



本书配套光盘内容包括：
与本书配套的电子书



计算机语音通信核心技术内幕 ——CT Media 存储器扩展技术指南

21 世纪计算机语音通信开发技术丛书编委会 编写



北京希望电子出版社

Beijing Hope Electronic Press
www.bhpe.com.cn

院图书馆
章

TN912.3
01



计算机语音通信核心技术内幕 ——CT Media 存储器扩展技术指南

21 世纪计算

委会 编写



北京希望电子出版社
Beijing Hope Electronic Press
www.bhpe.com.cn

内 容 简 介

计算机语音通信技术(CT)是新世纪中最热门的技术。Dialogic 公司是开放式 CT 技术的全球先行者。Dialogic 产品广泛应用于语音、传真、数据、语音识别、声音合成、互联网电话和呼叫中心管理等一系列商业领域。使用 Dialogic 公司提供的模块化和无阻塞的部件与技术服务,开发商可以迅速灵活地设计出融合语音与数据网络的商业通信方案,以满足不断增长的社会需求。为开发和设计新的商业通信方案,维护、管理和拓展已建立的通信系统,抓住商机发展通信事业,都要求更多地了解和精晓通信和网络通信中的语音技术。为通信领域开发商、技术支持和维护人员以及技术用户,提供系统、完善和最新的技术资料,我们组织了本套丛书,共 6 本,本书是其中之一。

CT Media 使得计算机语音技术在客户-服务器环境中得到开发,并使语音识别及音频压缩成为可能。本书着重介绍了这一新技术的最基本的内容,即 CT Media 存储类型以及同其他的兼容部件集成的方法。

全书共分九章,内容包括:CT Media 简介、CT Media 存储管理模型、建立开发环境、编写组织策略、编写存储策略、故障诊断、组织策略接口、存储策略接口、键值集和符号。其中第七章到第九章详细介绍了各函数的用法和示例,为读者创建 CT Media 的应用提供了学习的捷径。

本书内容新、实用性、可操作性和指导性强,层次清晰,内容详尽,不但是从事语音、传真、数据、语音识别、声音合成、互联网电话和呼叫中心管理的广大开发与编程人员、技术支持和管理与维护人员重要的指导书,同时也是高等院校相关专业师生教学、自学参考书和国内科研院所各图书馆重要馆藏图书。与本书配套的光盘内容包括:与本书配套的电子书。

系 列 书 名: 21 世纪计算机语音通信开发技术丛书(3)
书 名: 计算机语音通信核心技术内幕——CT Media 存储器扩展技术指南
文 本 著 者: 21 世纪计算机语音通信开发技术丛书编委会 编写
责 任 编 辑: 柴文强
CD 制 作 者: 希望多媒体创作中心
CD 测 试 者: 希望多媒体测试部
出 版、发 行 者: 北京希望电子出版社
地 址: 北京海淀路 82 号, 100080
网 址: www.bhp.com.cn
E-mail: lwm@hope.com.cn
电 话: 010-62562329, 62541992, 62637101, 62637102
010-62633308, 62633309 (图书发行, 技术支持)
010-62613322-215 (门市) 010-62531267 (编辑部)
经 销: 各地新华书店、软件连锁店
排 版: 希望图书输出中心
CD 生 产 者: 北京中新联光盘有限责任公司
文 本 印 刷 者: 北京媛明印刷厂
开 本 / 规 格: 787 毫米×1092 毫米 1/16 开本 16.25 印张 369 千字
版 次 / 印 次: 2000 年 7 月第 1 版 2000 年 7 月第 1 次印刷
印 数: 0001-3000 册
本 版 号: ISBN 7-900044-83-3/TP·83
定 价: 55.00 元(1CD, 含配套书)

说明: 凡我社光盘配套图书若有缺页、倒页、脱页、自然破损, 本社发行部负责调换。

序

计算机语音通信技术 (CT) 是新世纪中最热门的技术。Dialogic 公司是开放式 CT 技术的全球领导者, 是一个提供开放的、高品质的、基于标准的电信和计算机语音集成部件的国际供应商, 在该领域占有全球 64% 的市场份额。Dialogic 产品广泛应用于语音、传真、数据、语音识别、声音合成、互联网电话和呼叫中心管理等一系列商业领域。使用 Dialogic 公司提供的模块化和无阻塞的部件与技术服务, 开发商可以迅速灵活地设计出融合语音与数据网络的商业通信方案, 以迎合不断增长的社会需求。目前, 基于 Dialogic 产品的系统已开始进入国内不少部门, 被用于管理电话、传真和由计算机通过有线和无线网络来应答的多媒体呼叫系统。

