

基于 Symbian OS 的 MP3 播放器的设计与实现



学 院 计算机科学与技术
专 业 计算机科学与技术（双学位）
年 级 2005 级
姓 名 周伟
指导教师 张坤龙

2009 年 06 月 18 日

摘 要

随着智能手机的快速发展和手机音乐播放功能普及,对手机播放软件的需求也在快速增长,手机播放软件的市场前景广泛。

论文设计和实现了一个名为 MP3Player 的基于 Symbian OS 的 MP3 播放器软件。Mp3Player 的主要功能包括:(1)能够从手机存储卡中读取 MP3 文件,显示其歌曲名和歌唱者等 MP3 元数据信息;(2)能够开始、暂停和停止 MP3 文件的播放;(3)能够以可视化的方式调节 MP3 播放时的音量。

关键词: Symbian; 多媒体框架; MP3 播放器

ABSTRACT

With the rapid development of smart phones and the popularity of music functions on mobile phones, the demands of music players on mobile phones are also fast-growing, which have broad market prospects.

Paper is designed and implemented a Mp3Player for Symbian OS-based MP3 player software. Mp3Player main features include: 1) memory card from the phone to read MP3 files, showing their name and singing songs such as MP3 meta-data information; 2) to begin, pause, and the suspension of play MP3 files, MP3 can be a key to switch the Play / Pause state; 3) be able to adjust the way visual MP3 player at the time of volume, can be a key to increase or decrease the sound volume.

Key words: Symbian; multimedia framework; MP3 player

目 录

第一章	绪论	1
1.1	智能手机概况	1
1.2	手机多媒体	2
1.3	MP3 播放器现状	2
1.4	论文组织结构	5
第二章	Symbian OS 应用程序开发	6
2.1	Symbian 操作系统	6
2.2	S60 应用程序开发	7
2.3	S60 多媒体框架	9
2.4	开发环境的建立	10
第三章	MP3Player 的设计	12
3.1	需求分析	12
3.2	客户端体系结构	13
3.3	功能模块设计	16
第四章	MP3Player 的实现	19
4.1	动态菜单栏的实现	19
4.2	MP3 元数据的显示	20
4.3	音量调节	21
4.4	二阶段构造	22
4.5	中文用户界面	23

4.6 安装程序生成	24
第五章 结论	26
5.1 总结	26
5.2 展望	26
外文资料	
中文翻译	
致谢	

第一章 绪论

1.1 智能手机概况

随着科学技术的不断发展,手持移动设备如:PDA(个人数字助理)、掌上电脑、手持电脑、WAP 手机、GPRS 接收器、智能手机等发展迅速。随着这些手持移动设备功能的不断完善,已经出现了多合一的趋势,混合了 PDA 和移动电话各项功能的智能手机正是这种趋势的杰出代表。智能手机不仅仅是一种消费终端产品,更是多种应用的平台,能为消费者提供语音之外的很多额外功能,比如录播视频、发送视频短信、流媒体下载与播放、视频通话、3D 游戏、3D 立体声播放等等^[1]。

2009 年,中国的三大运营商(中国移动、中国电信和中国联通)已经推出了 3G 业务。智能手机作为一种多用途的移动设备,提供了足够的屏幕尺寸和带宽,既方便随身携带,又为软件运行和内容服务提供了广阔的舞台,让消费者更好地体验丰富的 3G 应用,增值业务开发潜力较大,成为了手机市场重点成长的产品。

2008 年全球智能手机的销售量达到 1.65 亿部,占全年手机总销量的 14%左右。市场研究公司 ABIResearch 预测虽然今年受经济危机影响,全球手机销量有所下降,但智能手机销量则逆势扩张,将超过 2 亿部。相信在不久的将来,智能手机必然占据手机市场的主导地位。

2008 年 Symbian 市场占有率虽然降低了 15%,但是市场份额依然占据 50%左右,保持着智能手机操作系统上的老大宝座。凭借多年的发展经验、众多顶级厂商的支持以及广泛的用户认识度,Symbian 协会掌门李·威廉斯认为这种下滑的局面将很快得到扭转。今年 Symbian 协会已经在与华为和中兴谈判。李·威廉斯预计中国将有 1 或 2 家移动运营商、3 至 5 家 ODM 厂商和至少 1 家芯片厂商加盟 Symbian 协会。中国厂商的加盟将有助于 Symbian 手机在中国的进一步发展。

目前市场上的手机操作系统主要有四类:Symbian、Palm、Linux 和 Windows Mobile。其中 Symbian 系统的手机占据了中国智能手机市场 60%以上的份额。Symbian 与其他移动终端操作系统相比,主要具有以下两个优势^[2]:

(1)Symbian OS 是专门为移动设备设计的操作系统,其采用了先进的设计思路,具有优良的连接性和强大的扩展性,并能有效的减少电能的消耗,非常适合体积小、功能强、资源有限的移动通讯终端设备使用。Symbian OS 是用 C++语言编写,它是一个完全面向对象的操作系统。这使得它具有极强的灵活性、有效性,并拥有代码重用的能力。该系统采用 C++类型的应用程序接口,可以十分有效的管理同时进行的多个任务和设备内存,有效的、最大限度的减少占用中央处理器和消耗内存的操作。

(2)Symbian 由众多顶级传统手机厂商联合成立,公司的股东诺基亚和索尼爱立信等几家手机巨头的支持也是其他竞争对手不可企及的巨大优势。

1.2 手机多媒体

诺基亚曾经进行了一项关于移动设备融合的全球调研。在该项包括中国、法国、印度、日本、美国等 11 个国家在内的全球性调研中,诺基亚有三个主要的发现:目前全球几乎每两个人中就有一个将他们的手机作为主要的相机使用;超过三分之二的人预言音乐手机将会取代他们的 MP3 播放器;半数以下的人们希望将移动设备与他们的家庭电器相连。其中中国消费者对手机的依赖高于世界平均水平;中国消费者对手机的摄影和音乐功能非常狂热;在中国,以下载的方式来购买音乐正成为主流;中国消费者喜爱流行乐的比例位居调查国家第二;中国消费者希望未来的手机增加音乐视频的功能。

手机多媒体已经成为一种主流趋势。国外的手机多媒体应用已得到了很好的发展,在音频和视频方面软件种类多样、性能优越且功能齐全,甚至已经有了专门的手机电视网络和音频播放网络。

国内的手机多媒体发展也相当迅速,这些多媒体业务主要以彩信、彩铃、手机游戏、手机多媒体播放为代表。智能手机的多媒体播放功能跟为强大,使得手机自身带的软件不能满足用户的需求。第三方软件作为智能手机重要组成部分,不仅可以美化手机界面,更可以提高手机性能,增加手机没有的功能。随着智能手机日益普及,随之而生的软件开发已发展成为一项产业。主要软件公司如东软集团、盛大数位红,掌讯、迅彩科技等^[2],都具有了一定规模。但是由于国内手机软件发展起步较晚,没有完整统一的体系,所以软件功能不齐全,性能较差,还有较长的路要走。

1.3 MP3 播放器现状

MP3 格式的音乐文件是现在最普及的音乐格式,下面简单介绍一下 MP3 编码。MP3 的全称应为 MPEG1 Layer-3 音频文件。MPEG(Moving picture Experts Group)在汉语中译为活动图像专家组,特指活动影音压缩标准。MPEG 音频文件是 MPEG1 标准中的声音部分,也叫 MPEG 音频层,它根据压缩质量和编码复杂程度划分为三层,即 Layer1、Layer2、Layer3,且分别对应 MP1、MP2、MP3 这三种声音文件,并根据不同的用途,使用不同层次的编码。MPEG 音频编码的层次越高,编码器越复杂,压缩率也越高。MP1 和 MP2 的压缩率分别为 4:1 和 6:1-8:1,而 MP3 的压缩率则高达 10:1-12:1,也就是说,一分钟 CD 音质的音乐,未经压缩需要 10MB 的存储空间,而经过 MP3 压缩编码后只有 1MB 左右。第三层音频压缩模式比第一层和第二层编码要复杂得多,但音质最高,可与 CD 音质相比。MP3 的数据是以帧流的形式存储或传输的,每帧由帧头和帧数据组成。

帧头(header)包括头信息(4 个字节), CRC 校验数据(可选, 2 个字节), 帧边信息(也可称为附加信息,单声道为 17 个字节, 双声道为 32 个字节)。其中头信息中包含了 MP3 帧流的特征信息, 如位串指数、采样频率、填充位等。

现在手机 MP3 播放器发展迅速, 众多播放软件层出不穷。下面主要介绍了一下天天动听、Coreplayer 和 PowerMP3 这三款比较流行的软件。

天天动听是一款集播放、均衡器、歌词、睡眠模式、闹钟等众多功能于一身、操作简捷、功能强大的手机音乐播放软件, 其界面如图 1-1 所示。由于其支持最多手机机型和音频格式, 支持丰富的皮肤下载等功能, 同时其与手机搭配和谐、操作简易、管理人性的特点, 深受用户喜爱。天天动听一直以用户的需要为发展方向, 是目前国内最受欢迎的手机音乐播放软件之一。



