以将某分布式应用视作一套协作对象并在新应用中重复使用现有对象。

电信管理网（TMN）采纳了作为分布式处理技术的 CORBA，主要是由于 CORBA 在信息技术产业中的广泛应用。由于在开发基于 CORBA 接口的过程中有了较好的开发工具并积累了丰富的经验，采纳 CORBA 将会推动基于 CORBA 接口的全面应用。基于 CORBA 技术的规范提供了标准的应用接口（API）和 CORBA IDL 与编程语言的绑定，因而有利于软件的可移植性。由对象请求代理和 Inter-ORB 协议提供的可操作性方案，对客户端和服务端的可操作性进行了探讨。公共管理信息协议（CMIP）和信息模型提供了管理方和代理方的可操作性方案，而 CORBA 在对象可能分布的地方定义了对象间的相互作用。

本书对 CORBA 服务（接口和对象）做了全面的描述，这些服务支持使用和实现对象的基本功能。CORBA 服务对于创建任何分布式应用都是非常必需的，而且总是与具体的应用域无关。CORBA 服务的设计还考虑到根据某一特定环境需要的服务质量提供广泛的实现途径。中文版的 CORBA 服务书籍的出版将对促进和推广 CORBA 在中国计算机和信息技术产业中的应用起到重要作用。

Dr. J. Carl. Hsu

贝尔实验室亚太和中国区前总裁

编译者序

随着 CORBA 技术的不断发展, ITU-T 不仅批准了 CORBA 作为一个公开的标准接口应用于网管, 而且也准备将 TMF (Telemanagement Forum) 的基于粗粒度 (coarse grained) 的用于 EML-NML 的 MTMN 2.0 (multi-technology management network) 和用于 NML-SML 的 CASMIM (Connection and Service Management Information Model) 作为标准接口应用于网管。

基于 CORBA 技术的广泛应用, 我们根据与美国对象管理组织 (OMG, Object Management Group) 的协议将 CORBA 技术中的重要规范译成中文。

本书是“CORBA 技术系列丛书”的第三册。它是根据 OMG 的《CORBA Services》编译而成的。内容包括命名服务、事件服务、通知服务、生命周期服务、持久性对象服务、事务服务、并发服务、关系服务、具体化服务、查询服务、许可服务、属性服务、时间服务、安全服务、交易对象服务和对象集合服务。

本书的编译工作是由朗讯科技(中国)贝尔实验室、朗讯科技(中国)、上海国信朗讯, 以及深圳朗讯科技光网络(中国)共同合作完成。

参加本书编译工作的人员包括:

朗讯科技(中国)贝尔实验室 孙晓蓉、田文伟、汪霞、李小杰、邢江峰、宋显军、胡恩华、刘旻

朗讯科技(中国) 蔡林

上海国信朗讯 何宗应、杨旻、赵颖、郑文蕴、董兴其

深圳朗讯科技光网络(中国) 江力

参加本书审校工作的人员包括:

深圳朗讯科技光网络(中国) 王雁晨、刘宇、刘亮、梁尧成

本书译文最后由朗讯科技(美国)贝尔实验室的薛君敖博士, 以及宋显军和张臻对全书进行了最终的审校。在本书的编译过程中, 得到了朗讯科技贝尔实验室 J.Carl.Hsu 博士、高培椿博士的具体指导和及时有力的帮助。在此对以上各位的支持, 以及译校者的努力和卓有成效的工作表示衷心的感谢。

