

第4章 多媒体应用设计原理

4.1 多媒体应用工程化设计

4.2 人机界面的设计

4.1 多媒体应用工程化设计

4.1.1 多媒体软件工程概述

多媒体技术在计算机领域的引入，使得计算机不仅可以处理文字、数字，还可以处理图像、声音、动画、视频等信息，增强了信息处理的种类和能力，再加上用户界面变得友好，极大地增强了系统的功能，扩展了应用市场。与其他应用系统相比，多媒体应用系统具有更加友好的用户界面，涉及的技术领域广、技术层次高，开发技术标准化，多媒体技术的集成化和工具化等特点。这些特点使得多媒体应用软件的设计已经不再是以程序设计为主，而是以创作为主。

从程序设计角度看，多媒体应用设计仍属计算机应用软件设计范畴，因此可借鉴软件工程施工方法进行。而软件工程是一种以系统的方法来开发、操作、维护及报废计算机软件的过程。软件从设计到完成都可以用一种生命周期模型来描述，常见的软件开发模型有瀑布法(Waterfall Method)和螺旋法(Spiral Method)两种。

4.1.2 两种典型的软件工程施工模型

1. 瀑布法

瀑布法是传统的软件生命周期模型中最典型的一种。该模型规定了各项软件工程施工活动，包括：制定开发计划，进行需求分析和说明，软件设计，程序编码，测试及运行维护。它们是自上而下，相互衔接的固定次序，如同瀑布流水，逐级下落，故此得名，如图4-1所示。

然而软件开发的实践表明，上述各项活动之间并非完全是自上而下，呈线形图式，实际情况是，每项开发活动均应具有以下特征：

(1) 从上一项活动接受本项活动的工作对象，作为输入

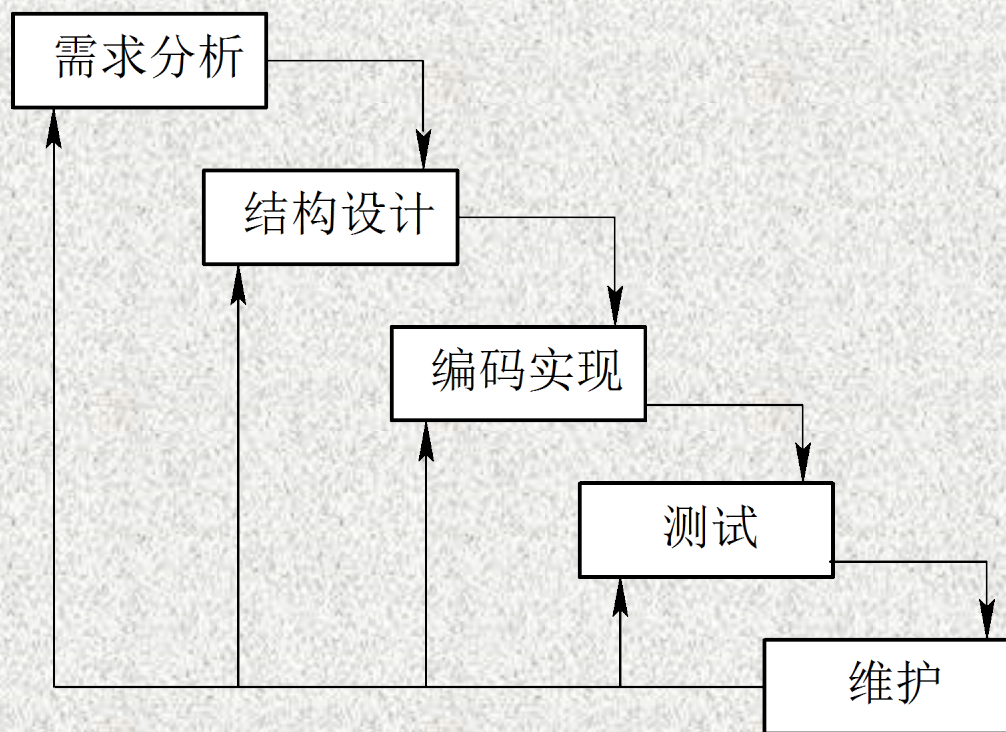


图4-1 瀑布式生命周期模型

(2) 利用这一输入实施本项活动应完成的内容。

(3) 给出本项活动的工作成果，作为输出传给下一项活动。

(4) 对本项活动实施的工作进行评审，若其工作得到确认，则继续进行下一项活动，否则返回前项，甚至需对更前项的活动进行返工。

2. 螺旋法

鉴于瀑布式生命周期的缺点，科学家布恩(Boehm)提出了称为螺旋式生命周期的模型概念，如图4-2所示。整个计划起始于图中饼形的中心，然后绕着中心做 360° 的旋转，每旋转一圈便是一个原型版本，也是其外围各圈的一个过渡性版本。对整个系统而言，它是开发过程中的一个步骤。

