21世纪软件工程专业规划教材

软件系统分析与设计实训教程

张家浩　编著

清华大学出版社

21世纪软件工程专业规划教材

软件系统分析与设计实训教程

张家浩　编著

清华大学出版社
北京

内 容 简 介

本教程是与计算机软件工程专业学生“软件体系结构”课程配套的实训教材，包括三个各为3～5周的实训课程。

全书共9章，分别是：第1章　软件架构设计实训课程导论，第2章　软件的逻辑与思维训练，第3章　基于Arduino的前端开发，第4章　基于树莓派的开发与集成，第5章　物联网服务器的开发与集成，第6章　STKUI的总体架构分析，第7章　STKUI的基本功能分析与二次开发，第8章　STKUI的扩展功能分析与二次开发，第9章　面向服务的STKUI架构再造。其中第2章、第3～5章、第6～9章分别为三个完整的实训课程，因为篇幅关系，分在各章中。

三个实训课程在软件架构知识、学生认知能力、系统与项目开发规模与综合体验的复杂程度上，是三个递进、迭代层次，目标分别是系统逻辑思维训练、小系统开发与集成实训、大型“企业级”应用系统分析与二次开发实训。可根据学生情况和课程需要，分别或持续组织教学实施。

由于是实训课程教材，因此，教程仅仅在需要的地方，简单回顾软件架构设计有关的相关知识点和内容，全书重点是围绕三个实训的若干个项目，让读者从项目开发实践中体会软件架构设计的深度和广度。

教程配有全部PPT和项目源代码，方便老师和学生使用。本书主要用作软件工程相关专业的“软件体系结构”课程的配套实训课程，也可作为其他相关专业的教学用书，或作为从事软件开发的科技人员的参考书、培训教材等。

图书在版编目(CIP)数据

软件系统分析与设计实训教程/张家浩编著．--北京：清华大学出版社，2016
21世纪软件工程专业规划教材
ISBN 978-7-302-43563-1

Ⅰ．①软…　Ⅱ．①张…　Ⅲ．①软件工程－系统分析－高等学校－教材　②软件设计－高等学校－教材　Ⅳ．①TP311.5

中国版本图书馆CIP数据核字(2016)第081946号

责任编辑：张　玥　薛　阳
封面设计：常雪影
责任校对：焦丽丽
责任印制：何　芊

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦A座　　**邮　　编**：100084
社 总 机：010-62770175　　**邮　　购**：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
课 件 下 载：http://www.tup.com.cn，010-62795954

印 装 者：三河市春园印刷有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张**：28　　**字　　数**：647千字
版　　次：2016年8月第1版　　**印　　次**：2016年8月第1次印刷
印　　数：1～2000
定　　价：59.50元

产品编号：069227-01

前言

PREFACE

作者自2011年离开学校，并未曾离开讲台，甚至还曾又回到原来的学校、站在曾经的讲台上，只是换了一个“身份”，真是颇具中国特色、传奇意味。

在培训机构任职，从985学校到高职，有机会接触不同的学校、更多类型的学生，作者发现，现在学生的状况，真的不是用“堪忧”两字可以轻松描述的。在作者讲授“软件架构设计与实践”课程的时候，大三的学生，编程和项目开发的基础之差，连VS 2010的IDE菜单按钮是干什么的都不知道；而刚学过C语言的大一学生，连在TC 2.0环境下，需要将C语言源程序不单要编译成.obj文件，还需要再链接成.exe文件后（也可以选择直接生成.exe），才能运行都不知道。总是问老师，我的代码为什么不能运行（学生只完成了生成.obj文件一步）。这说明，这个学生从来没有自己完成过一个完整的源程序编写、编译和运行过程。也有的老师教Java，不要求学生用Eclipse/MyEclipse，而是用记事本写源代码。虽然也可以用命令行进行Java编译，但是，不使用Eclipse/MyEclipse的IDE，难道代码不用调试吗？难怪学生看到老师用A作为变量名，自己用B的时候，觉得可能自己用错了。上述这些现象，可能是作者遇到的个别极端情况，但这难道可以简单地批评说，是某个学生的错吗？

本教程是作者近年任职培训机构，担任实训课程教学的总结，也是软件工程、软件架构设计课程实训产品的呈现。有不少老师，用了作者的《软件架构实践教程》后来邮件探讨，如何在大三学生“基础很差”的情况下，上好架构设计这门课。我在该书前言部分的第一句话就讲，上这门课是很吃力、并且是“吃力不讨好”的。因为基础差的根本原因不是这门课难（软件工程专业后端的课程，一定是比编程要难，任何专业的课程都类似），而是学生从一年级开始，就缺乏编程、系统、项目的扎实训练。一本高校情况好一点，二本高校以下难度很大。但是，二本以下也是软件工程专业，也不能放弃。所以，需要从大一开始，用实训的方式，进行补充和强化训练。这就是编写本教程的基本出发点。