CORBA 技术还在不断发展之中, 本书在编译过程中难免会有不足之处, 诚恳期待读者的批评和指正。

薛君敖 宋显军 张臻
朗讯科技贝尔实验室

目 录



第 1 章 命名服务规范	(1)
1.1 服务描述	(1)
1.1.1 概述	(1)
1.1.2 名称	(2)
1.1.3 实例	(3)
1.1.4 设计原则	(3)
1.1.5 技术问题解决方案	(4)
1.2 命名服务模块	(4)
1.2.1 CosNaming 模块	(4)
1.2.2 名称库	(10)
附录 A 完整的 OMG IDL	(14)
A.1 CosTransactions 模块	(14)
A.2 CosTSPortability 模块	(18)
附录 B 与 TP 标准的关系	(18)
B.1 事务处理服务与 TP 标准的相关性	(18)
B.2 X/Open TX 接口支持	(19)
B.3 X/Open 资源管理器支持	(20)
B.4 与事务处理协议的互操作	(23)
B.5 ODMG 模式	(26)
第 2 章 事件服务规范	(29)
2.1 服务描述	(29)
2.1.1 概述	(29)
2.1.2 事件通信	(29)
2.1.3 实例	(30)
2.1.4 设计原则	(31)
2.1.5 技术问题解决方案	(31)
2.1.6 服务质量	(32)
2.1.7 通用事件通信	(33)
2.2 模块和接口	(34)
2.2.1 CosEventComm 模块	(34)
2.2.2 事件通道	(36)
2.2.3 CosEventChannelAdmin 模块	(38)
2.2.4 类型化事件通信	(42)
2.2.5 CosTypedEventComm 模块	(44)
2.2.6 类型化事件通道	(45)
2.2.7 CosTypedEventChannelAdmin 模块	(45)
2.2.8 组合事件通道和过滤	(49)
2.2.9 查找事件通道的策略	(49)
附录 A 实现类型化事件通道	(49)

A.1 概述	(49)
附录 B 事件通道使用实例	(50)
第 3 章 通知服务规范	(53)
3.1 服务描述	(53)
3.1.1 概述	(53)
3.1.2 一致性问题	(54)
3.2 体系结构特征	(55)
3.2.1 概述	(55)
3.2.2 结构化事件	(62)
3.2.3 用过滤器对象进行事件过滤	(65)
3.2.4 默认的过滤器约束语言	(69)
3.2.5 服务质量管理	(79)
3.2.6 共享订阅	(90)
3.2.7 过滤类型化事件	(92)
3.2.8 事件类型库	(95)
3.2.9 互操作性问题	(96)
3.3 模块与接口	(96)
3.3.1 CosNotification 模块	(96)
3.3.2 CosNotifyFilter 模块	(102)
3.3.3 CosNotifyComm 模块	(113)
3.3.4 CosNotifyChannelAdmin 模块	(122)
3.3.5 CosTypedNotifyComm 模块	(149)
3.3.6 CosTypedNotifyChannelAdmin 模块	(150)
附录 A 事件类型库	(164)
A.1 事件类型元模式	(164)
A.2 其他功能	(166)
A.3 MODL 模式	(166)
A.4 生成的 IDL	(167)
附录 B 完整的 IDL	(173)
附录 C 对 CORBA 的改变	(193)
C.1 对 CORBA 标准的改变	(193)
C.2 没有涉及的 RFP 需求	(194)
第 4 章 生命周期服务规范	(195)
4.1 服务描述	(195)
4.1.1 概述	(195)
4.1.2 对象生命周期的客户模式	(197)
4.1.3 生成库查找器	(199)
4.1.4 设计原理	(200)
4.1.5 一些技术问题的解决	(200)
4.2 生命周期接口	(201)
4.2.1 CosLifeCycle 模块	(201)
4.2.2 实现生成库	(207)
4.2.3 目标对生成库和生成库查找器的使用	(208)

4.2.4	生命周期服务的总结	(209)
附录 A	复合生命周期规范	(210)
A.1	总览	(210)
A.2	关键特性	(210)
A.3	服务结构	(210)
A.4	接口总览	(211)
A.5	复合生命周期操作	(212)
A.6	CosCompoundLifeCycle 模块	(215)
A.7	CosLifeCycleContainment 模块	(224)
A.8	CosLifeCycleReference 模块	(226)
附录 B	参考文献	(228)
附录 C	过滤器	(228)
C.1	作为属性的资源	(229)
C.2	约束表达式	(229)
C.3	约束表达式的 BNF	(230)
附录 D	管理	(231)
D.1	联盟	(232)
D.2	LifeCycleService 模块的一个例子	(234)
附录 E	对 PCTE 对象的支持	(237)
E.1	总览	(238)
E.2	对象的创建	(238)
E.3	对象的删除	(239)
E.4	对象的拷贝	(239)
E.5	对象的移动	(240)
第 5 章	持久性对象服务	(241)
5.1	服务描述	(241)
5.1.1	概述	(241)
5.1.2	目标和属性	(242)
5.1.3	服务视图	(244)
5.2	持久性服务模块	(245)
5.2.1	服务结构	(245)
5.2.2	CosPersistencePID 模块	(246)
5.2.3	CosPersistencePO 模块	(248)
5.2.4	CosPersistencePOM 模块	(251)
5.2.5	持久性数据服务概述	(254)
5.2.6	CosPersistencePDS 模块	(255)
5.2.7	直接存取 (PDS_DA) 协议	(256)
5.2.8	CosPersistencePDS_DA 模块	(257)
5.2.9	ODMG-93 协议	(264)
5.2.10	动态数据对象 (DDO) 协议	(264)
5.2.11	CosPersistenceDDO 模块	(266)
5.2.12	其他协议	(267)
5.2.13	Datastore: CosPersistenceDS CLI 模块	(268)
5.2.14	其他 Datastore	(276)

