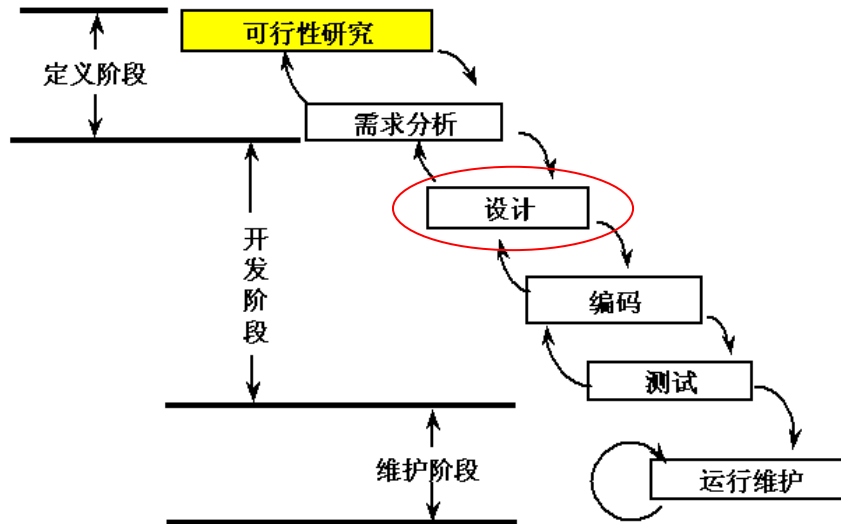


第五章 总体设计

Where we are?



一、总体设计概述

1、软件设计概述

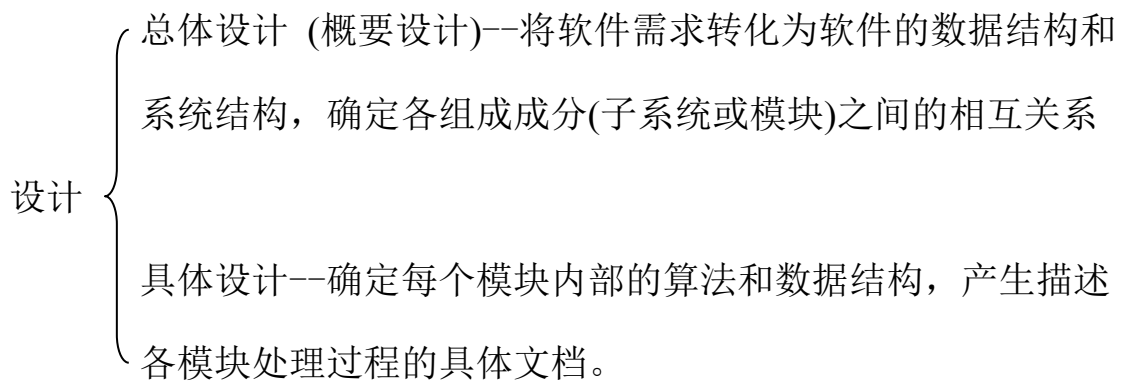
一旦软件需求确定之后，就进入开发阶段。开发阶段由三个相互关联的步骤组成：设计、实现（编码）和测试。每个步骤事实上都按某种方式进行信息变换，最终得到有效的计算机软件。其中“设计”活动是获得高质量、低耗费、易维护软件的一个最重要环节。

可行性分析 -- Why? Do or not? Who?

需求分析 -- What?

设计 -- How do?

在软件需求分析阶段已经完全弄清晰了软件的各种需求，较好地解决了目标软件系统要“做什么”的问题，并已在软件需求规格说明中详尽和充分地阐明白这些需求。下一步就要着手实现软件的需求，即要着手解决“怎么做”这个软件的问题。



在技术上，概要设计和具体设计又由若干活动组成。除上述的软件结构设计、数据结构设计和过程设计外，对现代应用软件，还应当包括一个独立的界面设计。

2、总体设计的任务

- ✚ 划分出组成系统的物理元素——程序、文件、数据库、人工过程和文档等等。但是每个物理元素仍旧处于黑盒子级，这些黑盒子里的具体内容将在以后细致设计。
- ✚ 设计软件的体系结构，也就是要确定系统是由哪些模块组成的，以及这些模块相互间的关系。

3、总体设计的必要性

可以站在全局高度上，花较少成本，从较抽象的层次上分析对比多种可能的系统实现方案和软件结构，从中选出最佳方案和最合理的软件结构，从而用较低成本开发出较高质量的软件系统。