本教程原设想是设计为4个实训阶段：逻辑思维训练、小系统开发与集成、企业级系统分析与二次开发，以及软件工程过程实训，分别安排在大一至大四或延伸到研究生的4个假期实训阶段。由于这样的实训课程的时间安排跨学年、跨课程，因此，不可能跟现阶段教学体系和课程计划中的某一门课程直接挂钩——归于某门课下。所以，目前本课程的名称被套在“软件系统分析与设计实训教程”的课程名称之下。为了减轻学生的负担，本书删去软件工程过程实训内容（可参考作者2011年清华大学出版社出版的《软件工程项目实训教程——基于微软VSTS》一书），并主要围绕软件架构设计的主题进行描述。

实整，改变了4个阶段实训、分阶段、综合一体化地培养软件工程专业学生系统、项目、软件工程过程能力的大目标，将其缩小到“软件架构设计”这个软件工程过程的具体目标上，从而降低了本教程原设想的高度和作用域。虽然软件架构在软件工程过程中具有一定的“统领”作用，但毕竟只是角色之一，实在遗憾。

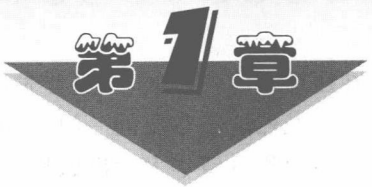
有关课程的目的、内容和教学安排等，在教程的第1章中介绍，不再赘述。

张家浩

2015年12月30日于南京百合果园

目 录

CONTENTS



软件架构设计实训课程导论

本章导读

实训课程与通常课堂教学课程的一个重要区别是：实训课程本身应该被看作是一个教学“项目”，从项目管理要求出发，既然是项目，就必须要有明确的目标；有尽可能细致的过程规划和设计；在项目实施过程中，有可检查的里程碑定义，最终有交付成果的验收方法和评价标准。以项目的方式实施实训课程教学，是本书贯彻始终的基本教学方法，请从第1章开始，就适应这种“教”与“学”的方法。

因此，本章首先介绍并描述“软件系统分析与设计”实训课程项目的项目目标等相关内容。特别需要强调的是：这里所讨论的项目，不是课程中老师分析、学生完成的软件开发项目，虽然这是本课程所必不可少的。本章所讨论的重点，是这次实训课程本身，是本次课程的目标选择、内容安排、过程设计、交付结果验收以及与其他课程的相关关系等内容。

承担本课程教学任务的老师，当然应该首先仔细阅读并理解本章的内容，这是“实训课程项目”总体设计和规划的“教学指导书”。而且，也需要让学生知道和了解，因为这是需要老师和学生一起参加、共同完成的“项目”。

1.1 实训课程的培养目标

软件系统的分析与设计技术，在软件企业而言，是在软件产品设计、系统构建、项目开发、过程控制、系统维护等各项工作中，起主导作用的关键技术。微软的比尔·盖茨很多年前有一句著名的话，大意是，自己在微软可以什么位置都不要（例如董事长、CEO等），但唯一要做的，就是做微软产品的架构师。

在软件企业，架构师为什么那么重要，为什么架构师可以拿高薪，Robert Walters（华德士人才咨询有限公司，英国一家著名的猎头公司）2014—2015全球薪资调查报告（http://www.robertwalters.co.uk/career-advice/salary-survey.html）显示，就算在金融工程这样“烧钱”的行业，企业级应用架构师的年薪也是很高的，甚至高于高级金融分析师，并仅次于总监等高管。

在学校，“软件系统分析与设计”是软件工程专业的一门专业主干课程，是每个软件工程专业学生的必修课。但是，由于种种原因，这门课与其他软件工程高端课程一样，成为软件专业的“政治课”。老师难上、学生难学，最后的结局，必然成为课程体系中的“鸡肋”，

在压缩课时的时候，难逃首先被压缩的结局。

既然这么重要的一门课，为什么难上、难学？原因有很多，本书作者在《软件架构设计实践教程》(清华大学出版社 2014 年 8 月出版)的第 1 章中已有分析，不再赘述。

要很好地解决这个问题，作为一名教师，我自己能做的事情，就是在教学过程中，不断探索、不断挖掘，积累更多的教学资源，采用更合适的教学方法，并把这些成果提供出来，供大家分享，大家一起共同提升这门课的教学水平和教学质量，最终真正让学生受益。