5.2.15 标准兼容性	(276)
第 6 章 事务服务规范	(277)
6.1 服务描述	(277)
6.1.1 概述	(277)
6.1.2 服务描述	(284)
6.1.3 服务构架	(289)
6.2 事务服务接口	(289)
6.2.1 概述	(289)
6.2.2 用户视角	(301)
6.2.3 实现者视角	(311)
附录 A OMG 的完整 IDL 定义	(326)
A.1 CosTransactions 模块	(326)
A.2 CosTSPortability 模块	(330)
附录 B 与 TP 标准之间的关系	(330)
B.1 事务服务和 TP 标准之间的关系	(330)
B.2 X/Open TX 接口的支持	(330)
B.3 X/Open 资源管理者的支持	(332)
B.4 和事务性协议的互操作	(335)
B.5 ODMG 模式	(338)
第 7 章 并发服务规范	(339)
7.1 服务描述	(339)
7.1.1 概述	(339)
7.1.2 并发控制的基本概念	(339)
7.1.3 锁定模型	(341)
7.1.4 两阶段事务锁定	(343)
7.1.5 嵌套事务	(343)
7.2 模块和接口	(344)
7.2.1 CosConcurrencyControl 模块	(344)
7.2.2 LockCoordinator 接口	(346)
7.2.3 LockSet 接口	(346)
7.2.4 TransactionalLockSet 接口	(347)
7.2.5 LockSetFactory 接口	(348)
第 8 章 关系服务规范	(349)
8.1 服务概述	(349)
8.1.1 概述	(349)
8.1.2 服务结构	(353)
8.2 关系服务模式	(356)
8.2.1 基础关系模式	(356)
8.2.2 CosObjectIdentity 模块	(360)
8.2.3 CosRelationships 模块	(361)
8.2.4 相关对象图	(369)
8.2.5 CosGraphs 模块	(373)
8.2.6 特殊关系	(379)

8.2.7 CosReference 模块	(381)
第 9 章 具体化服务规范	(383)
9.1 服务描述	(383)
9.1.1 概述	(383)
9.1.2 服务结构	(383)
9.1.3 对象和接口层次体系	(387)
9.1.4 接口概述	(388)
9.2 具体化服务模块	(389)
9.2.1 CosExternalization 模块	(389)
9.2.2 CosStream 模块	(392)
9.2.3 特定的具体化关联	(400)
9.2.4 CosExternalizationContainment 模块	(401)
9.2.5 CosExternalizationReference 模块	(402)
9.2.6 标准流数据格式	(403)
第 10 章 查询服务规范	(405)
10.1 服务描述	(405)
10.1.1 概述	(405)
10.1.2 设计原则	(405)
10.1.3 结构	(406)
10.1.4 查询语言	(408)
10.1.5 主要特点	(410)
10.1.6 服务结构	(410)
10.2 查询服务接口	(412)
10.2.1 集合模型	(412)
10.2.2 CosQueryCollection 模块	(413)
10.2.3 查询框架模型	(417)
10.2.4 CosQuery 模块	(419)
附录 A 参考文献	(423)
第 11 章 许可服务规范	(425)
11.1 服务描述	(425)
11.1.1 背景信息	(425)
11.1.2 许可服务概览	(430)
11.1.3 许可系统中的关键组成部分	(430)
11.1.4 CORBA 环境下的许可	(432)
11.1.5 设计原则	(433)
11.2 许可服务接口	(434)
11.2.1 许可服务接口	(434)
11.2.2 CosLicensing 模块	(437)
附录 A 参考文献	(441)
附录 B 其他服务的使用	(441)
B.1 属性服务	(441)
B.2 关系服务	(443)
B.3 安全服务	(443)