随着通信市场的不断增长, 通信技术以及领先的 Dialogic 技术已深入到与每一位社会人息息相关的程度。开发和设计新的商业通信方案, 维护、管理和拓展已建立的通信系统, 抓住商机发展通信事业, 都要求更多地了解和精晓通信和网络通信中的语音技术。为通信领域开发商、技术支持和维护人员以及技术用户, 提供系统、完善和最新的技术资料, 无疑已迫在眉睫。

Dialogic 公司的 CT 成为通信行业最具特点的技术领域。为满足国内相关领域技术用户、系统管理、网络维护、应用编程和开发人员的要求, 我社组织了本丛书——21 世纪计算机语音通信开发技术丛书。本丛书由以下 6 种图书组成。

1. 计算机语音通信核心技术内幕——CT Media 程序设计参考手册 (第一卷)。本书分一、二两卷, 主要针对 CT Media 系统管理员和应用程序开发人员在 Windows NT 环境下开发应用程序而编写的, 是一本 API 函数参考手册, 支持面向对象和 Client/Server 编程技术。

Dialogic 公司研发的 CT Media for Windows NT 是第一个开放的软件平台, 用来设计标准的电信服务以支持各种开发商的信息、交互式语音响应系统、传真、自动呼叫转发和其他应用。CT Media 是一种 Client/Server 方式的资源管理软件, 它使得按照 ECTF S.100 和 TAPITM 标准设计的多个应用程序可以共享公共的电脑语音(CT)服务器和现存的技术。另外, CT Media 还为 SCBus 和 H.100 技术硬件提供了一个开放的接口, 允许将新技术加到服务器上, 而不需要改变现存的应用程序。

本书全面地介绍了如何利用 CT Media API 函数对 CT Media 的管理、应用程序配置文件、会议、连接、容器、CT Media 函数、组、KVset 与其它数据类型、系统呼叫路由器、会话、符号等进行操作。提供了相关 API 函数的详细说明、完成事件、可能返回的错误、所使用的数据结构和参数类型以及相应的示例程序, 从而使 CT Media 用户能够方便地编写应用程序。本书是一本开发工具书。

2. 计算机语音通信核心技术内幕——CT Media 程序设计参考手册 (第二卷)。本书与第一卷构成一套完整的手册。它全面介绍了数据通信领域的八个应用程序编程接口, 它们是: 自动语音识别 (ASR: Automatic Speech Recognition)、呼叫频道资源 (CCR: Call Channel Resource)、传真接收器资源 (Facsimile Receiver Resource)、传真发送器 (Facsimile Sender)、播放器 (Player)、记录器 (Recorder)、信号检测器 (SD: Signal Detector)、信号发生器 (SG: Signal Generator)。

本书详细地介绍了上述各类编程接口的属性和参数、常数和符号、完成事件和主动提供的事件、函数引用和出错码, 并且给出了相应的示例程序代码。

3. 计算机语音通信核心技术内幕——CT Media 应用程序开发指南。Dialogic Computer Telephony Server (简称 CT Server 或 Server) 和相关的 CT Media 是一个客户/服务端开发环境, 用于开发计算机电话应用程序和核心信号处理技术。本书为计算机电话应用程序开发人员介绍了如何使用和掌握 CT Media。

全书分 11 章, 讨论了 CT Media 基本知识、编写一个 CT 应用程序、应用程序前的准备、将应用程序连接到 CT Server、在应用程序中获得呼叫、在程序中录制消息、检测应用程序中的信号、运行应用程序、

CT Sim、演示 WinVote 应用程序以及异步编程等。全书条理清晰，讲解与示例相结合，有助于读者掌握 CT Media 的使用。

4. 计算机语音通信核心技术内幕——CT Media 存储器扩展技术指南。计算机电话 (Computer Telephony: CT) 和相关的计算机电话服务软件开发包使得在客户机/服务器环境中开发计算机电话应用程序成为可能。本书针对 CT Media 存储扩展技术讨论了 CT Media 基础、CT Media 存储管理模型、建立开发环境、编写组织策略、编写存储策略、故障诊断、组织策略接口、存储策略接口和 KVset 符号等内容。

5. 计算机语音通信核心技术内幕——CT Connect: C 程序设计指南。本书是 CT Connect (CTC) 编程参考手册。CT Connect 是 Dialogic CT 分部研制开发的一套用于呼叫控制的软件。CT Connect 的服务器软件部分是基于 Windows NT 或 SCO UNIX 的，它通过与交换机的连接完成复杂的呼叫控制及监视功能。CT-Connect 的服务器软件部分完成与多种交换机的 CTI (Computer Telephony Integration) Link 的通信。它可将不同交换机的 CTI Link 的不同的协议及消息映射为同样的基于 CSTA 的消息，并管理服务器及相应应用程序间电话服务请求及状态消息的交换。通过 CT Connect, OEM 厂家、应用程序开发商及集成商很容易就可以在他们的应用程序中完成完善的电话路由及监视功能。