图 1-1 天天动听界面

天天动听的优点有: (1) 可以方便的对播放列表进行收藏、删除、修改等管理; (2) 天天动听界面直接标记了快捷键, 方便用户使用, 并且安排了众多特殊功能的快捷键; (3) 使用过程中先要扫描自己的音乐库, 将手机中的歌曲添加到天天动听的播放列表中; (4) 软件提供了睡眠模式, 在你睡觉的时候不会因为忘记关闭播放器而使用手机耗尽电池; (5) 天天动听还提供了诸如 一键锁键盘, 闹钟等功能; (6) 支持更换皮肤; (7) 支持自定义均衡器, 同时有多种预置均衡器风格, 譬如流行, 舞曲, 平坦, 超重低音; (8) 提供歌曲 ID3 修改器, 解决了乱码的问题。

Coreplayer 是一款集中视频与音频的全能播放器, 界面如图 1-2 所示。这款播放器在音频输出方面选项非常丰富, 通过自定义、低音、古典、舞曲以及摇滚等模式的设定用户可以轻松的享受每一曲音乐。CorePlayer 支持 MP3、MP2、AAC、MKA、WMA、Midi、WAV、OGG、Speex、WAVPACK、TTA、FLAC、MPC、AMR、ADPCM、ALaw、MuLaw、G.729 等格式。



图 1-2 CorePlayer 界面

在音频的播放方面，CorePlayer 存在有缺点。很多用户觉得这款播放器虽然功能强大，但是音乐处理方面还欠缺专业性，操作不是特别方便，并且里面的均衡器很多都会出现严重破音。

PowerMP3 是目前 s60 平台中音质最好的 MP3 播放器之一，界面如图 1-3 所示。它不但界面漂亮，拥有丰富的 EQ 调节功能，还可以播放 MP3 及 OGG 等格式的音乐文件。PowerMP3 的主要优点包括：（1）制造播放清单来管理音乐文件；（2）精致的皮肤；（3）10 个频道的图像式平衡器，附有许多类型的音乐预设调整；（4）图像式的光谱分析器；（5）使用轻摇的方式来增减音量；（6）能在歌曲播放时调较声音的平衡；（7）能于 Flip Open 和 Flip Close 模式内播放音乐；（8）背景回播；（9）接完电话或拨打电话完毕后，PowerMP3 能自动恢复播放；（10）能使用时间切割器及快速回播设定播放位置。PowerMP3 的缺点是它只能搜索 Sound 文件夹中的 MP3 文件，有待改进。

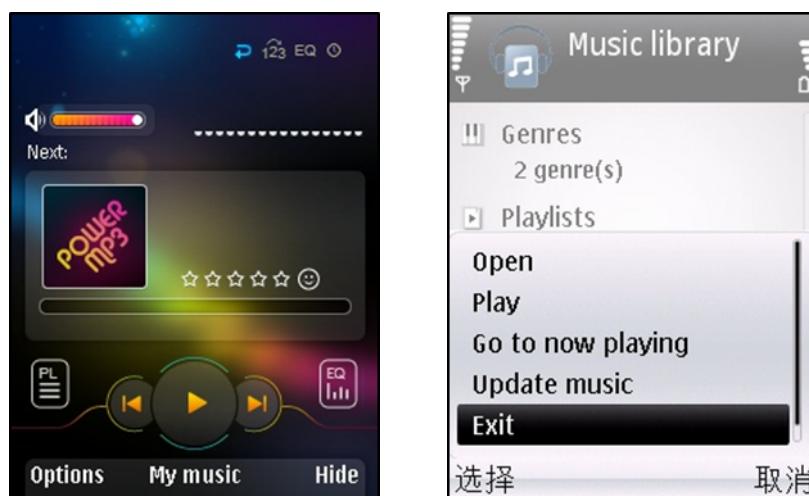


图 1-3 PowerMP3 界面

。

1.4 论文组织结构

论文设计和实现了一款基于 Symbian OS 的 MP3 播放软件,该软件被命名为 MP3Player。论文的组织结构如下。

第一章分析了基于 Symbian 的 MP3 播放器的发展状况。

第二章讲述了 Symbian 应用程序开发过程,介绍了 Symbian 音频多媒体技术和开发环境的搭建方法。

第三章给出了 MP3Player 的设计。首先分析对 MP3Player 的需求,然后给出了 MP3Player 所遵循的设计模式、MP3Player 的客户端体系结构设计和 MP3Player 的功能模块设计。

第四章说明了 MP3Player 实现中所用到的一些关键技术,包括动态菜单栏的实现、元数据的显示和中文界面等。最后介绍了安装程序生成。

第五章对全文进行总结,并展望了 MP3Player 未来的发展前景。

第二章 Symbian OS 应用程序开发

2.1 Symbian 操作系统

Symbian 是一家由各主要无线通信技术公司持股的软件许可证授权公司，是由 Psion(宝意昂)公司和它的 EPOC 操作系统发展而来的^[4]。

截止到 2006 年 Symbian 的持股公司包括爱立信、松下电器、诺基亚、西门子、三星和索尼爱立信六家公司，持股比例如图 2-1^[5]：

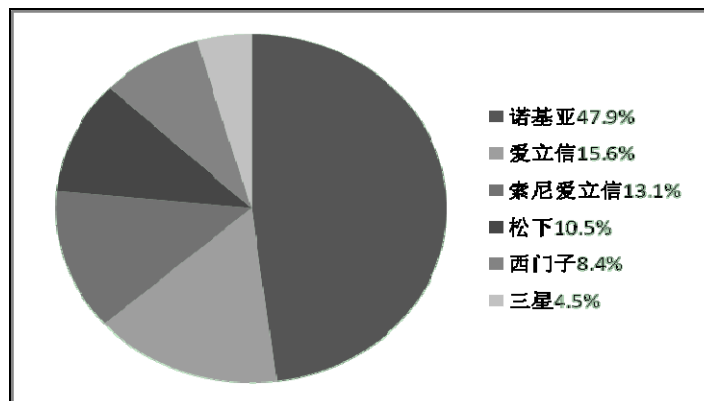


图 2-1 2004 年 7 月 Symbian 公司的持股公司持股比例图

北京时间 2008 年 12 月 2 日，诺基亚宣布该公司已经完成了对手机操作系统研发公司 Symbian 的收购。创立一个开放的移动软件平台。2009 年高通、惠普、MySpace 及其它 12 家公司加盟诺基亚主导的 Symbian 协会(Symbian Foundation)，至今该协会已获得 78 家公司的支持。现在基于 Symbian 平台已经有超过两亿部 235 种型号的手机由多家厂商生产发货，而成千上万的为基于 Symbian OS 的手机服务的第三方应用程序早已推向市场。

Symbian OS 是一种高级的开放的标准的多任务操作系统^[6]。但它没有提供特定的用户界面，只提供操作系统的核心功能。目前基于 Symbian OS 的用户界面平台主要有：S60 平台、80 系列平台、90 系列平台及 UIQ，如图 2-2^[6]展示了 Symbian 公司与各个用户界面平台之间的关系。

S60 是在 Symbian 下进行智能手机应用程序开发的主要平台，拥有广大的用户群，许多不同生产商生产的多种终端上都能使用。S60 中提供了一套功能强大的组件和许多不同的 AIP。成套的“标准化”应用广泛使用了所提供的这些 API，这些 API 也被设计成可为第三方应用开发伙伴们重用。由独立的软件商、S60 获取授予商、技术中心、承包商和移动网络运营商形成了一个完整的体系，进一步扩大了市场^[7]。

目前 Series 60 有三个主要版本，即第一版 Developer Platform V1.0、2nd Edition 以及 3rd Edition。第三版本为目前最新的版本，也是应用最为广泛的版本。

第三版集成了 S60 操作平台第二代版本的所有功能，还有一些附加功能。最重要的附加功能是操作平台安全性的提高，以确保引入和实施实时内核（eka2）。此外还提供了增强的浏览器和灵活的 XML 等^[8]。在多媒体功能方面，Series60 第三版将提供全新的音乐播放器。全新的音乐播放器将提供无缝和易操作的带有个性化主题和背景的音乐下载。丰富的媒体内容提供增强的功用，并同时实现高安全内容的保护。此外 CUSB Mass Storage 协议更方面地实现从 PC 到移动电话的音乐传送。

