**结构化设计和面向设计对象的区别和联系**

麻锦涛 16281262 计科1602

（1）结构化程序设计

结构化程序设计的基本思想是采用自顶向下、逐步细化的设计方法和单入单出的控制结构。其理念是将大型程序分解成小型、便于管理的任务。如果其中的一项任务仍然过大，则将它分解为更小的任务。这一过程将一直持续下去，直到将程序划分为小型的，易于编写的模块。

\*这种设计方式有着显著的一些问题：

程序难以管理、数据修改存在问题、程序可重用性差、用户要求难以在系统分析阶段准确定义，致使系统在交付使用时产生许多问题。

用系统开发每个阶段的成果来进行控制，不能适应事物变化的要求。

这种问题的根源在于数据和数据的处理不能分离。

（2）面向对象程序设计

面向对象（ObjectOriented ）是认识事务的一种方法，是一种以对象为中心的思维方式面向将系统看成通过交互作用来完成特定功能的对象的集合。每个对象用自己的方法来管理数据。也就是说只有对象内部的代码能够操作对象内部的数据。

\*面向对象的基本特征

1、抽象2、封装3、继承4、多态5、继承和多态  (3)面向对象程序设计优缺点

面向对象编程的优点：

易维护、质量高、效率高、易扩展

面向对象编程的缺点：运行效率会下降10%左右

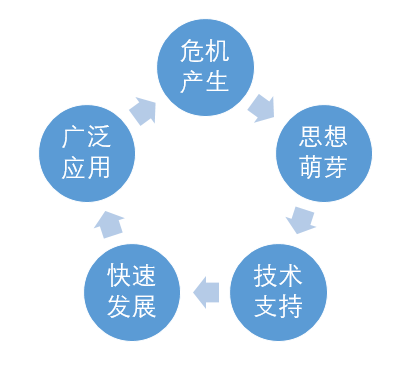
**二者的联系和区别主要有以下几点：**

**（一）从起源上看**

结构化方法与面向对象方法都起源于相应的程序设计思想和语言。20世纪60年代后期，《程序结构理论》和《GOTO陈述有害论》的提出，证明了任何程序的逻辑结构都可以用顺序结构、选择结构和循环结构来表示，确立了结构化程序设计思想，产生了如FORTRAN、PASCAL、C等语言。结构化方法把对程序的分析、设计，延伸至对项目工程的分析、设计，结合程序设计语言的技术支持，得以产生和发展。

20世纪80年代，随着应用系统的日趋复杂、庞大，结构化开发方法在工程应用中出现了一些问题。同期，面向对象程序设计思想经过20年的研究和发展逐渐成熟，一大批面向对象语言相继出现，面向对象方法自产生就广受青睐。90年代中期，互联网兴起，JAVA语言因跨平台特性得以蓬勃发展；21世纪初，不受限于时空的联合开发成为常态；今天，移动APP市场火爆，Andriod开发成为热点。面向对象方法已经成为软件开发方法的中坚力量。

结构化方法和面型对象方法的起源和发展具有模式一致性：

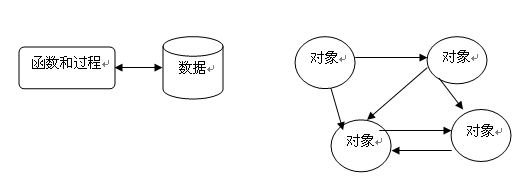


**（二）从思想上看**

　　结构化方法承袭了结构化程序设计的思想，把待解决的问题看作一个系统，用系统科学的思想方法来分析和解决问题。结构化方法遵循抽象原则、分解原则和模块化原则；以数据和功能为中心；以模块为基本单位；以算法为程序核心；强调逐步求精和信息隐藏。

　　面向对象方法的思想是模拟了客观世界的事物以及事物之间的联系。面向对象以类取代模块为基本单位；通过封装、继承和多态的机制，表征对象的数据和功能、联系和通信；通过对对象的管理和对象间的通讯完成信息处理与信息管理的计算和存储，实现软件功能。

　　对于结构化方法，模块由函数实现，完成对输入数据的加工和计算，数据和功能是分离的；而面向对象把数据和功能封装在对象中，形成一个整体。两种方法在数据和功能上的不同处理是其思想上的本质差别。



图二：结构化思想和面向对象思想

面向对象程序设计

　　从上例可见，结构化程序设计是一种过程式的“解题”的方式，程序关注且只关注对于输入数据，输出正确的结果，代码是算法的直接体现，代码效率高；面型对象程序设计是整体式的“建模”的方式，程序关注现实客体，而非某些数据，代码是功能的直接体现，复杂的算法往往是一两行库函数处理，代码效率低。

**（三）从应用上看**

　　结构化方法的实质是问题求解，结构化程序是由算法决定的，算法是程序员分析设计的，程序的执行过程主要是由程序员控制，而不是由用户控制；面向对象方法中，程序员设计的是对象属性及操作方法，但在什么时间、使用什么方式操作对象则是完全由用户交互控制。

　　结构化方法的建模工具难以表达交互性强的软件系统，程序设计融入系统的分析和设计中，处理大型系统时会过于复杂，甚至很难控制；面向对象方法的抽象机制提供了自然的建模方法，特别是能很好地把握对象之间复杂的相互关系。

　　结构化方法比较适合工程计算、实时数据的跟踪处理、各种自动控制系统等等； 面向对象分析更加适用于复杂的、由用户控制程序执行过程的应用软件，比如大型游戏软件以及各类管理信息系统软件[3]。

**（四）二者之结合**

　　经过上述分析，我们可知结构化方法和面向对象方法对于不同的软件系统各有优劣。结构化方法把解空间分数据和功能两部分，可以更加清晰地进行需求分析和功能分解，数据流图能够细致地说明数据在各个功能模块之间的流动和变化，更适于系统设计的前期阶段。设计人员清楚地了解数据和系统要求的操作后，面向对象方法能够把数据和功能以对象为单位封装成一个整体，更直观地表达对象的状态变化和对象间的交互，更加准确地分析功能的实现过程，更适于在软件后期细化系统的具体行为。基于此，设计的混合式软件开发方法如下：

　　1)使用SA进行需求分析，建立数据字典，构建总的和分层的数据流图。

　　2)使用模块结构图设计系统的独立功能块，做出模块内的程序流图。

　　3)结合数据流图，聚合同类模块，规约类，根据程序流图，设计类的属性和类的方法。

　　4)使用OOD建立系统的动态模型，分析对象的行为和协作。

　　5)总体面向对象程序设计，细节结构化程序设计优化，实现代码层。

　　使用混合式方法，我们能够充分利用两种方法的优点，扬长避短，提高开发的效果和效率。

　　无论是结构化方法，还是面向对象方法，都是用来解决日益矛盾的软件危机的系统方法。从直接开发，到结构化方法，再到面向对象方法，软件构件的愈发独立、可重用，开发在一个更高的层次进行，分析层、设计层和代码层关联性减少。这些都有利于系统开发员更加关注功能本身，提高软件质量。硬件性能的提高会使计算机的使用越发广泛，软件工作的环境更加复杂，软件的功能更加丰富，软件的性能更需提高，对软件开发方法提出了更多的要求，会涌现更高层次的新的方法。无论使用哪种开发方法，或者是混合哪几种开发方法，我们都要因地制宜，依据需求分析和系统要求，做出最好的选择或组合。