附录 C 实现问题	(444)
C.1 生产者客户实现问题	(444)
C.2 异步事件	(444)
C.3 伪代码	(444)
附录 D 质询机制	(445)
D.1 默认情况	(445)
D.2 可供选择的办法	(446)
第 12 章 属性服务规范	(447)
12.1 服务描述	(447)
12.1.1 概述	(447)
12.1.2 关键特性的总结	(449)
12.2 属性服务接口	(449)
12.2.1 CosPropertyService 模块	(449)
12.2.2 PropertySet 接口	(452)
12.2.3 PropertySetDef 接口	(456)
12.2.4 PropertiesIterator 接口	(459)
12.2.5 PropertyNamesIterator 接口	(460)
12.2.6 PropertySetFactory 接口	(461)
12.2.7 PropertySetDefFactory 接口	(461)
附录 A OMG IDL	(462)
第 13 章 时间服务规范	(467)
13.1 服务描述	(467)
13.1.1 概述	(467)
13.1.2 通用对象模型	(468)
13.1.3 基本时间服务	(469)
13.2 时间服务接口	(475)
13.2.1 TimeService 接口	(475)
13.2.2 定时器事件服务	(476)
13.2.3 定时器事件处理器	(478)
13.2.4 定时器事件服务	(479)
13.2.5 一致性	(480)
附录 A 实现指南	(480)
A.1 要获取安全时间必须遵守的准则	(480)
附录 B 时间的管理	(481)
B.1 操作和强制审查的保护	(481)
B.2 代理和时间不确定性	(481)
附录 C OMG IDL	(482)
C.1 时间服务	(482)
C.2 定时器事件服务	(484)
附录 D 用户注意事项	(485)
D.1 代理和时间	(485)
D.2 在网络中传递时间	(485)
附录 E 扩展例子	(486)

E.1 对象模型	(486)
E.2 扩展总结	(486)
E.3 数据类型	(487)
E.4 异常	(487)
E.5 FriendlyTime 对象	(487)
E.6 扩展的时间服务	(488)
E.7 结束语	(488)
附录 F 参考文献	(488)
第 14 章 安全服务规范	(489)
14.1 概述	(489)
14.1.1 安全问题简介	(489)
14.1.2 规范简介	(494)
14.2 接口	(499)
14.2.1 安全性参考模型	(499)
14.2.2 安全体系结构	(519)
14.2.3 应用开发者接口	(548)
14.2.4 管理员接口	(581)
14.2.5 实现者安全接口	(602)
14.3 协议和机制	(622)
14.3.1 安全互操作协议	(622)
14.3.2 安全 ORB 间协议 (SECIOP)	(645)
14.3.3 基于 SECIOP 的 CSI 协议	(662)
14.3.4 SPKM 协议	(668)
14.3.5 GSS Kerberos 协议	(670)
14.3.6 CSI-ECMA 协议	(672)
14.3.7 SSL 和 CORBA 安全机制的集成	(705)
14.3.8 具有安全功能的 DCE-CIOP	(706)
附录 A 参考文献	(713)
附录 B 关于安全业务的完整 OMG IDL 文件	(714)
B.1 介绍	(714)
B.2 通用安全数据模块	(715)
B.3 应用接口-级别 1	(722)
B.4 应用接口-级别 2	(723)
B.5 安全管理接口	(727)
B.6 用于非否认的接口	(730)
B.7 安全可替换服务接口	(733)
B.8 安全 ORB 间协议 (SECIOP)	(736)
B.9 SSL	(740)
B.10 安全 DCE CIOP	(741)
B.11 标准数据类型的取值	(741)
附录 C 和其他服务的关系	(745)
C.1 介绍	(745)
C.2 同对象服务和通用设施的一般关系	(745)
C.3 同专用对象服务的关系	(745)

C.4 同一般设施的关系	(746)
附录 D 一致性的细节和陈述	(747)
D.1 简介	(747)
D.2 一致性需求	(749)
D.3 安全功能级别 1	(749)
D.4 安全功能性级别 2	(751)
D.5 安全功能可选包	(752)
D.6 安全可替换性	(753)
D.7 安全互操作性	(754)
D.8 一致性陈述	(757)
附录 E 可信任系统导引	(762)
E.1 介绍	(762)
E.2 防止威胁的保护	(763)
E.3 用于结构性模型的导引	(767)
E.4 用于应用接口模型的导引	(772)
E.5 用于管理模型的导引	(774)
E.6 安全对象实施模型	(775)
附录 F 非本规范设施	(778)
F.1 简介	(778)
F.2 不相似域之间的互操作性限制	(778)
F.3 非面向会话的 SECIOP 协议	(779)
F.4 强制性安全机制	(779)
F.5 专用安全策略	(779)
F.6 其他稽核服务	(779)
F.7 可能的增强性	(780)
F.8 在使用非否认时的可互操作性	(780)
F.9 稽核文件可互操作性	(781)
F.10 管理	(781)
F.11 引用限制	(781)
F.12 消息保护的端控制	(781)
F.13 高级授权特性	(782)
F.14 交迭和分级域	(782)
F.15 基于能力的访问控制	(784)
F.16 非否认服务	(785)
附录 G 互操作导引	(785)
G.1 简介	(785)
G.2 在 IOR 中用于机制 TAG 定义的导引	(785)
G.3 SECIOP 举例	(786)
第 15 章 交易对象服务	(789)
15.1 服务描述	(789)
15.1.1 概述	(789)
15.1.2 概念与数据类型	(791)
15.1.3 协调工作机制	(802)
15.1.4 异常处理	(805)