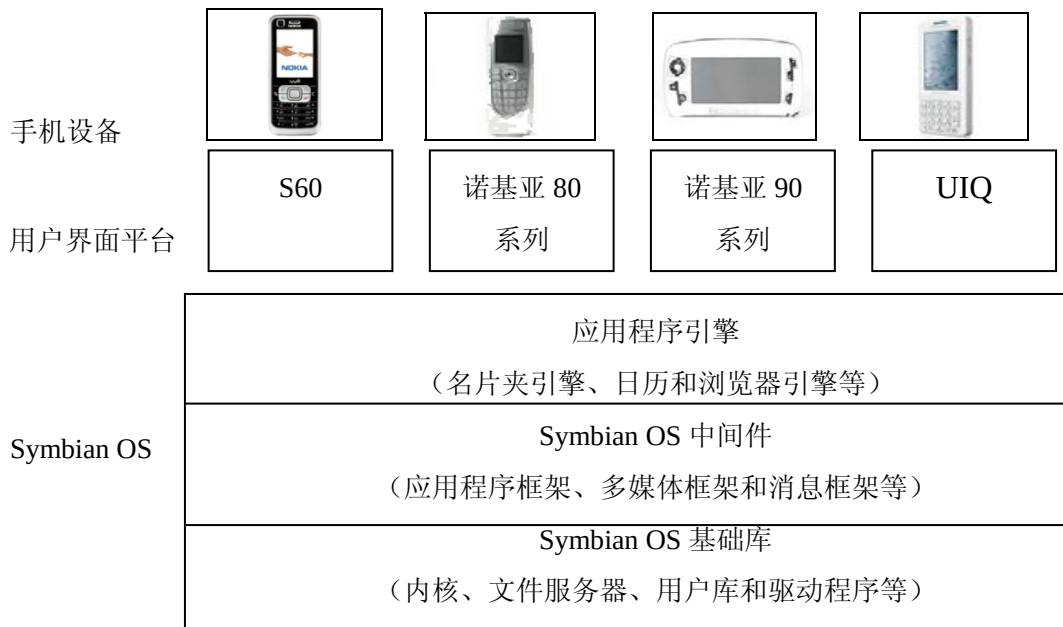


图 2-2 Symbian 手机界面图

2.2 S60 应用程序开发

应用程序框架（Application Framework）定义图形界面应用程序中使用的程序框架，包括应用程序的基本组成类、响应用户事件的流程、处理程序的方法等。这里的应用程序是指图形界面的应用程序，不包括运行在后台的服务器程序。

通常一个应用程序至少包括四个从应用程序继承的类：Application 类、Document 类、AppUi 类和视图类。应用程序的架构如图 2-3 所示。图的上半部分是 S60 平台提供的应用程序框架中的类，下半部分是用户在应用程序中自己定义的类。

应用类是应用程序的启动对象，它定义应用程序的属性，创建文档类。应用类的基类是 CAknApplication。文档类用来存储应用程序的永久状态，一个应用程序必须有一个文档类的实例，尽管它除了加载 AppUi 类似乎没有什么别的用途。文档类的基类是 CAknDocument。应用界面类负责处理与应用有关的事情，比如菜单命令。通常没有屏幕显示，它将图形绘制和基于屏幕的交互操作委派给自己所拥有的视图类，自己则负责处理来自各个视图和引擎的消息。应用界面类

是整个应用程序的核心。应用界面类的基类是 `CAknAppUi` 或 `CAknViewAppUi`。视图类负责在屏幕上显示那些用户可以与之交互的数据。它将用户发出的命令传递给 `AppUi` 类进行处理。需要注意的是，在 S60 的 SDK 中，对视图的称呼，经常使用 `container`（容器），而不是 `View`（视图）。您可以认为它们是等价的。实际上，当视图仅有一个控件时，我习惯称它为控件；当视图有多个控件时，我习惯称它为容器。视图类继承自 `CCoeControl` 类或者 `CAknDialog` 类。当应用程序被设计成应用/视图体系结构时，它继承自 `CAknView` 类。

Symbian 操作系统中典型的三种应用程序框架分别是：传统的应用程序框架、基于对话框的应用程序框架和基于视图的应用程序框架^{[9][10]}。选择哪种应用程序结构取决于应用程序的复杂程度、视图导航需求、通信需求以及屏幕布局要求，但无论采用哪种结构，每个应用的顶层应用程序用户界面类总继承自某个单一应用程序的用户界面基类。这个基类并不强制开发伙伴选择某种用户界面结构。

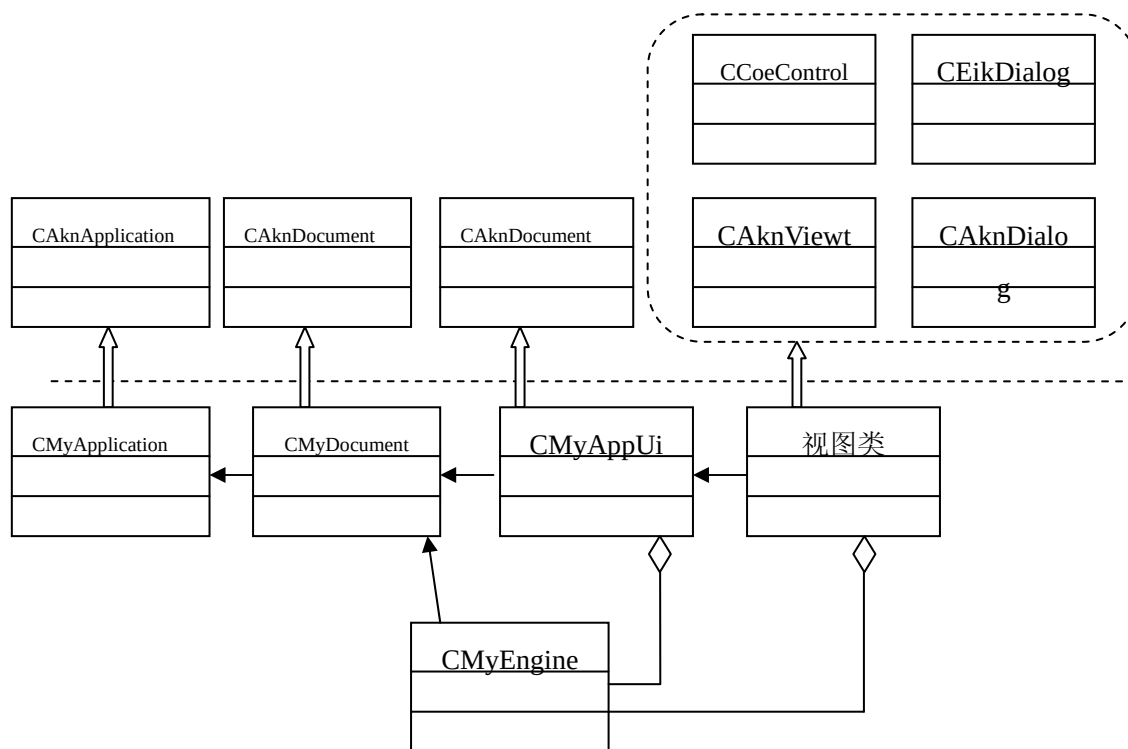


图 2-3 应用程序架构图

2.2.1 传统的应用程序框架

传统上，Symbian OS 应用程序是用继承自 `CCoeCnotrol` 的自定义视图控件编写的，这些自定义视图控件被放置于应用程序控件堆栈中，作为应用程序的各种视图。可以根据应用程序提供相应功能的需要或创建或清除，或显示或隐藏这些控件。

视图切换是通过创建、清除或更改主视图控件的可视性来实现的。应用程序用户界面在控件堆栈中放置或移去各个控件，从而将各个按键事件定向到当前控件，也可以通过手工操作将各个按键事件重定向到各个控件。

传统的应用程序框架是一种构建应用程序用户界面的最灵活的方法。就视图结构来说其主要缺点是它没有使用系统提供的视图管理系统。同时，这也是它的主要优点，因为视图管理系统存在很多限制。

传统的应用程序框架非常适合于多个视图进行频繁的切换，通常也会应用于客户端/服务器系统。

2.2.2 基于对话框的应用程序框架

这种方法非常适合 UI 控件组成传统的 OS 控件的简单排列应用程序。可以在资源文件中定义控件，让对话框自动处理布局和绘画，这比实现自定义绘画行为更为容易。

同时这种方法也有一个缺点就是如果不仔细编写代码，嵌套的对话框会占用相当多的堆栈空间，消耗掉过多的内存资源，使得程序性能下降。

2.2.3 基于视图的应用程序框架

视图结构是一种允许应用程序注册各种视图的系统，运行中的应用程序每次都有一个视图是处于活动状态，但并不规定是什么样的视图，然而它确实支持视图作为在屏幕上的一个显示页。

应用程序用户界面创建并注册其提供的每一个视图，并允许外部应用使用一个视图 UID 来切换到这些视图。这种框架最适用于多视图的应用程序。但它也具有局限性，如：视图切换方案没有任何内置的方法来保存视图切换的上下文。

2.3 S60 多媒体框架

多媒体子系统定义了多媒体框架（MMF，Multi Media Framework），包括音频、视频、摄像头、图像转换等功能，如图 2-4 所示。

Symbian 多媒体框架类似于一个多媒体处理插件程序的仓库，它为硬件设备提供一般的接口。其中的关键部分是控制器框架，由其管理插件程序的选择，装载以及插件间的交互。

MMF 采用 Client/Server 架构模型^[11]，基本结构包括一个客户端接口层、一个控制器框架、控制器插件和底层子系统^[12]。其中客户端接口层为应用程序访问插件提供途径，具体的途径由插件实用工具类对象提供。控制器框架选择和启动相应的插件程序，即传递客户端消息到相应的插件程序。相应的控制器插件负责处理从客户端来的数据。音频设备通过硬件设备接口为音频硬件提供通用接口。音频设备策略解决客户端同时请求音频硬件使用的优先权问题。底层子系统提供一些底层的硬件接口。