二、总体设计的过程

典型的总体设计过程包含以下两个阶段，共 9 个步骤：

- ✚ 系统设计阶段，确定系统的物理实现方案

(1) 设想供选择的方案

在设想供选择方案时，以需求分析得到的数据流图为动身点，画出自动化边界，形成不同的实现的物理方案。P6-8

(2) 选取合理的方案 P9

(3) 举荐最佳方案 P10

✚ 结构设计阶段，确定软件的结构

(4) 功能分解，从实现的角度细化逻辑模型 P12

(5) 设计软件结构 P13

(6) 设计数据库 P14

(7) 制定测试安排 P14

(8) 书写文档 P15~16

(9) 审查和复审 P16

三、模块化设计原理

是软件设计过程中应遵循的基本原理和相关概念。包括：模块化、抽象、逐步求精、信息隐藏和局部化、模块独立。

1、模块与模块化

模块是软件中可独立命名和编制的部件，每个部件称为一个模块，当把全部模块组装到一起时，便可获得满意问题须要的一个解。

依据模块的定义，过程、函数、子程序和宏等，都可作为模块。面对对象方法学中的对象是模块，对象内的方法(或称为服务)也是模块。模块是构成程序的基本构件。

模块化就是把程序划分

成独立命名且可独立访问的模块，每个模块完成一个子功能，把这些模块集成起来构成一个整体，可以完成指定的功能，满足用户的需求。

✚ 模块化的依据：

$$C(P1+P2) > C(P1)+C(P2); \quad E(P1+P2) > E(P1)+E(P2)$$

C为问题的困难度，E为解题须要的工作量。P18

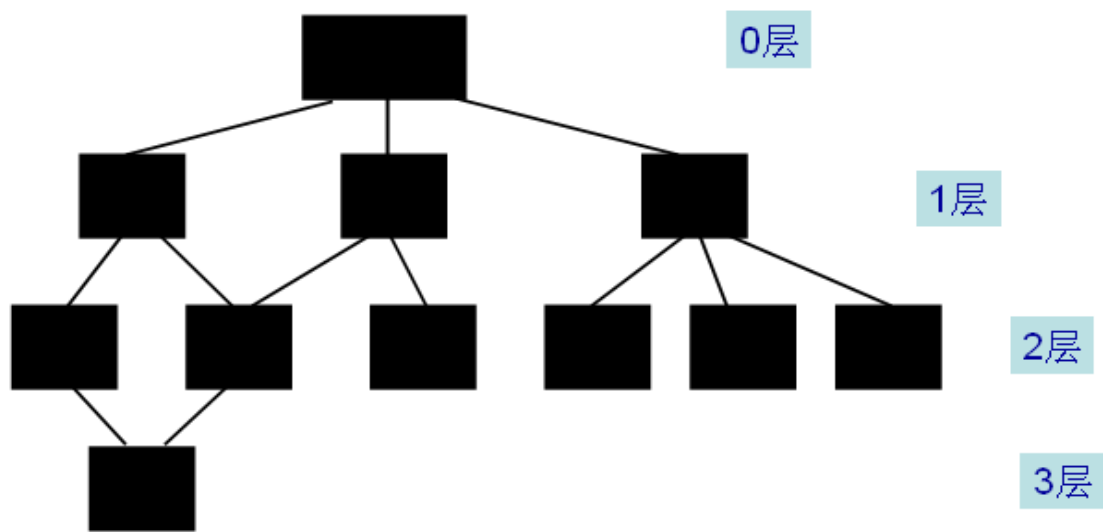
这个不等式导致“各个击破”的结论——把困难的问题分解成很多简洁解决的小问题，原来的问题也就简洁解决了。这就是模块化的依据。

✚ 模块化与软件成本： P20~P21

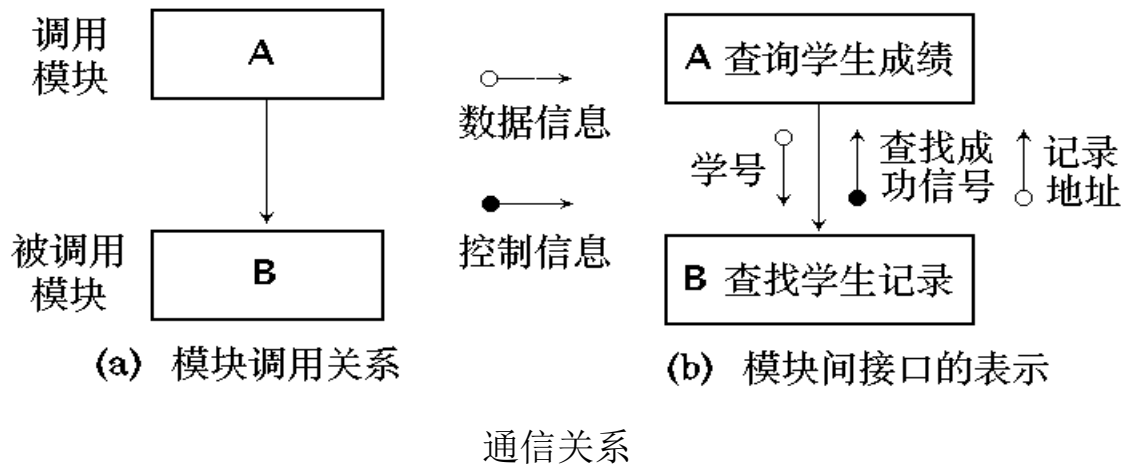
导致一个结论：应当模块化，但不能过分模块化。

✚ 模块化给软件设计带来好处： P22

✚ 模块间关系：层次关系、通信关系——软件结构是模块化的和分层次的



层次关系



2、抽象与逐步求精

抽象就是抽出事物的本质特性(共性)，而短暂不考虑它们的细微环节。

抽象是人类相识困难现象和解决困难问题时运用的思维方法，是限制困难性的基本策略。抽象要求人们将注意力集中在某一层次上考虑问题，而忽视那些低层次的细微环节。

✚ 抽象的特点

- (1) 忽视细微环节和差异 (2) 分层理解问题 (3) 自顶向下分解模块

逐步求精最初是由 Niklaus Wirth 提出的一种自顶向下的设计策略。依据这种设计策略，程序的体系结构是通过逐步精化处理过程的层次结构而设计出来的。通过逐步分解对功能的宏观陈述而开发出层次结构，直至最终得出用程序设计语言表达的程序。

✚ 软件工程过程的每一步都是对软件解法的抽象层次的一次精化；

P25

✚ 随着软件开发工程的推动，在软件结构每一层中的模块，表示了对软件抽象层次的一次精化。

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/376142215015010142>