15.2 交易对象服务接口	(809)
15.2.1 抽象接口	(809)
15.2.2 功能接口	(810)
15.2.3 服务类型库	(831)
15.2.4 动态属性评估接口	(837)
15.2.5 一致性标准	(839)
附录 A OMG IDL	(843)
A.1 基于 CORBA OMG IDL 的交易功能规范	(843)
A.2 OMG 交易功能模块	(843)
A.3 动态属性模块	(853)
A.4 服务类型库模块	(854)
附录 B OMG 强制语言 BNF	(857)
B.1 介绍	(857)
B.2 语言要素	(857)
B.3 强制语言 BNF	(859)
附录 C OMG 强制处方语言	(861)
C.1 OMG 强制处方语言	(861)
C.2 处方构造结构	(861)
C.3 范例	(862)
第 16 章 对象集合服务	(863)
16.1 服务描述	(863)
16.1.1 概述	(863)
16.1.2 服务结构	(863)
16.1.3 组合集合	(869)
16.1.4 受限访问集合	(872)
16.2 模块和接口	(874)
16.2.1 CosCollection 模块	(874)
16.2.2 Collection 接口	(878)
16.2.3 OrderedCollection 接口	(884)
16.2.4 SequentialCollection 接口	(887)
16.2.5 SortedCollection 接口	(891)
16.2.6 EqualityCollection 接口	(892)
16.2.7 KeyCollection 接口	(896)
16.2.8 EqualityKeyCollection 接口	(903)
16.2.9 KeySortedCollection 接口	(903)
16.2.10 EqualitySortedCollection 接口	(905)
16.2.11 EqualityKeySortedCollection 接口	(907)
16.2.12 EqualitySequentialCollection 接口	(907)
16.2.13 KeySet 接口	(908)
16.2.14 KeyBag 接口	(909)
16.2.15 Map 接口	(909)
16.2.16 Relation 接口	(912)
16.2.17 Set 接口	(912)
16.2.18 Bag 接口	(912)

16.2.19	KeySortedSet 接口	(912)
16.2.20	KeySortedBag 接口	(913)
16.2.21	SortedMap 接口	(913)
16.2.22	SortedRelation 接口	(914)
16.2.23	SortedSet 接口	(914)
16.2.24	SortedBag 接口	(914)
16.2.25	Sequence 接口	(914)
16.2.26	EqualitySequence 接口	(915)
16.2.27	Heap 接口	(915)
16.2.28	抽象的 RestrictedAccessCollection 接口	(915)
16.2.29	具体的限制访问 Collection 接口	(916)
16.2.30	CollectionFactory 接口	(920)
16.2.31	迭代器接口	(930)
16.2.32	函数接口	(959)
16.2.33	Command 和 Comparator 接口	(962)
附录 A	OMG 对象查询服务	(963)
A.1	对象查询服务差别	(963)
A.2	其他 OMG 对象服务定义集合	(969)
A.3	OMG 永久对象服务	(969)
A.4	OMG 对象并发服务	(969)
附录 B	和其他标准的关系	(969)
B.1	ANSI 标准模板库	(969)

第 1 章 命名服务规范

1.1 服务描述

1.1.1 概述

名称与对象的关联称为名称绑定 (name binding)，而定义名称绑定总要提及命名上下文 (naming context)。命名上下文是一个对象，包含有一套名称绑定，这其中的每一个名称都是惟一的。相同或不同上下文中的一个对象可以同时被赋予不同的名称，但是，并不需要对所有的对象命名。