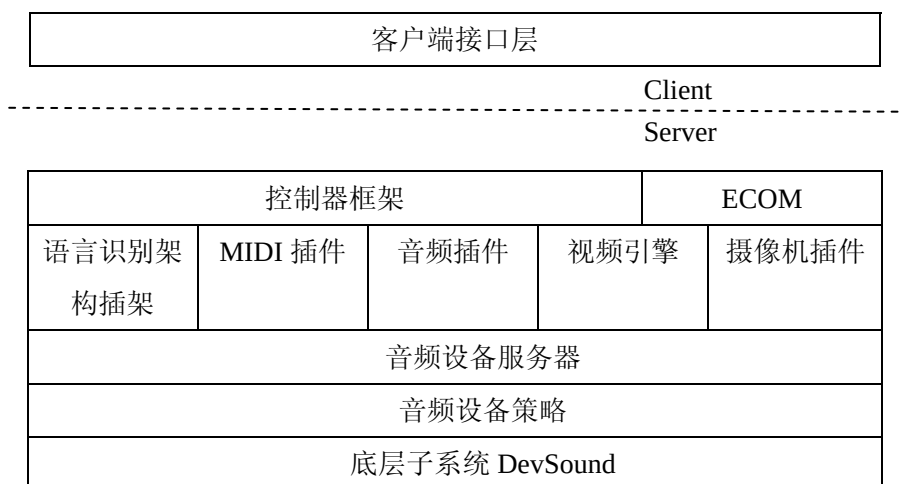


图 2-4 MMF 架构模型

2.4 开发环境的建立

Symbian C++是 Symbian OS 最基本的编程语言^[13]。“最基本”这三个字往往具有很大的魅力——它意味着更纯粹的功能和更高效的表现。Symbian C++是改进后的 C++。之所以改进是为了使 C++更适于在手机这个受限设备中施展舞技。手机设备的突出特点是资源宝贵，为此，我们不得不为 C++这位优秀的舞者套上脚镣。本文选择 C++语言作为开发语言。

对于 Symbian OS C++开发人员，目前有 3 种可用的 IDE：1)Carbide.C++，基于被称为 Eclipse 的开放源码；2)Microsoft Visual Studio，大多数 Windows 程序员选择的 IDE；3)Code Warrior，以前的 Palm 程序员和游戏站开发选择的 IDE。

Carbide.C++是运行在 windows 操作系统之上的，基于 Eclipse3.3.1.1 和 EclipseCDT4.0.1 的 Symbian C++软件开发工具^[14]。Carbide.C++提供的 UI Designer 可以大幅度提高 Symbian 程序的 UI 设计速度，是非常重要的开发工具。Carbide 是 Nokia 推出的新一代移动开发工具，实质上代表着 Nokia 为了将其开发工具融入一个共通框架而采取的深思熟虑的举措。Carbide 使移动开发在功能和效率上迈向一个新高度，它的出现意味着开发者拥有了一个可以在多种平台下用多种语言进行开发的开发软件家族。

所有安装都必须以管理员的权限进行，否则一些环境变量将得不到正确的设置。SDK、IDE 和工程文件需要放在同一个逻辑分区上，否则在以后的开发过程中，可能会引起预想不到的问题。SDK 的安装途径里不能有空格。安装顺序为 ActivePerl、Java Runtime、IDE 和 SDK。

在搭建开发环境之前，首先我们下载需要的 SDK 及辅助工具：

- 1) 下载 ActivePerl，Nokia 推荐的是 5.6.1.X 版本。
- 2) 如果计算机中没有 JRE，可以到 SUN 网站下载安装。JRE 选择最新版本

即可。

3) 从 Nokia 的论坛上免费下载(需要先注册成会员)Carbide.C++, 在选择版本时一般选择 Express 版本。

4) 下载 S60 的 SDK, 需要下载 3rd Edition Maintenance Release 版本的 S60 SDK。

安装过程:

第一步: 安装 ActivePerl 和 java 运行环境(J2RE);

第二步: 安装 Carbide.C++。

第三步: 安装 S60 3rd SDK, 安装完成后需要注册。

把上面的三步都执行完成, 就完成了第三版的 SymbianOS 开发环境的建立了。

第三章 MP3Player 的设计

3.1 需求分析

用例需求是系统提供的功能块，换句话说来说用例演示了人们如何使用系统。通过用例观察系统，能够将系统实现与系统目标分开，有助于了解工作重点。通过用例，用户可以看到系统提供的功能，先确定系统范围再深入开展项目工作。MP3Player 具体用例分析如图 3-1 所示。

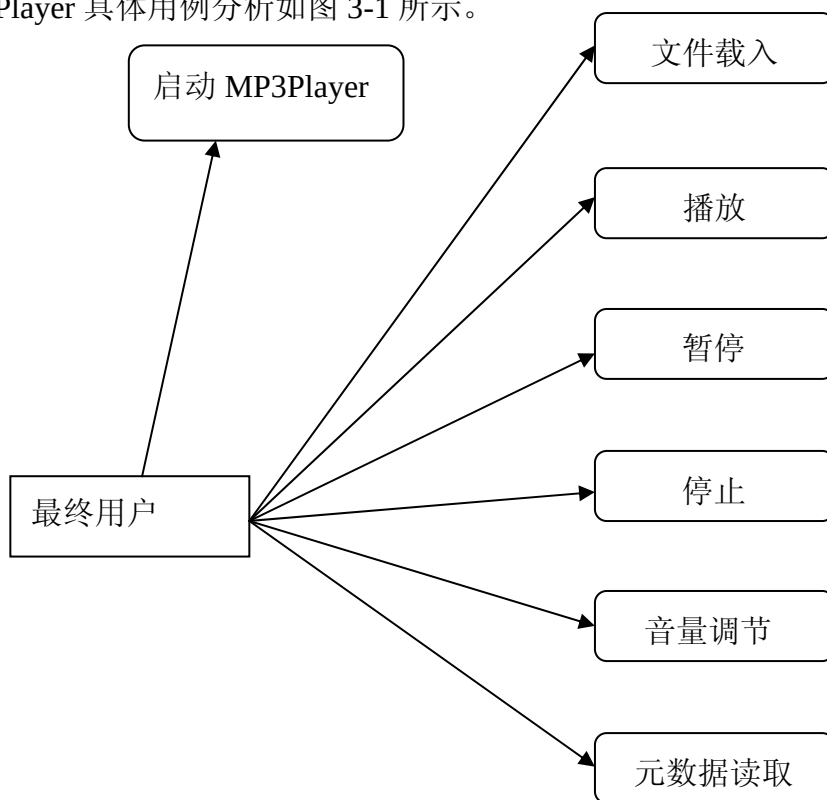


图 3-1 MP3Player 用例图

功能需求定义了开发人员必须实现的软件功能，使用户利用系统能够完成他们的任务，从而满足了业务需求。MP3Player 需要实现的主要功能描述如下：

- （1）载入：打开手机存储的 MP3 文件；
- （2）播放：载入成功后，软件即可以打开所载入的文件进行播放；
- （3）暂停：在文件播放过程中停止播放文件，但是任然定位在停止播放的位置，可以恢复原位置的播放；
- （4）停止：停止播放文件并退出该文件信息；
- （5）音量调节：在播放过程中调节播放的音量大小；
- （6）元数据：显示播放文件的信息，如：歌曲名称，演唱者等。

根据需求分析，MP3Player 采用的软硬件设施如下：

运行系统：SymbianS60 第三版本的智能手机；

运行硬件环境：普通 PC 机；

开发系统：WINDOWS XP

开发语言：Carbide C++；

模拟器：Symbian S6o 第三版 SDK。

3.2 客户端体系结构

该多媒体客户端平台基于 Series 60 应用程序架构，并且采用了模块化的设计模式，为未来的扩展和技术维护提供便利。客户端平台划分为媒体播放模块、UI 模块和数据存储模块三个部分。其可以支撑各种基于 Symbian 智能手机多媒体应用软件的开发。客户端平台架构如图 3-3 所示。

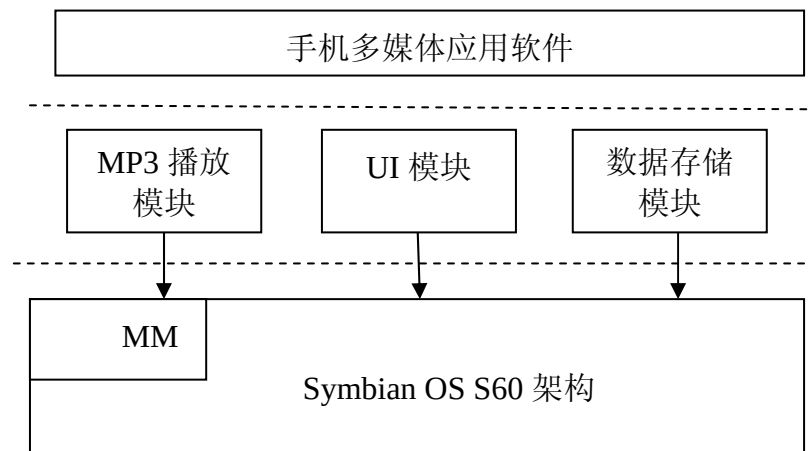


图 3-3 客户端平台架构图

3.2.1 MP3 播放模块

本文是基于 Symbian MMF 音频播放接口的播放器，接口 API 主要由 MMdaAudioPlayerCallback 和 CMdaAudioPlayerUtility 两个类组成。