螺旋模型中，从第一步开发到第五步，便是一个版本，从第六步可构成一个循环。起初的循环内容较简单，功能也较少，每循环一次，功能增强一些，核心仍是初始计划。这种不断循环(即模型图描出的旋转)可以大大节省开发与维护的时间，降低开发成本，因为每一次新的循环都是在对前几次循环的累加、完善与维护的基础上进行的，整个生命周期便是一个不断革新的原型。

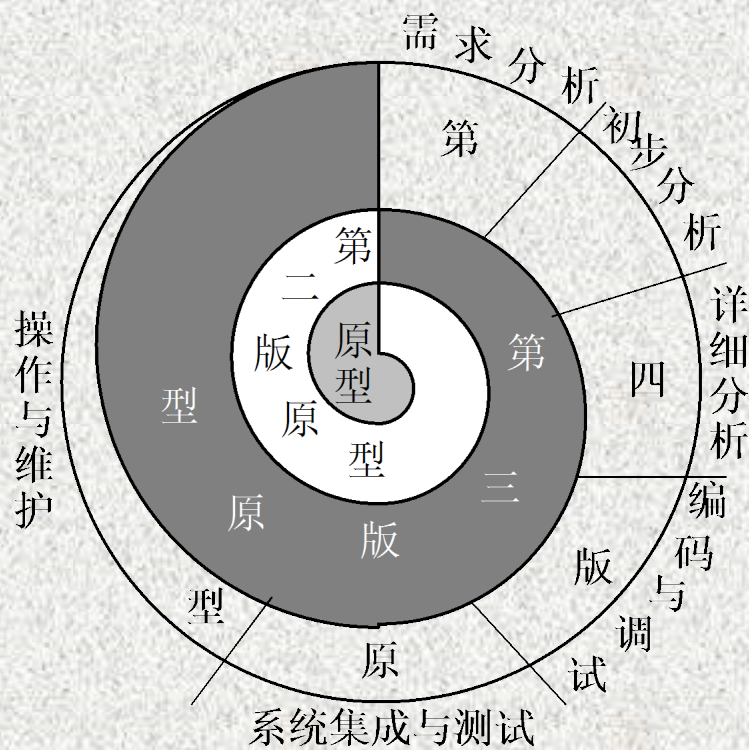


图4-2 螺旋式生命周期模型

在螺旋模型中，允许设计者很快地根据用户的需求描出最早的软件版本，然后让用户使用一段时间，并评估其正确性与可用性后给予反馈。因此用户便融入整个开发组内，形成该软件开发的一部分。这种描述可以在很短的时间内完成，形成一个原型，这个原型可以在下一个版本出来时将其抛弃。虽然这个原型在功能上近似于最后版本，但缺乏细节，没有错误恢复能力。如果对原型进一步进行某些细部的开发和修正，便形成下一个版本。如此反复地开发与修正，最后的版本也就是成品了。其实最后版本也可能成为原型，因为若再有功能加强或适应性需求等因素加入时，它仍可能产生下一个版本。

4.1.3 多媒体应用系统的设计过程与设计原则

1. 需求分析

一个多媒体应用系统总是从某种想法或需要开始的。开始之前，首先必须定出它的范围和内容，让该应用系统在我们的头脑中大致定形，然后制定出一个计划。

需求的来源往往出于一种需要。需求分析是创作一种新软件产品的第一阶段，也是软件产品生命周期的一个重要阶段。该阶段的任务就是对整个系统的需求进行评估，确定用户对应用系统的具体要求和设计目标。为使一种想法或构思得到充分的实现，可进一步从以下几个方面考虑。

。

1) 问题识别

可用画草图的方式尽可能地将所有相关的信息表示出来，或者用详细列出构思的方式从各种不同的角度来分析问题，以期获得各种不同的结论。

2) 分析与综合

分析员需从数据结构出发，逐步细化所有的软件功能，找出系统各元素之间的联系、接口特征和设计上的限制，分析它们是否满足功能要求，是否合理。依据功能需求、性能需求、运行环境需求等，剔除其不合理部分，增加其需求部分，最终综合成系统解决方案，给出目标系统的详细逻辑模型。