本书主要面向需要编写 CTC 应用程序的程序员。有两种编写 CTC 应用程序的编程界面：C 语言格式应用程序接口和 JAVA 语言的应用程序接口。本书是专门针对 C 语言格式编程接口进行描述的。

全书共分为两大部分，第一部分三章，具体对 CTC 应用程序接口例行程序的机制、数据结构和多线程设计进行了描述，给出了每个可调用例行程序参数和用法的具体讲解，并针对返回的错误信息进行了说明。第二部分附录，针对不同厂商生产的交换机描述了特定类型交换机的例行程序。首先对通用例行程序进行了概括说明，然后分别针对 CSTA 交换机、朗讯 DEFINITY 交换机和 Nortel Meridian 交换机进行了指定例行程序的详细说明。

全书层次清晰，内容详尽，简洁明了，适合于有一定编写电话应用程序经验的程序员在编写 CTC 应用程序时参考。

6. 计算机语音通信核心技术内幕——GDK 3.2 程序设计。本书讨论 GammaLink Developer's Kit (GDK) 的编程。GDK 3.2 是 Dialogic 最新软件开发包之一。Dialogic 软件开发包为应用系统的开发提供了完整的语音处理开发环境，它的运行稳定性以及丰富的功能最适合计算机语音处理系统。规模的伸缩性使得同一应用程序既适用于小系统，也适用于大系统。完整的软件开发包包括函数库、驱动程序和固件，以及附加的应用程序，如固件下载程序和安装程序等。软件开发包为不同种类的 Dialogic 产品提供了完整的集成开发环境，包括语音处理、传真、文本语音转换、语音识别、多方会议、交换以及不同的电话网络接口。由于这些功能均使用相同的程序设计风格，以及共享许多相同的函数调用，因此，能使应用系统天衣无缝地集成以上功能。所有 Dialogic 板卡均使用同一个下载程序。Dialogic 软件开发包的可靠性在世界各国都得到证实，且得到越来越多用户的支持。

本书详细介绍了 GammaLink 传真系统的结构、编程模型、相关应用编程接口 (API) 函数调用及传真应用编程方面的指令，适用于在 Microsoft Windows NT 平台上进行传真软件和语音软件的应用程序开发。本书共分 8 章，分别介绍了传真技术和 GammaLink 传真系统的结构，描述配置命令和队记录编程，编程模型、PEB 与 SCbus 下的 API 编程，以及传真状态文件等内容。

本丛书集中讨论了 Dialogic 公司 CT 核心技术，反映了 90 年代末、21 世纪初 CT 技术的最新结果，内容定位与国内外技术和产品市场同步，技术内涵高、指导性强，特别是从事语音、传真、数据、语音识别、声音合成、互联网电话和呼叫中心管理的广大开发与编程人员、技术支持和管理与维护人员重要技术参考书，同时也是高等院校相关专业师生教学、自学参考书和国内科研院所各图书馆重要馆藏图书。

藉本丛书出版之际，特别感谢 Dialogic 公司副总裁恰克·霍斯先生，Dialogic 公司通信产品部资深经理霍华德·巴勃博士，本丛书就是在他们的大力帮助和协调下才得以完成。感谢 Dialogic 公司产品经理约翰·拉道博士、Dialogic 公司 CT 部资深经理蒂恩·却伦波博士、MIT CT 实验室主任马克·威尔斯博士、MIT CT 实验室资深研究员托蒂·鲍勃博士，以及资深记者阿道夫·吉尔先生，由于他们的技术指导和全

力参与，本丛书才得以及时完稿。还要感谢黄太成、龙启铭、陆卫民、张中民、李国华、柴文强等，是他们夜以继日的辛勤劳动，使本丛书及时面市。真诚感谢参与本丛书编写的全体专家和技术人员，以及编辑、美工设计人员和录排人员、光盘制作人员等，是他们的加班、加点、忘我的工作，才使本丛书如期付梓出版。