其中 MMdaAudioPlayerCallback 是一个接口类，针对 CMdaAudioPlayerUtility 的回调类，它报告错误并向应用提示文件打开操作或播放已经完成等状态。CMdaAudioPlayerUtility 类提供了对播放、暂停、快进、后退以及音量设置等操作的 API，还支持元数据读取。

由于 CMdaAudioPlayerUtility 中的所有操作都是异步的，所以有必要让一个客户端类来侦听音频播放操作。这个类必须继承自 MMdaAudioPlayerCallback 接口类，它提供了两个方法，即 MapcInitComplete()和 MapcPlayComplete()。当载入一个 MP3 文件的工作结束之后，不管其成功与否，MapcInitComplete()方法将定义所需的客户端行为。而当播放某段音频采样操作完成后，不管其成功与否，MapcPlayComplete()也将定义所需的客户端行为。

此种开发方式的优点是实现简单，缺点是在两段音频切换之间有一定的间

隔。类图如图 3-4 所示。

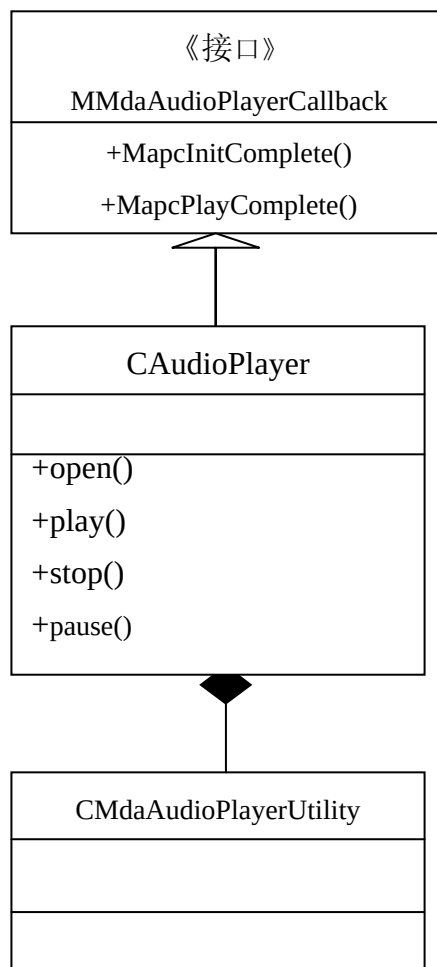


图 3-4 音频剪辑播放器类图

3.2.2 UI 模块

由于在多媒体平台应用中，会用到很多视图切换编程，因此选用基于视图程序框架。视图切换只需要从 **View** 服务器中请求视图激活，同时传入应用程序唯一编号 **UID**，视图 **UID**，以及其他相关信息，然后 **View** 服务器会自动启动通讯应用程序，并且直接切换到适当的视图。

其中，**MyAppUi** 类负责创建视图（**View**）并处理视图切换的请求，由于它派生于 **CAknViewAppUi**，因此可以用 **CAknViewAppUi** 所提供的方法来实现视图的注册、激活和取消激活。在同一时刻只有一个视图是处于激活状态。当用户请求激活视图时，会调用 **MyAppUi** 的视图切换命令 **ActivateViewL()**。**MyAppUi** 根据传入的视图 **ID** 通知视图服务器调用 **DoActivateL()**方法激活指定视图。视图主要用于处理自己菜单命令。

在图 3-5 中，**MyAppUi** 管理两个视图，分别是主界面(**MainView**)和播放界面(**PlayView**)。而每一个视图的容器都负责自己控件的创建销毁，控件间的转换，

以及按键事件处理。如 PlayContainer 里包含的控件有进度条(Progressbar)、文字箱(TextBox)等。

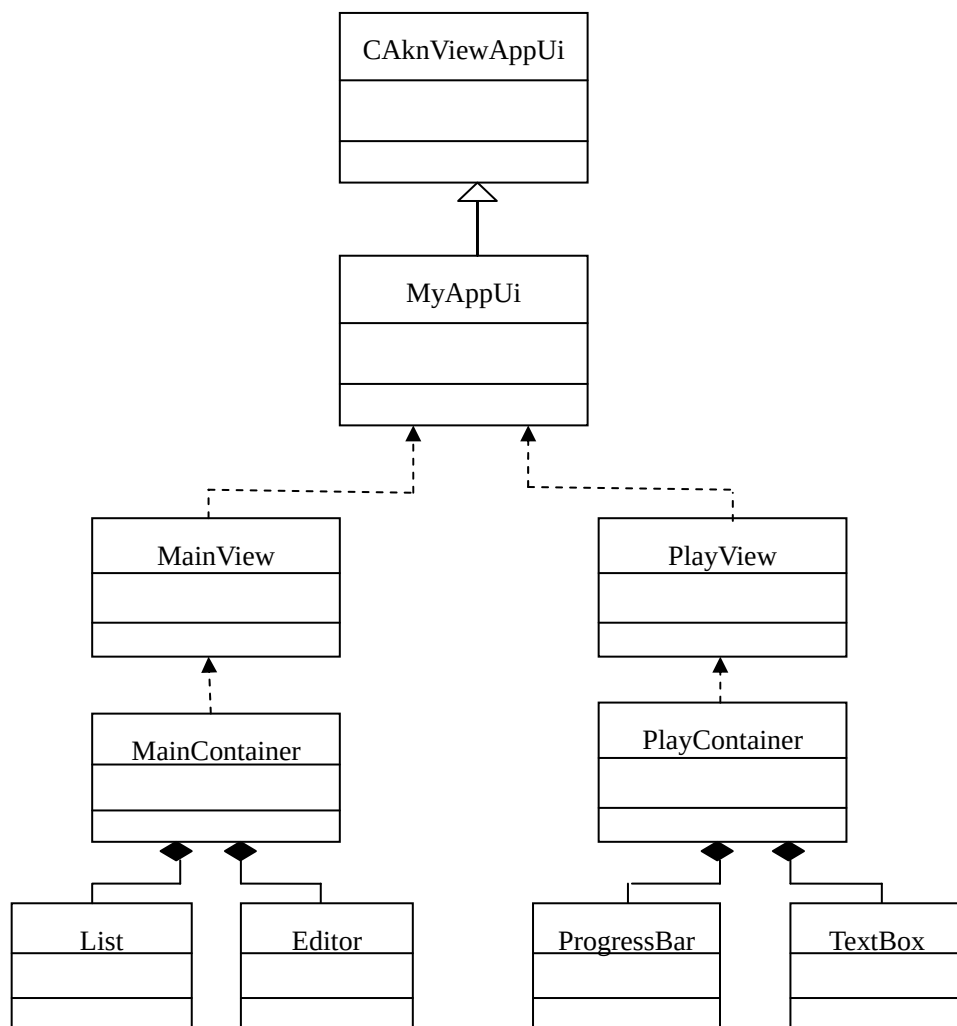


图 3-5 UI 视图

Symbian OS 库提供了大量的 UI 标准控件，如对话框、列表、菜单栏等等。但是当标准控件无法支持多媒体应用程序的 UI 时，可以根据需要自行设计所需控件。大多数 Symbian OS UI 控件派生自 CCoeControl 类，CCoeControl 是控件的基类，直接由它派生自定义控件实现比较繁琐；而一些标准控件也可以用作自定义控件的基类。例如 CEikListBox 是所有列表框控件的基类，可以通过将 CEikListBox 作为基类，创建出各种类型的列表框。

3.2.3 数据存储模块

在移动多媒体软件中，许多数据需要存储在手机存储器里，比如系统信息数据、位图文件、音频文件等等。手机上的存储器一般有两种：一种是内部固化的 NandFlash 芯片，通常容量在 2MB 到 64MB 不等，不过很少有超过 100MB 的；

另一种是外部存储卡，目前很多手机都支持扩展卡（通常是智能手机），扩展卡的容量一般在 128MB 到 8GB 不等。Symbian OS 存储系统支持最多 26 个驱动器，从 a:到 z:。其中 z:驱动器总是保存为系统 ROM，c:驱动器为内部读写驱动器，d:以后的驱动器可以是内部的，也可以是外部存储卡。

Symbian OS 提供了文件服务器让应用程序能够操作驱动器、目录和文件以及在文件中读写数据的基本服务。与 Symbian 其他服务器一样，文件服务器使用基于会话的通信，将客户端操作转换为发给服务器的消息。请求的函数在服务器中执行，然后将结果传回客户端。