3) 编制软件需求规格说明书

该过程实现对已确定的需求分析的清晰准确的描述，文档通常有数据描述、功能描述、性质描述、质量保证及加工说明等。该文档既是软件系统逻辑模型的描述，也是下一步进行设计的依据，应具有准确性和一致性，并且尽量采用标准图形、表格、符号，使设计者一目了然。同时，为了明确表达用户对输入/输出的要求，还需要制定数据要求说明书及编写初步的用户手册，着重反映准备开发软件的用户界面和用户使用的具体要求。

4) 需求分析评审

评审的目的在于确认各种可能的方案是否能真正使问题得到解决。因此，必须将之与原来用户的需求相互对照并且列出其中应有的功能，请真正的用户来判断这些方法的正确性。要对所有的方案进行筛选，舍弃不正确的方案，从众多的分析方案中找出一个可行性高而最有价值(创意新颖)的方案。

2. 应用系统结构设计

当通过需求分析，确定了设计方案后，就要决定如何构造应用系统结构。需要强调的是，多媒体应用系统设计后，必须将交互的概念融于项目的设计之中。在确定系统整体结构设计模型之后，还要确定组织结构是线性、层次、网状链接，还是复合型，然后着手脚本设计，绘制导向图，并通过脚本与导向图的很好结合来确定如下内容：

(1) 目录主题。目录主题即项目的入口点，一旦目录主题选定，即同时设定了其他主题内容，所以应以整个项目为一体，形成一致而有远见的设计。

(2) 层次结构和浏览顺序。许多时候，信息所表示的是前一屏幕的后续部分而不是其他层的信息内容，故此需建立其浏览顺序，使用户更好地理解内容。

(3) 交叉跳转。通常我们要把相关主题连接起来，可采用主题词或图标作为跳转区，并指定要转向的主题。但交叉跳转功能需慎重使用，大量跳转使用户能随意浏览信息，但会使查找过于复杂，而且要花费许多时间对跳转进行检测以保证跳转的正确性。

3. 建立详细设计的标准和细则

在开发应用系统之前必须制定高质量的设计标准，以确保多媒体设计具有一致的内部设计风格。这些标准主要有：

(1) 主题设计标准。当把表现的内容分为多个相互独立的主题或屏幕时，应当使声音、内容和信息保持一致的形式。例如，决定是要用户在一个主题中用移动屏幕的方法来阅读信息，还是限制每个主题的信息量，使其在标准窗口中显示。

(2) 字体使用标准。利用Windows提供的字型、字体大小和字体颜色来选择文本字体，使项目易读和美观。

(3) 声音使用标准。声音的运用要注意内容易懂，音量不可过大或过小，并与其他声音采样在质量上保持一致。

(4) 图像和动画的使用。选用图像，要在设计标准中说明它的用途。同时要说明图像如何显示及其位置、是否需要边框、颜色数、尺寸大小及其他因素。若采用动画则一定要突出动画效果。

在开发应用系统之前制定高质量的设计标准，需要花费一定时间。但按照精心制定的标准工作，不仅会使项目的外观更好，也使它易于使用和推广。

4. 素材的准备

准备多媒体素材是一项十分重要的基础工作。在一般的多媒体系统中，文字的准备工作的比较简单，所占的存储量也很少，即使是100万的汉字，也不过占2兆字节，因此在一个多媒体系统中，基本可以不考虑文字所占用的存储空间。但另外几种媒体信息，例如声音、动画和图像等占用的存储空间就比较大，准备工作也较复杂。对图像来说，扫描处理过程十分关键，不仅要进行剪裁处理，而且还要在这个过程中修饰图像，拼接合并，以便能得到更好的效果。对于声音来说，音乐的选择、配音的录制也要事先做好，必要时也可以通过合适的编辑做特殊处理，如回声、放大、混声等。其他的媒体准备也十分类似，如动画的制作、动态视频的录入等。最后，这些媒体都必须转换为系统开发环境下要求的存储和表示形式。

5. 制作生成多媒体应用系统(编码与集成)

在完全确定产品的内容、功能、设计标准和用户使用需求后，要选择适宜的创作工具和方法进行制作。目前的多媒体应用系统开发工具可分为两大类：基于语言的编程开发平台和基于集成制作的创作工具。

在生成应用系统时，如果采用程序编码设计，首先要选择功能强、可灵活进行多媒体应用设计的编程语言和编程环境，如VB、VC++和Java等。这需要经过编程学习和训练之后才能胜任。有经验的编程人员可较好地完成设计要求，精确地达到设计目标。而训练有素的程序员则能熟练地采用工程化设计方法，缩短开发周期。