因出版时间紧迫，书中错误在所难免，敬请读者谅解，并请拨冗指正，以期再版时修订。

21 世纪计算机语音通信开发技术丛书编委会

2000 年 6 月

目 录

第1章 CT Media 简介	1	5.1 序言	57
1.1 CT 媒体概述	1	5.2 示范代码的限制	57
1.2 ECTF S.100 服务器	1	5.3 安装	58
1.3 CT Media 的组成	2	5.4 加入框架存储策略	60
1.4 CT Media 运行系统组成部分	3	5.5 容器的创建与撤消	74
1.5 用 CT Media 数据类型工作	5	5.6 数据对象创建与撤消	86
第2章 CT Media 存储管理模型	11	5.7 在数据对象中读、写和搜索	97
2.1 概述	11	5.8 完成存储策略	103
2.2 Media 数据对象	12	第6章 故障诊断	112
2.3 容器子系统	14	6.1 测试代码	112
2.4 容器子系统接口	15	6.2 运行调试	112
第3章 建立开发环境	18	6.3 跟踪	113
3.1 实现厂商特有组织或存储策略	18	6.4 清除	114
3.2 用 Visual C++ 建立 DLL	18	6.5 打开容器和资源	114
3.3 增加要求的库	18	第7章 组织策略接口	115
3.4 改变 C++ 设置	19	7.1 介绍	115
3.5 改变链接设定	19	7.2 厂商函数	115
3.6 设置文件路径	19	7.3 CT Media 定义的 (SC) 函数	146
第4章 编写组织策略	20	第8章 存储策略接口	171
4.1 导言	20	8.1 序言	171
4.2 安装	21	8.2 厂商函数	171
4.3 加入骨架组织策略	23	8.3 CT Media 定义的 (SC) 的函数	221
4.4 数据对象的创建与撤消	36	第9章 KVset 和符号	226
4.5 在数据对象流中写、读和搜寻	46	9.1 概述	226
4.6 其它组织策略函数	56	9.2 键值集函数参考	226
第5章 编写存储策略	57	9.3 符号函数参考	240

第 1 章 CT Media 简介

1.1 CT 媒体概述

Dialogic 计算机电话学服务器 (Computer Telephony Server, CT Server, 或称为 Server) 和相关的计算机电话学服务器软件开发包 (CT Media SDK) 使得在客户机-服务器环境中开发计算机电话学应用和核心信号处理, 如语音识别和音频压缩, 成为可能。

CT Media 提供:

- 允许厂商应用和信号处理硬件与 CT 服务器集成的 C 语言功能库。
- 测试厂商代码和监视其执行的工具。
- 讲述如何使用提供的功能库工具和文档。

本手册预期使第三方开发者能够创建对 CT Media 存储类型的扩展, 把他们的扩展集成到 CT 服务器, 并且和其他 CT Media 兼容部件结合起来。

1.2 ECTF S.100 服务器

CT Media 是计算机电话系统规范 ECTF S.100 (称为 S.100) 的软件和硬件组成部分的实现。

S.100 依从系统的目标是为复杂的计算机电话学硬件和软件的快速开发和有效配置提供一个可选择的平台。支持这个目标的概念是:

- 对资源位置应用的透明度
- 对资源管理和分配的应用的透明度
- 不需要重写应用的情况下系统的可伸缩性 (即资源的增加或删除)
- 保证来自不同厂商资源协同一致性的标准资源接口
- 适应不同客户方程序设计语言和新资源技术的标准可替换应用程序设计接口

S.100 规范通过如下所述实现目标:

- 应用接口层——一套丰富的为计算机电话学优化的参考应用程序接口 (API), 包括呼叫控制、媒体处理和控制、系统服务 API。
- 服务器层——一个抽象的服务器模型, 称为电话学应用对象 (Telephony Application Objects, TAO) 框架。该服务器模型包括一个通信协议, 允许在一个异构系统中机器之间的语言 and 平台独立通信。
- 硬件接口层——一个对流媒体数据实时控制的抽象转换结构模型。将来可望把对 Server 服务器的参考资源接口增加到 S.100 规范中。

实现这三层中的每一个对 Server 厂商都不是强制性的, 注意这一点是有价值的。S.100 规范允许各种级别的依从, 特别是以下一个或多个的组合:

- 愿意支持 S.100 兼容应用的厂商可以仅仅实现应用接口层的形式和功能。但是，如果可能用这样一个平台建立多底盘的系统或增加新的信号处理硬件，使用的接口将是专利的。
- 愿意支持 S.100 兼容信号处理硬件的厂商可以仅仅实现转换结构模性和可能的未来资源接口。不过，应用和多底盘系统接口将是专利的。
- 愿意建造可扩展平台或在多底盘系统中提供专门拥有服务平台的厂商可以仅仅实现服务器层。不过，应用和硬件接口将是专利的。

CT Media 实现完整的 S.100 规范。因此基于 CT Media 的系统有几个显著的特点：

- 他们通常支持一起运行的多个小的应用，每个应用都提供各自非常专门的服务。这和在其他的体系结构中一般情况下具有试图完成每一项工作的单一整体应用形成对照。
- 通常，应用较少关心低级调用控制和硬件资源管理，而更多关心专门的信息处理、存储和传递。
- 如果信号和调用处理硬件开发者愿意实现平台独立，它们将被责成向 S.100 服务器提交高水平的接口。它们还必须为自己的产品提供安装、卸载和管理工具，作为产品包的一部分。