3.3 功能模块设计

MP3Player 的功能模块如图 3-6 所示：

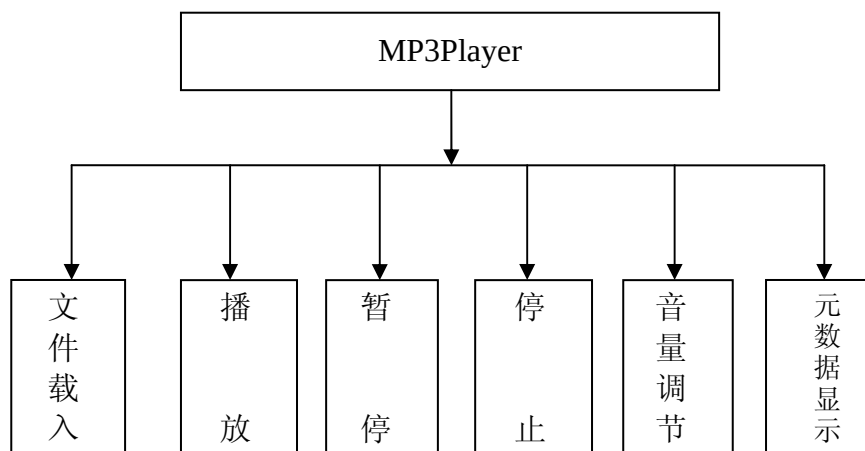


图 3-6 MP3Player 功能图

3.3.1 载入

载入是比较简单的工作，可以分为两种方式：MapcInitComplete() 和 MapcPlayComplete()。

如果需要指定 MP3 文件，需要使用第一种情况。CMdaAudioPlayerUtility 提供了 NewFilePlayerL() 函数构造并初始化音频播放器用例的一个新实例，用于播放文件（aFileName）中的采样音频数据。函数都必须得到获支持格式的音频数据，且当不能创建该对象时抛出异常。

第二种情况只是实例化播放器对象，不具体指定 MP3 文件，需要使用 CMdaAudioPlayerUtility 成员函数 NewL() 函数，当创建失败时抛出异常。但使用 NewL() 函数创建了该实例，就需要打开该文件以便可以回放该文件。可以通过向其传递文件的完整路径信息或该文件的描述符来打开一个文件，可以使用 OpenFileL() 函数从某个文件（aFileName）中打开一个音频片段。音频数据必须具备某种获支持的格式。如果之前有未完成的打开操作，这些方法将抛出

KerrInUse 异常。需要注意的是 MP3 描述符需要在数据开头处包含 mime 类型信息。

3.3.2 播放

打开 MP3 文件后，就可以播放这个音频文件了。文件播放是通过调用 CMdaAudioPlayerUtility::Play()实现的，这个方法将从音频文件的当前位置开始回放音频数据。

每当打开 MP3 文件后，在 MapcInitComplete()回调中要把 iState 变量设为 EReadyToPlay，并在开始播放文件后立即把 iState 变量设为 EPlaying。

回放过程一旦顺利完成或被 Audio Policy 叫停，MapcPlayComplete()方法将会得到 KerrNone（表示操作完全成功）或某个可能的出错代码，KErrCorrupt 表示数据已损坏，KErrInUse 说明音频设备正在被具有更高优先级的客户使用。如果当前的音频任务正被具有较高优先级的其他音频任务所抢占时，也会在播放期间返回这个出错。KErrNoMemory 表示没有足够的内存来播放这段音频采样。

对于 CMdaAudioPlayerUtility 类，会在两种情况下完成播放操作：该用例 (utility)已经播完了该 MP3 文件，或在播放中途出错。完成所有播放及/或不再使用已加载音频文件时，应该执行 CMdaAudioPlayerUtility::Close()清除操作。这个方法关闭当前的 MP3 文件，允许打开别的文件，因而 iState 变量必须被设定为 ENotReady，因为该音频文件已不可再回放。

3.3.3 停止

停止播放 MP3 文件只需调用 CMdaAudioPlayerUtility::Stop()函数，操作完成后，播放位置被移到该音频片段的起始处。在尚未载入文件之前不调用该方法，否则会造成 CMdaAudioPlayerUtility 的异常出错。为确保安全，一种做法是把停止功能设为“失效”，直到载入进程开始。

将 MP3 文件播放操作定为异步操作，应用对象可通过调用合适的函数在操作结束前停止其操作。

3.3.4 暂停

只要使用 CMdaAudioPlayerUtility::Pause()函数就可以暂停音频文件的回放；该音频文件的当前播放位置会被记录下来，以便调用 PlayL()继续播放。

暂停期间可能会出错，此时 Pause()函数会返回一个 Tint 值表示任何可能的出错。当没有任何出错时，iState 变量才被设定为 EReadyToPlay，这么做的原因是：暂停后音频文件仍然处于可回放状态。

3.3.5 音量调节

音量调节功能是MP3播放的必须功能之一，以改变播放文件音量的大小，适

应各种需求。音量调节需要事先定义SetVolume(TInt iVolume)、IncreaseVolume()和DecreaseVolume()等函数，本文定义了10个音量格，每次按键可以增大或者减小一格，具体实现过程将在软件实现中讨论。

3.3.6 元数据显示

随着音频文件一些新格式的出现，一些新功能也被添加到这些文件中。其中一项功能就是支持嵌入在文件中的元数据。元数据信息是内嵌在某些音频文件中的音轨信息，通常包括歌曲标题，歌手或艺术家的名字及音乐流派信息等。

加载一个需要播放的文件，同时要确保播放该文件是必须安全的资源。从这个文件读取元数据信息的过程直截了当。第一步是初始化 Play Object 并向其加载一个音频文件，创建实例之后，下一步就是打开该音频文件。一旦对那个对象进行了初始化并把音频数据正确加载到这个对象后，还需要考虑该文件中含有多少项元数据。提取出每一项元数据，并将其与一组预先设定的条目名进行匹配。如下所示：

KMMFMetaEntrySongTitle: Title of song. 歌曲标题；

KMMFMetaEntryArtist: Name of singer or artist. 歌手或艺术家名字。

为确定所提取的值属于哪个条目名，以及它所代表的是什么信息，名字和值必须提取自各个条目。

第四章 MP3Player 的实现

4.1 动态菜单栏的实现

本文采用动态菜单，即根据不同的播放状态，菜单的具体内容选项不相同。首先定义了菜单显示内容，主要包括：打开 MP3 文件、播放文件、暂停播放、停止播放、增大音量、减小音量、关于 MP3 文件和退出。

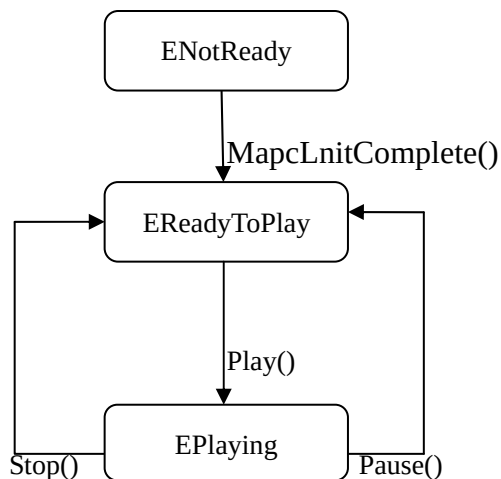


图 4-1 文件播放状态转换图

由于采用动态菜单，为此定义了三种状态，如图 4-1。根据状态不同显示三种不同的菜单选项。但是“关于 MP3 文件”和“退出”两选项为一直显示的信息。不同状态的菜单栏如图 4-2 所示。图(a)，在 ENotReady 状态时只有打开 MP3 文件选项显示；图(b)，在 EReadyToPlay 状态是有打开 MP3 文件、播放文件、暂停播放、停止播放、增大音量、减小音量选项；图(c)，EPlaying 状态是有打开 MP3 文件、暂停播放、停止播放、增大音量、减小音量选项。



图 4-2 MP3Player 动态菜单栏界面图

视图功能主要是通过调用的 SetItemDimmed(TInt aCommandId,TBool aDimmed)函数和定义的 UpdateMenuL()、HandleCommandL()等函数来完成。

4.2 MP3 元数据的显示

本文定义现实的元数据主要是歌曲名称和歌曲演唱者。调用了 `GetNumberOfMetaDataEntries(TInt& aNumEntries)` 函数，来提取每一项元数据信息，并返回一个 `CMMFMetaEntry` 对象，其中含有标记名及其值。调用 `GetMetaEntryL()` 函数来判断操作是否成功，函数给出具体的出错代码：用 `KerrNotFound` 抛出异常，表示所选择的元数据条目不存在；如果控制器不支持这种格式的元数据信息则用 `KerrNotImplemented` 抛出异常；这个函数也能以其它系统级出错代码抛出异常。