由于进行多媒体应用系统制作时要很好地解决多媒体压缩、集成、交互及同步等问题，编程设计不仅复杂，而且工作量大，使无编程经验的人望而却步，因此多媒体创作工具应运而生。各种创作工具虽然功能和操作方法不同，但都有操作多媒体信息进行全屏幕动态综合处理的能力。根据现有的多媒体硬件环境和应用系统设计的要求选择适宜的创作工具，可高效、方便地进行多媒体编辑集成和系统生成工作。

具体的多媒体应用系统制作任务可分为两个方面：一是素材制作，二是集成制作。素材制作是各种媒体文件的制作。由于多媒体创作不仅媒体形式多，而且数据量大，制作的工具和方法也较多，因此素材的采集与制作需多人分工合作。如美工人员设计动画，程序设计人员实现制作，摄像人员拍摄视频影像，专业人员配音等。但无论文本录入、图像扫描，还是声音和视频信号采集处理，均要经过多道工序才可能进行集成制作。

集成制作是应用系统最后生成的过程。许多多媒体/超媒体创作工具，实际上是对已加工好的素材进行最后的处理与合成，即就是集成制作工具。设计者面对所选用的创作工具或开发环境应有充分的了解和熟练的操作，才能高效地完成多媒体/超媒体应用系统的制作。

集成制作应尽量采用“原型”和逐渐使之“丰满”起来的手法，即在创意的同时或在创意基本完成之时，就先采用少量最典型的素材对少量的交互性进行“模拟版”制作。因为多媒体产品的制作受到多种因素的影响，大规模的正式开工必须在“模拟版”获得确认之后方可进行，而在“模拟版”的制作过程中，实际上也已经同时解决了将来可能会碰到的各种各样的问题，所以“模拟版”制作是一种非常好的方法。

需说明的是，在多媒体创作中，素材准备占有大部分工作量，而集成制作工作量仅占整个工作量的1/3左右。在素材编辑量大的情况下，由于集成创作工具提供了高效方便的平台，使集成工作量只占整个工作量的1/10左右。目前绝大部分创作工具软件都是基于Windows环境下，其中许多创作工具还为多媒体应用程序提供了创作模式，这些不同的模式会影响到用其开发的多媒体应用程序的特征。

6. 系统的测试与应用

无论是用编程环境，还是用创作工具，当完成一个多媒体系统设计后，一定要进行系统测试。系统测试的工作是繁琐的，测试的目的是发现程序中的错误。测试工作实际从系统设计一开始就可进行，每个模块都要经过单元测试和功能测试。模块连接后要进行总体功能测试。开发周期的每个阶段，每个模块都要经过单元测试和功能测试。模块连接后要进行总体功能测试。开发周期的每个阶段，每个模块都应经过测试，不断改进。

对可执行的版本测试、修改后，形成一个可用的版本，便可投入试用，在应用中再不断地清除错误，强化软件的可用性、可靠性及功能。经过一段时间的试用、完善后，可进行商品化包装，以便上市发行。

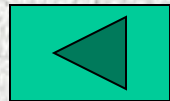
软件发行后，测试还应继续进行。这些测试应包括可靠性、可维护性、可修改性、效率及可用性等。其中可靠性是指程序所执行的和所预期的结果一样，而且前一次执行与后一次执行的结果相同；可维护性是指如果其中某一部分有错误发生时，可以容易地将之更改过来；可修改性是指系统可以适应新的环境，随时增减改变其中的功能；效率高则是程序执行时不会占用过多的资源或时间；可用性是指一项产品可以满足用户想要完成的全部工作。

经过上述应用测试后，再进行用户满意度分析，进而详细整理并除去影响用户满意的因素，完成开发过程。

7. 系统的维护

软件交付使用后，由于在开发时期需求分析不彻底，或测试与纠错不彻底，仍存在一些潜藏的错误，某些功能需要进一步的完善和扩充，所以要进行维护、修改工作，从而延长系统的生命周期。

软件维护的内容有：改正性(纠错性)维护(发现和改正潜藏的软件错误)；适应性维护(在硬件和软件的支撑环境改善的情况下，交付使用的软件系统也要做相应的修改，以适应新的系统环境)；完善性维护(用户在使用系统中提出了一些新的功能与性能要求而做的完善工作)；预防性维护(为了适应未来的软硬件环境的变化，主动地增加预防性的新版本功能)。



以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/738105020115007006>