1.3 CT Media 的组成

CT Media 是一套软件和硬件产品，用于相互工作，并与第三方厂商的依从产品交互工作。CT Media 的组成为三个主要部分：

1. 运行组成部分。包括：

- 第三方厂商应用软件
- 客户机/服务器软件
- CT 服务器
- 第三方厂商服务，包括资源软件
- 资源驱动
- 资源固件
- 资源硬件
- 转换结构固件
- 流数据传输硬件

2. 应用和服务开发组成部分。包括：

- 应用开发支持软件库
- 服务开发支持软件库
- 调试工具
- 上述内容的文档

3. 系统管理组成部分。包括：

- 组成部分安装和卸载工具

- 系统安全工具
- 系统维护工具
- 上述内容的文档

CT Media 的各种组成部分能被“混合和匹配”，以满足各种开发和运行环境。特别值得注意的是图 1 所示的典型的自主运行系统。

1.4 CT Media 运行系统组成部分

典型的 CT Media 运行系统由下列组成部分：

- 一个或更多的应用（见 1.4.1 节）
- 一个可替换的应用软件提取层（见 1.4.2 节）
- CT 服务器（见 1.4.3 节）
- 一个资源软件提取层（见 1.4.5 节）
- 资源软件（见 1.4.6 节）
- 信号处理、电话网络接口和流数据传输硬件

一个典型的 CT Media 运行系统如图 1 所示。在下面的章节中，我们将简要地对这些组成部分进行讨论。

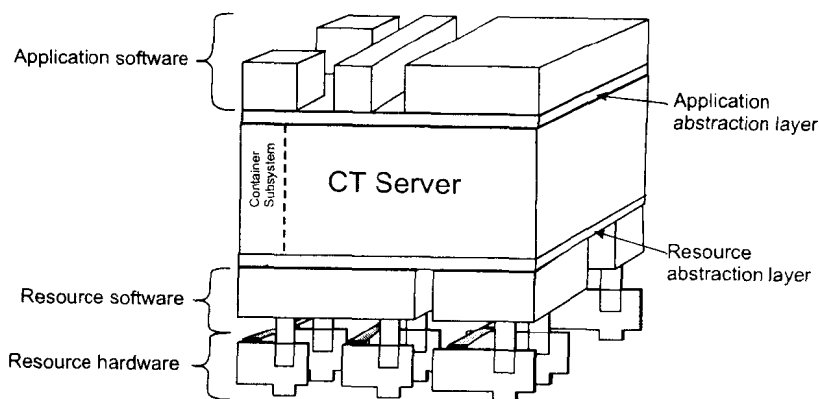


图 1 CT Media 运行系统的高级方框图

1.4.1 应用软件

典型的 CT Media 兼容应用：

- 可以请求、转交和释放调用。
- 可以控制服务器内的媒体处理活动的种类和顺序。
- 可以执行服务器内的管理和配置任务。
- 必须处理运行错误，包括电话网络和硬件故障。
- 形成调用处理或系统管理包的用户界面。

服务器通常将主持一个以上应用，因为 CT Media 兼容应用倾向于是非常专业化的。中

小企业系统可以支持：

- 处理进入呼叫的汽车-服务员申请
- 通知的声音存储与转发申请
- 转换顾客的前提方呼叫的 PBX 申请
- 发送通过商务局域网络送给它的传真申请，接受和转发通过公共电话网络送给它的传真申请

服务器保证这些应用的每一个向函数请求的信号处理资源都被适当地分配，并将管理按照请求在应用之间的调用传输（经常称作“传递”）。

应用通常被写成与一套 C 语言函数和关联事件相一致。

1.4.2 应用软件抽象层——应用接口适配器

应用软件抽象层由一套叫作应用接口适配器（Application Interface Adapters, AIAs）的动态链接库构成。CT Media AIAs 提供的接口由 ECTF S.100 API 规范详细说明。此外，AIAs 同时支持同步和异步编程模式。

应用接口适配器把 API 功能调用转换为被发送到目的地供处理的 API 语言独立信息（请求）。在正常的过程中，请求信息会引起响应，生成响应信息并它回送到应用接口适配器。应用接口适配器把响应信息转换成处理事件。这些事件排队供目标应用收集。图 2 说明应用接口适配器的操作。