下面给了元数据读取的时序图：

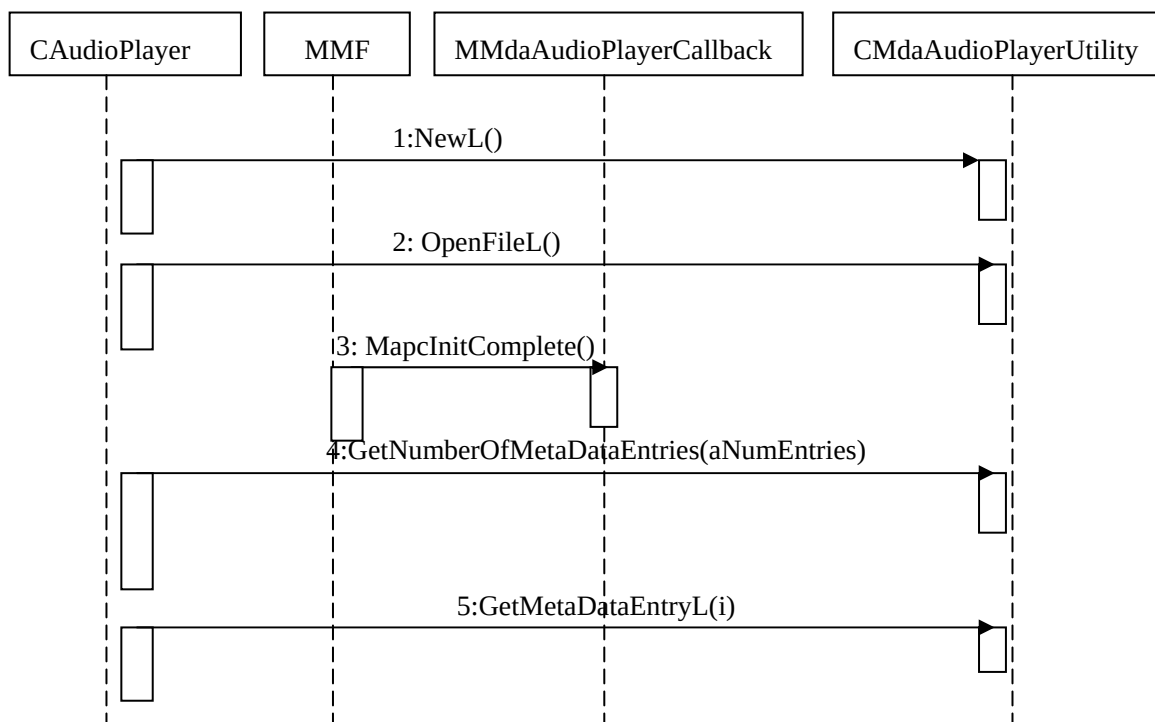


图 4-3 元数据读取时序图

图 4-4 为元数据的三种不同的界面。元数据可以显示中文和英文，图 4-4(a)、(b)分别为中英文的界面。若元数据读取失败，则显示“歌曲名：未知；播放者：未知”信息，图 4-4(c)所示。



图 4-4 MP3Player 元数据界面图

4.3 音量调节

本文音量调节采用 10 格，每次调节可以增大或者减小一个音量。首先调用函数 `MaxVolume()` 读取最大音量，将音量均分为十份，音量从小到大标记为整形数字 1-10。由于当前值不一定为整型，所以强制转换过程中数据类型变为整形。

音量调节具体流程如图 4-8 所示，载入文件后判断文件当前状态，若 `iState=ENotReady` 则重新载入，否则读取当前音量。或当前音量大于零则可以增加音量，若小于 10 则可减少音量。

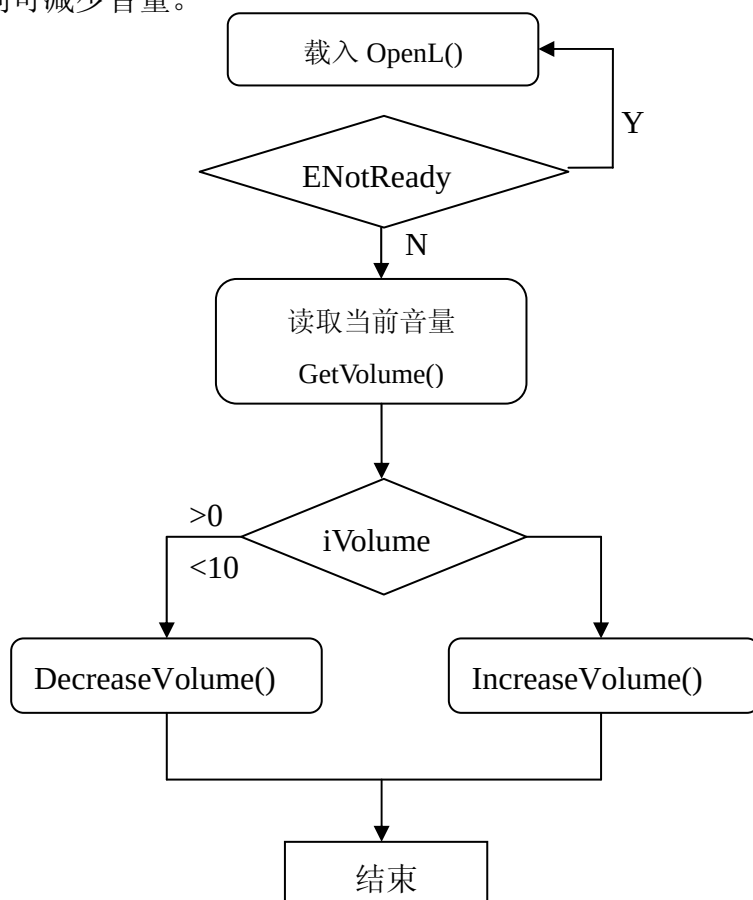


图 4-8 音量调节流程图

4.4 二阶段构造

如果按照传统的 C 或 C++构造函数会产生一个内存泄漏的隐患，就是当内存不足时会导致创建子类对象失败时，该构造函数不能返回出一个错误代码以表明该构造函数没有正常完成，但 new 操作符一样会返回一个指向复合类的无效的指针。在析构该复合类实例时，由于指针的不确定性便会引起内存泄漏或混乱，从而导致程序崩溃。

Symbian OS 提供的解决方案是在 C++构造函数实行二阶段构造。两阶段构造机制的核心思想就是，使类的构造函数不会产生异常，将那些可能产生异常的初始化操作从构造函数中抽出，放在一个稍后执行的函数（第二阶段构造函数）中。这样一来，类的构造实际上就分为两个阶段了。依照惯例，第二阶段构造函数统一命名为 ConstructL。

我们以类 CMyPlayer*为例。如果它遵循两阶段构造，那么要完整地构造它，就要用到下面的代码：

```
CMyPlayer* p=new(ELeave) CMyPlayer ();
CleanupStack::PushL(p);
p->ConstructL();
```

我们看到，创建一个类，原来用一条语句，现在要用三条或四条语句。这对于类的使用者来说太麻烦了。为了解决这个问题，Symbian C++又为需要两阶段构造的类添加了两个静态函数：NewL()和 NewLC()。我称呼它们为构造器。这两个构造器的内容一般都是固定的，NewL()的定义如下：

```
static CMyPlayer* CMyPlayer::NewL()
{
    CMyPlayer* self=NewLC();
    CleanupStack::Pop();
    return self;
}
```

NewLC()的定义如下：

```
static CMyPlayer* CMyPlayer::NewLC()
{
    CMyPlayer* self=new(ELeave) CMyPlayer();
    CleanupStack::PushL(self);
    self->ConstructL();
    return self;
}
```

我们可以从以上定义中看出，用 NewL 构造器构造的对象，没有被压入清除

栈；用 NewLC 构造器构造的对象，已经被压入清除栈。

为 CMyClass 加入静态函数 NewL()和 NewLC()后，我们再创建它，就可以用这样的代码：CMyClass* iMyClass=CMyClass::NewL();

或者：CMyClass* myClass=CMyClass::NewLC();

4.5 中文用户界面

由于搭建的环境只识别英文字符，为了方便使用所以需要汉化。汉化是在英文版的基础上修改完成的。主要包含界面资源与控件的汉化两种情况。

（1）资源汉化需要修改的为资源本地化文件，即.rls 文件，它用来存放本地化的语言文字。具体汉化过程如图 4-5 所示。

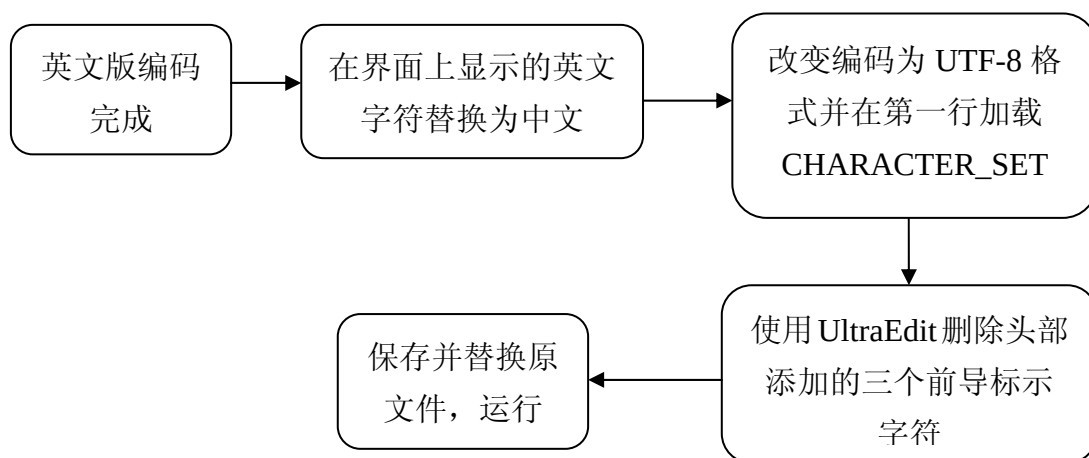


图 4-5 MP3Player 资源汉化流程图

其中改变为 TUF-8 格式可以使用记事本来修改，也可以通过使用 UltraEdit 等软件来修改。

（2）控件的汉化不仅需要改变编码，同时还要在具体的编码过程中实现。控件信息一般定义在头文件中，具体过程如图 4-6 所示：

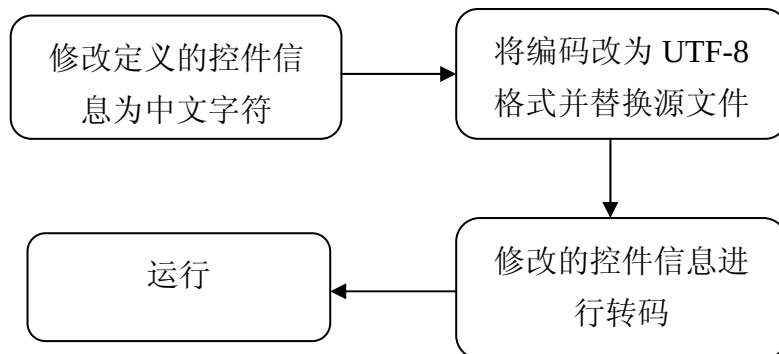


图 4-6 MP3Player 控件汉化流程图

需要说明的是修改控件信息为中文使是用 Lit8 来定义；转码时需调用函数

来分别对要修改的信息进行转码。

```
TBuf<255> temp;
```

```
CnvUtfConverter::ConvertToUnicodeFromUtf8(tmp,qtn_welcome);
```