解析名称 (resolve a name) 是确定在指定上下文中与名称相关联的对象。绑定名称 (bind a name) 则是在指定上下文中创建名称绑定。名称解析总是与上下文有关的，没有绝对的名称。

鉴于上下文与任何其他对象类似，它也可以在命名上下文中被赋予一个名称。其他上下文中的绑定上下文构成一个命名图 (naming graph)，该图是由节点和标注边界组成的有向图，节点即代表上下文。

命名图允许多个复杂名称指向一个对象。在命名图中指定一个上下文，会有一个名称序列指向一个对象，这个名称序列 (称为复合名，compound name) 规定了在命名图中解析名称的路径。如图 1-1 所示。

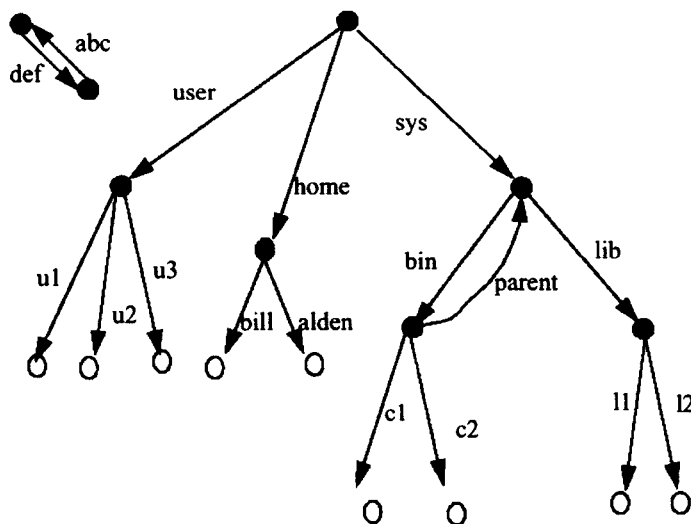


图 1-1 命名图

1.1.2 名称

命名上下文定义的许多操作采用名称作为参数。名称是有结构的，它是一个分量（component）有序序列。

具有一个分量的名称称为单名（simple name），具有多个分量的名称则称为复合名。除末尾分量外的其余每个分量均用于命名一个上下文，而末尾分量则指示绑定对象。

分量序列用以下形式表示：

< component1 ; component2 ; component3 >

注：分号（；）仅是本书中使用的符号，并不意味名称是由分号分隔的字符序列。

名称分量包含有两种属性：标识属性（identifier attribute）和类别属性（kind attribute）。标识属性和类别属性均用 IDL 字符串。

类别属性采用与语法无关的方式，增加了名称的描述能力。举例来讲，类别属性的值包括 `c_source`、`object_code`、`executable`、`postscript` 或“ ”。命名系统不采取任何方式解释、分配或管理这些值，而软件系统可以在更高层制定策略来使用和管理这些值。这一特性满足了应用中采用语法命名习惯区分相关对象的需求。例如，UNIX 系统使用 `.c` 和 `.o` 等后缀。应用软件（如 C 编译器）依据这些语法常规转换名称（例如，将 `foo.c` 转换为 `foo.o`）。

面对国际化问题时，名称语法的缺陷显得尤为突出。不依赖名称语法常规的软件在本地化时，即使本地语言语法常规不同，也不需要修改。这一点与依赖语法常规的软件截然不同，后者必须经过修改来适应新的规则。

为避免名称语法不一致问题，命名服务（Naming Service）采用结构化表（structural form）来处理名称。这样，就不存在什么规范语法或间隔符问题了，前提是各种程序和系统能采用适当的方法将名称从当前的表示映射到结构表中。

1.1.2.1 名称库

为保证在不影响现有客户的情况下顺利转换名称表示，希望在客户代码中隐藏名称表示。理想情况是名称本身也是 OMG IDL 对象。然而名称也必须是轻量（lightweight）实体，可以在内存中有效地产生和处理，也可以作为参数来传递其值。为了简化名称处理和提供表示的独立性，可以利用名称库在程序中表示名称。但是，需要指出的是，不必利用名称库来使用命名服务中的基本操作。

名称库将名称作为伪对象实现，客户调用伪对象的方法和调用普通对象的方法是同样的。库名称采用伪 IDL 描述。名称库支持两种伪 IDL 接口：LNameComponent 接口和 LName 接口。LNameComponent 接口定义获取和设置与名称分量标识属性和类别属性相关的操作。LName 接口操作包括处理库名称和库名称分量伪对象，产生和转换结构，该结构可作为参数传递给一般对象请求。