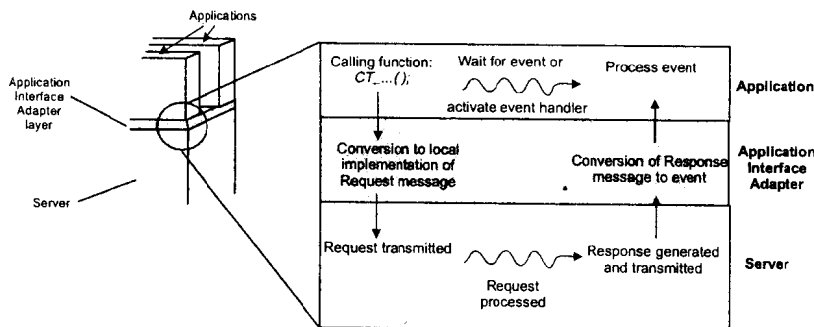


图 2 运转中应用接口适配器

每一个 S.100 定义的 API 集都有一个应用接口适配器。由于应用接口适配器是可更换的，因而 API 的目标语言可以变化，例如从 C 到 C⁺⁺，并且为了支持新出现的技术，还会开发新的 API。

1.4.3 CT 服务器

对于应用和资源开发者来说，CT Media 的核心组成部分是 CT 服务器。支持应用开发者使用的 API 的是服务器，并且，提供资源开发者必须工作的操作环境的还是服务器。作为 S.100 TAO 构架的实现，CT 服务器提供了一个可伸缩和可扩展的平台。在这个平台上，开发和配置遵循 S.100 规范的应用软件（例如，交互式的语音应答、传真回复等等）和资源软件（如：控制电话网络接口的软件、自动语音识别引擎，等等。）

CT 服务器由计划执行专门的任务并且通过来回地通过信息相互作用的软件对象组成。

个别地，或者在合作的群体中，这些被称作服务提供者的对象为 CT 服务器提供请求的系统和服务，以符合 S.100 TAO 构架规范。

服务提供者使用系统信息协议（SMP）互相通信。SMP 是一种每个服务提供者在逻辑服务器内拥有唯一 SMP 地址的无连接协议。

1.4.4 容器子系统

容器子系统是进行数据对象处理的 CT 服务器的一部分。CT Media 安装的容器子系统处理简单的默认类型的数据对象，并被设计成对于其它数据对象类型是厂商可扩展的。本手册特别涉及建立这些厂商扩展。

1.4.5 资源软件抽象层

资源软件抽象层由一套叫作服务接口适配器（SIAs）的动态链环库构成。尽管 SIAs 在功能上与 AIAs 非常相似，但 CT Media SIAs 提供的接口通常不被 ECTF 规范集规定。此外，SIAs 仅支持异步编程模式。

服务接口适配器把服务请求信息转换为厂商提供的控制信号处理和网络接口硬件（资源服务软件）的代码的调用（称为回叫）。反过来，厂商的资源服务软件能调用 SIA 提供的功能来产生服务器要求、响应和事件信息。图 3 描述服务接口适配器的操作。

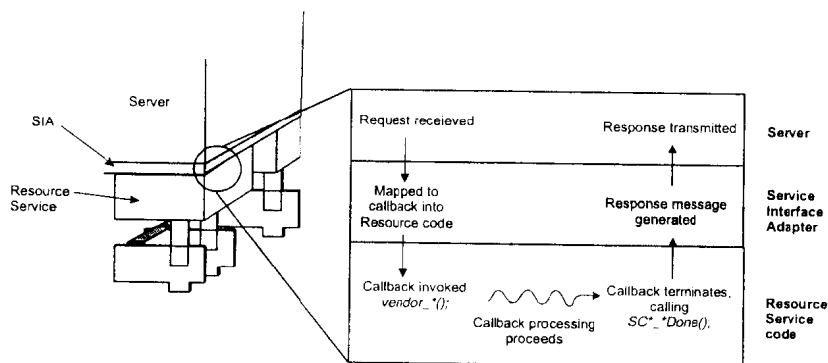


图 3 服务接口适配器的运行

1.4.6 资源软件

CT Media 系统中的信号处理和电话网络接口硬件由叫作资源服务的软件控制。资源服务通过为每一个 S.100 定义的资源功能级提供一个标准的抽象接口来对服务器和客户应用隐藏物理资源的执行。此外，资源服务还提供一个标准化的方法，用以管理（例如开始、停止）资源硬件和控制它的独立服务提供者。

1.5 用 CT Media 数据类型工作

1.5.1 概述

CT 服务器给第三方开发者提供的接口需要使用很多特殊的数据类型。它们在表 1 中列

出。

表 1 CT 服务器数据类型

数据类型	元素
char*	字符串。这里是这些是以空（NULL）结束的 ASCII 字符组
SC_Handle	句柄。用于标识对象，如 SPs(包括 SPCs)、SDPs、Streams 和批量数据对象
SC_ReqDesc	标识符，用来唯一标识显著的命令请求示例
SC_RTCID	标识符，用来唯一标识显著的 RTC 示例
CTaddr	SMP 地址
CTbool	布尔型。布尔变量只能取两个值：CT_boolTrue 或 CT_boolFalse。
CTint	32 位带符号整数
CTkvs_ct	KVSets
CTstatus	功能执行的状态。
CTsymbol	符号。用来标识事件、命令、Set 键、限定符、列举值、参数，等等
Ctuint	32 位无符号整数
CTuint8	8 位无符号整数
struct	固定长度结构。包括： SC_Bulk 批量数据描述符 SC_EventHandlerInfo 事件处理者描述符 SC_RequestHandlerInfo 请求处理者描述符 SC_RTCActionHandlerInfo RTC 行动处理者描述符 SC_RTCCCondition RTC 条件描述符 SC_Status 功能状态 CTfrange 浮点范围 CTirange 整数范围