图 4-7 展示了资源界面和空间界面的中英文界面：

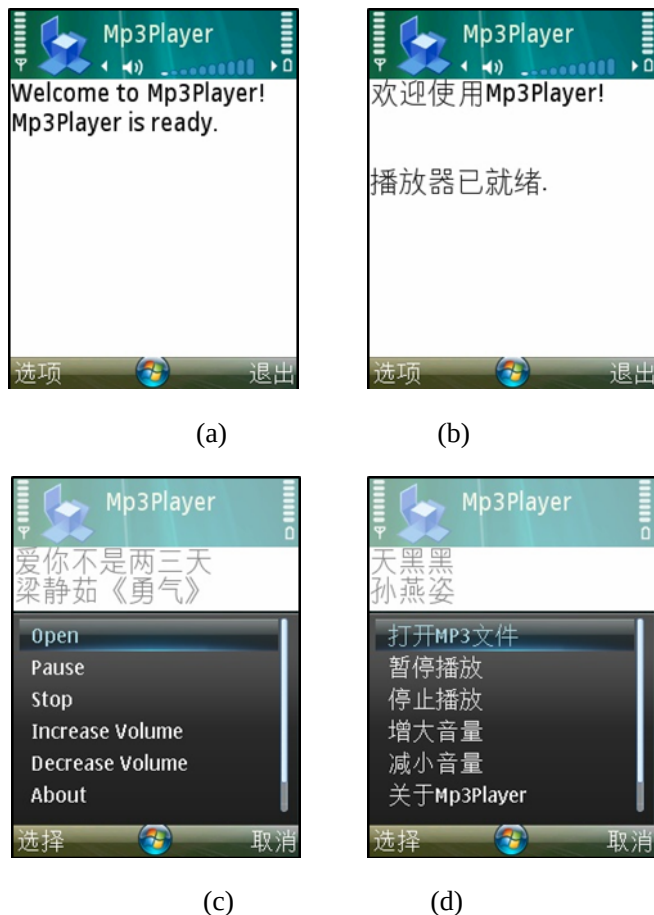


图 4-7 中英文资源和控件界面对照图

4.6 安装程序生成

Symbian S60 设备上应用程序的发布是通过安装包（SIS 文件）的形式进行的，设备上的软件安装器负责软件的安装工作。安装包包括应用程序文件与说明文件。其中应用程序是.SIS 或者.SISX 格式压缩文件存在的，说明文件则指导软件安装器进行软件安装过程。

Symbian 公司提供了创建 SIS 文件的一系列工具，包括文件生成器、证书生成器、安装文件签名和安装文件创建器。

具体的打包过程：

(1) 针对设备对应用程序进行编译。在命令行模式下进行编译，需要使用命令：

```
bldmake bldfiles
```

```
abld build<target>[udeb | urel]
```

第一行命令生成 `abld.abt` 文件和工程的 `makefiles`，第二行命令调用生成的 `abld.abt` 文件对目标进行编译。

（2）定义打包文件。打包文件的作用是说明 SIS 文件的生成方式，是一个文本文件，其后缀名是 `.pkg`。包含的信息：支持的语言，SIS 组件的名字、UID 和版本信息，开发者信息等。安装文件是单个压缩文件。

（3）生成 SIS 文件。可以在命令行模式下使用命令来生成 SIS 文件，通常命令行会以一个打包文件输入，输出 SIS 文件。本文使用命令：

```
Makesis MP3Player.pkg
```

命令执行完毕后生成了 `MP3Player.SIS` 文件。

（4）对 SIS 文件进行签名。如果把 SIS 文件安装到 S60 系统的手机上还需要对 SIS 文件进行签名。需要利用 `makekeys.exe` 命令创建证书/密钥对，然后使用 `sign.exe` 命令来对 SIS 文件进行签名。

（5）把签名后的 SIS 文件安装到目标设备上。可以通过蓝牙、红外或者 USB 线将 SIS 文件传输到设备上，系统软件安装器会自动进行安装。

第五章 结论

5.1 总结

智能手机发展迅速，Symbian OS 作为应用最广泛的智能手机操作系统，具有良好的市场前景。MP3 播放是智能手机的重要应用之一，MP3 播放器的性能不仅在一定程度上决定了音频质量的高低，还决定了手机内存的消耗，当占用内存过大时容易导致手机死机甚至崩溃。本文通过研究基于 S60 第三版的 MP3 播放器，熟悉了 Symbian 系统，培养了独立的开发能力。

本文首先讨论了 Symbian 开发基础，研究了应用程序框架和多媒体框架并建立了 S60 的开发环境。接着，本文应用观察者模式对 MP3Player 进行了设计，将其体系结构分为播放模块、UI 模块和数据存储模块。然后，本文对播放模块进行了详细的分析，将播放模块进一步划分为载入文件、播放、停止、暂停、音量调节和显示元数据信息等功能模块。最后，本文对实现 MP3Player 的关键技术进行了研究，主要包括动态菜单栏的实现、MP3 元数据的现实、音量调节和中文用户界面等等。

5.2 展望

由于时间和能力所限，MP3Player 还有很多不成熟的地方，例如播放列表、均衡器、网络下载歌曲等等功能，还需要在以后的工作中不断改进。

相信随着科技的不断进步，Symbian OS 的智能手机越加普及，MP3 播放软件有个良好前景。

参考文献

- [1]舒华英, 胡一闻等. 移动互联网技术及应用[M]. 北京: 人民邮电出版社, 2001. 3—29.
- [2]刘阳. 基于 Symbian OS 的面向手机 3D 游戏的图形引擎设计与实现[D]. 成都: 电子科技大学, 2007.
- [3]董佩佳. 基于 Symbian OS 智能手机应用软件开发[D]. 成都: 成都理工大学, 2007.
- [4]吴小波, 金纯等. 基于 Symbian OS 智能手机游戏开发的研究[J]. 计算机与数字工程, 2007, 35(3) : 67—67.
- [5]王康. 基于 Symbian OS 的 S60 手机游戏开发. 天津: 天津大学, 2005.
- [6]姚盛旺. Symbian OS C++程序开发[J]. 计算机与数字工程, 2007, 35(1): 65—66.
- [7]刘晰宇, 陈晓光, 刘斌. 基于 Symbian OS 的手机开发与应用实践[M]. 北京: 清华大学出版社, 2008. 86—86.
- [8]Richard Harrison 等著, 周良忠译. Series 60 应用程序开发[M]. 北京: 人民邮电出版社, 2006. 5—5.
- [9]杨常青, 彭木根. Symbian OS 手机程序开发与使用教程[M]. 北京: 机械工业出版社, 2007. 19—19.
- [10]马建, 陈健, 牛建伟等. 智能手机系统编程——Symbian 及 60 系列[M]. 北京: 科学出版社, 2005. 78—83.
- [11]Jo Stichbury 著. Symbian OS C++高效编程[M]. 北京: 人民邮电出版社, 2006. 110—115.
- [12]李佳昱. 基于 Symbian OS 的 ORC 通用应用程序框架设计与实现[D]. 北京: 北京邮电大学, 2008.
- [13]Nokia. Symbian OS: Multimedia FrameWork And Other Multimedia APIs. 2005.
- [14]Paul Coulton, Reuben Edwards, Helen Clemson 著. 朱晓暄, 何楠, 王文君译. S60 编程指南[M]. 北京: 人民邮电出版社, 2009. 18—18.
- [15]Forum Nokia Carbide. c++ Introductory White Paper [Z]. version 1.1: August13, 2007.

致 谢

本论文是在导师张坤龙副教授的悉心指导下完成的。衷心感谢张老师。在论文的选题、软件设计以及论文写作过程中，张老师尽心尽力，考虑全面，帮助热心，指导认真。张老师严谨的治学态度和深厚的专业知识，使我受益颇多。再次感谢张老师朋友般的关怀。

感谢同宿舍黄世悟、刘亭玉等同学在生活和学习上对我的帮助。黄世悟同学在论文的完成上给与了大量帮助。在此对北邮的同学李智杰一并表示感谢。

感谢爸爸妈妈对我的支持，谢谢所有关心我的人。