<<软件工程与计算>>

图书基本信息

书名：<<软件工程与计算>>

13位ISBN编号：9787111407508

10位ISBN编号：7111407504

出版时间：2012-12

出版时间：机械工业出版社

<<软件工程与计算>>

内容概要

《高等院校软件工程专业规划教材：软件工程与计算（卷2）：软件开发的技术基础》作为国家精品课程“软件工程与计算”系列课程的第二门课程配套教材，《高等院校软件工程专业规划教材：软件工程与计算（卷2）：软件开发的技术基础》以经典软件工程方法与技术为主线，软件开发技术与程序设计知识为教学重点，培养学生简单小组级别、中小规模软件系统的软件开发能力。

全书主要分为六部分。
第一部分介绍软件工程的基本框架。
第二部分介绍项目启动阶段的知识。
第三部分介绍软件需求开发的基础知识，包括软件需求工程的概要、软件需求的内涵、常见的需求分析方法、软件需求文档。
第四部分首先介绍软件设计的基础概念，之后沿着设计过程和设计技术两条主线，深入描述软件设计的相关知识。
第五部分介绍软件构造、测试、移交与维护等软件开发的下游工程的基础知识。
第六部分是对第一部分的延续，通过总结性回顾，进一步加深读者对软件工程的理解。

《高等院校软件工程专业规划教材：软件工程与计算（卷2）：软件开发的技术基础》可作为高等院校软件工程、计算机及相关专业本科生软件工程课程的教材，也可作为从事软件开发的相关技术人员的参考书。

<<软件工程与计算>>

书籍目录

章节摘录

版权页： 插图： “ 配置 ” 是对 “ 形式 ” 的发展，定义了 “ 部件 ” 以及 “ 连接件 ” 之间的关联方式，将它们组织成系统的总体结构。
按照这个模型，[Shawl996]给出了一个简洁的软件体系结构定义：“ 一个软件系统的体系结构规定了系统的计算部件和部件之间的交互。
” 理解软件高层结构需要注意的第一个内容：连接件是一个与部件平等的单位。
在软件的详细设计中，交互与计算是交织在一起的——过程、对象和模块是第一等级的软件抽象实体，实现交互的程序调用、消息协作、导人／导出等则都是嵌入在第一等级软件抽象实体内部的，处于附属和派生的地位。
而在软件体系结构中，连接件将交互从计算中独立出来进行抽象和封装，这使得实现交互的连接件与部件一样，都是第一等级的元素单位。
理解软件高层结构需要注意的第二个内容：部件与连接件是比类、模块等软件单位更高层次的抽象。
就像类既拥有抽象规格，又拥有 “ 数据结构+算法 ” 的具体实现一样，部件和连接件也既拥有抽象规格，又拥有 “ 模块+连接 ” 的具体实现。
9.2.2区分软件体系结构的抽象与实现 要正确理解软件体系结构就必须能够区分软件体系结构的抽象与实现。
以部件、连接件和配置为基本单位组织的模型就是软件体系结构的抽象，基本目的是描述软件系统的整体功能组织，不涉及程序设计语言提供的各种编程机制。
要最终建立软件产品，就必须考虑如何利用编程机制实现抽象的软件体系结构，这需要从部件、连接件、配置等单位向模块、构件、进程等传统单位进行转换。
模块、构件、进程等传统单位是依赖于编程机制的，它们组成的模型就被称为软件体系结构的实现。
软件体系结构设计是先使用抽象机制完成软件系统的总体功能部署，然后再将抽象模型等价转换为实现模型。
这既保证了软件系统的效用和质量（依靠抽象机制），又顺利实现了从总体结构设计到详细设计的过渡（依靠抽象机制向实现的转换）。
例如，[Allen1997]提到了一个简单的字符大写转换系统Capitalize，它将输入字符流中的候选字符转换为大写，并将其他字符转换为小写。
图9—1a就是以部件和连接件抽象规格形式表示的Capitalize体系结构。
split、upper、lower和merge四个部件（图中表现为矩形框），分别进行读取并分割字符、大写转换、小写转换与重新拼合并输出。
连接件是连接这四个部件的管道（图中表现为线），分别是split→upper、split→lower、upper→merge和lower→merge，它们负责完成字符流的传送。
整个结构的布局体现了软件体系结构的总体功能组织，这就是配置。
很明显，图9—1a所描述的软件体系结构模型是无法实现的，因为基本的程序设计语言机制中并没有能够实现数据流传输的手段，而且split、upper、lower和merge四个功能在执行中的先后顺序和衔接也需要规范，因为程序设计语言支持的是顺序结构，不是并行结构。
所以，图9—1a的抽象模型需要被转换为图9—1b的实现模型。
四个过滤器被实现为split、upper、lower和merge四个功能模块。
四个连接件的实现则与其抽象规格之间有较大的差别，它们都被实现为config和i／o library两个模块，其中config模块帮助各功能模块定位自己的输入和输出字符流，i／o library模块帮助各功能模块读和写字符流。
main模块负责控制各模块的执行顺序。

<<软件工程与计算>>

编辑推荐

<<软件工程与计算>>

版权说明