1.5.2 符号

符号是对非数字列举值分类的方法。它们为了允许名字/数字空间控制被构成。这意味着用户可以：

- 判别标准的和私人所有的符号
- 如果符号是非标准的，表示厂商
- 产生它们自己唯一的符号

S.100 对定的符号具有下列结构：

<Object>_<Vendor>_<Item>

下面是有关符号的例子：

- **Player_ECTF_Play.** 这个符号用来在一个请求信息有效负荷中指定一个资源指令，在本例中是 Player Resource 命令“Play”。
- **Recorder_ECTF_Tvm.** 这个符号用来在一个请求信息有效负荷中标识一个命令参数，在本例中是 TVM，它应当在“记录”操作期间使用。
- **SD_ECTF_Duration.** 这个符号用来在请求和响应信息有效负荷中标识一个资源参数，在本例中是信号探测器的“Duration”参数。

- **Debug_DLGC_Error.** 这个符号用来为调试流函数 **SC_dbgSetupName()** 的参数 **Name** 标识多个列举值中的一个。厂商段的值是把它标识为 Dialogic 拥有的符号的“DLGC”。

开发者为实现开放的容器 DLL 所需要的很多符号在函数描述旁边的本文档中描述。

开发者可以还需要建立厂商特有的符号。

符号操作可以使用的函数在表 2 中列出，并在 9.3 节描述。

表 2 符号处理函数

函数	描述
CTsym_Build()	建立一个符号，为用户提供对象、厂商和项目组成部分的值。
CTsym_Item()	从符号中提取项目组成部分的值。
CTsym_Object()	从符号中提取对象组成部分的值。
CTsym_Vendor()	从符号中提取厂商组成部分的值。

注：开发者应该向 ECTF 秘书处申请以便得到他们拥有的厂商 ID 组成部分。为了测试的目的，Dialogic 公司已预订了 **Private_DLGC_** 符号集供开发者使用。由于这里没有厂商名称空格，开发者有责任保证把使用这些符号的信息等发送到正确的服务提供者。不这样做会导致不可预测的服务行为。

1.5.3 KVSets

键值集（Key-Value Sets，或 KVSets）为 SMP 转递提供包装无序数据的统一手段。

KVSet 是一套键值对，这里键组成部分是符号（见 1.5.2 节），值组成部分是任何支持的数据类型（见表 1），包括另一个 KVSet。

KVSets 具有下列结构：

```
{[Key1:Value1][Key2:Value2]...[Keyn:Valuen]}
```

因为数据（以值的形式）由一个标准的符号（键）标识，所以 KVSets 具有下列特点：

- 它们是无序的
- 它们可能有任意增加和移去的 KV 对
- 值是任意类型
- 在 Kvset 中的每个键必须有唯一的名字

KVSet 的一个例子是 **vendor_StorageSetParameters()** 函数中的 **kvsOpts** 参数。为了设置参数，必须创建一个 KVSet，把代表参数名字和参数值的 KV 对放进来。KVSet 可以如下面所示：

```
{
    Container_ECTF_FileSize: 20
    Container_ECTF_MediaType: FileStrategy
}
```

KVSets 作为 CT 服务器的对象（封装的数据类型的例子，或 EDT 级）被实现，并且必须由它们的公共成员函数操作。KVSet 对象有 10 个成员函数，如表 3 所列。

表 3 KVSet 操作函数

函数	描述
CTkvs_Clear()	从象 KVSet 那样删除全部 KV 对。
CTkvs_Copy()	复制一个 KVSet。
CTkvs_Create()	建立一个空的 KVSet。
CTkvs_Destroy()	废弃一个 KVSet。
CTkvs_Get/Typ()	读由一个给定的键标识的值。值是一种特别的类型。
CTkvs_Get/Typ/Array()	读由一个给定的键标识的值的数组。数组是一种特别的类型。
CTkvs_GetNextKey()	给出一个键, 返回 KVSet 中的下一个键。
CTkvs_GetType()	得到由一个给定的键标识的值的类型。
CTkvs_Put/Typ()	为 KVSet 增加一个 KVP 对。值是一种特别的类型。
CTkvs_Put/Typ/Array()	为 KVSet 增加一个 KVP 对。数组中的值是一种特别的类型。
CTkvs_Remove()	从 KVSet 中删除一个或更多的 KV 对。

注: 在资源和应用接口级的 KVSet 函数是相同的。由于这个原因, KVSet 函数(当用在资源接口级时)是和资源函数的大多数不同的, 并且返回一个 CTstatus 值和一个指向错误的指针来显示它们的状态。这与多数资源接口功能返回在 SC_Status 结构中的状态和错误值形成对照。

由于 KVSet 作为 CT Media 的对象来实现, 它们通过句柄来操作。KVSet 句柄通常被归类 (typedefed) 为空的 (void*)。因而, 编译器将接受任何数据类型作为有效的 KVSet 句柄。用户必须保证预期是 KVSet 的函数参数确实是 KVSet, 否则, 将会导致不可预测的行为。

1.5.4 句柄

句柄是唯一标识 CT 服务器中非数据类型对象的方便方法。资源的编写者最初对句柄感兴趣是因为:

- 服务提供者 (SP)
- 转换域端口 (SDP)
- 连接到 Media 数据对象的流, 如时间变化媒体对象 (TVM)
- 读 Media 数据对象时管理缓存器的批量数据对象

在进行以下工作时句柄会典型地返回:

- 建立一个服务提供者
- 打开一个 SDP
- 为 Media 数据对象打开一个流
- 读一个 Media 数据对象

SPs、SDPs、批量数据对象和流作为 CT 服务器中的对象被实现, 并且必须由它们的公共成员函数操作。这些对象的超类有 3 个公共成员函数, 如表 4 所列。注意这些功能操作由句柄指定的对象, 而不是句柄本身。

表 4 面向函数的句柄

函数	描述
SChdl_AttachContextData ()	把用户定义的数据附着到由一个给定的句柄标识的对象

(续表)

函数	描述
SChdl_ContextData ()	读附着到由一个给定的句柄标识的对象的上下文数据
SChdl_Destroy ()	废弃由一个给定的句柄标识的对象

1.5.5 状态结构

状态结构用来返回包括 CT 服务器资源接口的操作的当前或完成状态，并且由除 KVSet 及一些批量数据和调试流函数之外所有资源级函数返回。它们：

- 当函数调用返回（即，从厂商代码到 CT 服务器资源接口的调用）时被返回，并显示成功或其它调用。KVSet、批量数据和调试流函数是这条规则的例外。这被称为同步状态。如果函数调用是打算开始命令处理，那么，同步状态显示命令处理是否成功地开始。
- 作为在从 CT 服务器接口到厂商代码的任意完成函数调用中的参数传递，并显示成功或其它来自厂商的命令请求处理。这被称为异步状态。
- 为在任意从厂商代码到 CT 服务器资源接口的 **SC* *Done()**调用中的参数传递，并显示成功或其它来自服务器的命令请求处理（即回收处理）。

状态结构如下所示：

```
typedef struct
{
    Ctstatus    status;
    Cterror     error;
    CTint       subError;
} Status;
```

状态结构中的成分 **status** 取四个可能的值之一：

- **CT_statusOK** 按预期完成操作或命令
- **CT_statusWarning** 意外原因完成操作或命令
- **CT_statusFail** 因函数调用中的语法错，操作或命令失败
- **CT_StatusFatal** 因意外的硬件或系统故障，操作或命令失败。厂商处理应立即停止。

状态结构中的成分 **error** 解释返回的状态的原因，并且通常仅在 **status** 的值为 **CT_statusFail** 或 **CT_statusFatal** 时提供。这个成分为特别的函数和命令取特别的值。

状态结构中的成分 **subError** 为特别的函数和命令取特别的值。

CT Media 为使用状态结构提供了方便的函数，见表 5 所列。

表 5 建立状态结构

函数	描述
SC_CreateStatus()	为 status 、 error 和 subError 给定值建立类型 SC_Status 的结构

1.5.6 存储器管理

资源开发者必须遵守的一条存储器管理规则：

废弃自己建立的任何数据元素和对象是厂商代码责任。这包括当从厂商代码到 CT 服务器资源接口的函数调用时作为参数传递的元素和对象。

这条规则的推论是：

不能要求在回收或从 CT 服务器资源接口到厂商代码的完成函数调用时作为参数传递的数据元素和对象在超出回收或完成函数执行的持续期间继续有效。一旦回收或完成函数返回，用户应要求 CT 服务器资源接口废弃这些对象和数据元素。如果用户在回收或完成函数持续期间之外需要元素或对象，它必须作一个拷贝。然后还要负责废弃该拷贝。