本课程以《软件架构设计实践教程》一书作为理论基础和相关知识来源，而本课程则重点突出基于系统分析和设计理论、技术和方法的项目化的实战体验。

不是学过这门课的所有学生，未来都能够成为架构师，除了能力之外，还有环境、条件、机遇等多种“非技术”因素。但是，如果没有成为架构师的潜质，当环境、条件、机遇出现的时候，你一定不可能成为一名架构师，这就是本课程(包括所有软件工程高端课程)的所谓“启蒙”作用——机会只为那些有能力、有潜质的人准备。

1.1.1 架构师是软件开发的“老兵”

与管理岗位不同，在很多软件企业，可能并没有非常严格、规范的软件架构师的岗位设置和岗位描述，可能实质干的是架构师的活、发挥着架构师的作用，却并不曾被公司或项目组正式任命为一名架构师。

在组织行为学中，架构师、项目经理一类“人物”，具有职位和角色双重身份。前者具有明确的职权和位置，可以直接支配资源，并发布命令，是名词意义上的“领导”。而后者主要的作用是影响、带动、感染、示范、沟通、协调，是动词意义上的“领导”。在微软举办的高校教师软件架构师培训班上，有老师提问：架构师不是“领导”，架构纪律如何贯彻执行？之所以会有这样的问题，就是没有搞清楚两种“领导”在软件企业中的角色差异，以及各自是如何发挥各自的作用的。而后者的角色，在高校环境中，现在还真的不好找。因为正常高校的院长、系主任其实就应该是后者的角色，但现在已经完全“异化”为前者了。

简单来说，要想成为一名新的架构师，除了首先需要具有架构师的知识与技能之外，在软件企业，往往架构师是先“上路”(先实际上成为负责架构设计的那个人)、后“拿证”(后被指派、任命为某某系统的架构师)的。而且，与项目经理一样，你今天是某某系统的架构师，明天到另一个系统项目上，可能未必就是架构师。因为后一系统可能更大、更复杂，或者可能是一个你并不熟悉的系统。所以，架构师是靠自身的能力活着的。

为什么架构师一定会有一个“无证驾驶”的阶段？

首先，与“学车”相比，架构师岗位是没有“驾校”的，在学校学习的那些内容，最多也只能算是在“教练场”开车。一名刚从学校毕业的学生，来到软件企业，必定是从“码工”做起，那么，一名新的架构师是从何而来的？只能从实战战场“拼杀”而来。所以，架构师是一个要求“实战”能力很强的岗位，是一个要求战场经验、生存能力很强的角色，当然，更不要讲这些实战体验，培养了你的眼光、大局、策划、布局和综合能力。一个资深的“码工”并不等于“架构师”。

所以，架构师的“驾照”实际上是公司、同事对架构师的“追认”，一个好的架构师，当他规划、设计出完美的架构蓝图，并付诸实施之后，不知道过了多少年，后来的人为了升级、

维护旧系统再次回顾和浏览架构设计蓝图和设计文档的时候，可能会禁不住由衷赞叹前任架构师为3～5年，甚至更久远的作者留下一个松耦合的系统模块结构划分、清晰的关联、简洁的接口，预留下充分的扩展空间、灵活的控制手段，省去后来者在系统扩展和维护中可能遇到的分析、定位、修改、重写，甚至大幅调整的麻烦和烦恼；也可能让后来人拍桌子大骂，前面的架构师害人不浅。

在战场上，我可能宁愿被“老兵”骂而活命，也不愿听从一个“书呆子”长官的瞎指挥而送命。在职场也有点类似，架构师就是这样的“老兵”。

1.1.2 架构师的知识与能力积累

老兵是身经百战、在枪林弹雨中摸爬滚打锤炼出来的，软件开发的老兵是不是就是代码写得多，项目做得多？答案显然是否定的。不然，上过战场，没有被打死的战士，最后人人都成为了将军。软件企业也是这样。

从战士成长为将军，还需要“领导”才能。这就是软件架构师所需要、码工则不具备的与架构设计有关的思维和基本技能。

从架构的作用和架构设计的过程，可以归纳出架构师需要具备以下4大关键技能。

(1) 业务背景的认知能力：大型软件系统开发，都是针对领域的。架构师对特定业务领域的业务背景的认知、关键业务需求的理解、核心业务发展的走向，都需要具有深刻的认知和理解能力。这个能力的养成，需要在该领域耕耘多年，才能够达到一定的理解深度。设想，如果没有这方面的理解，他所设计的系统架构，且不要说如何能够充分满足用户的需求，更谈不上经历若干年后，可以完美地应对系统的迭代、扩展开发或必不可少的修改与维护。而现代软件开发更朝向基于专业领域框架的迭代开发方向发展，业务逻辑的抽象层次越来越高，业务实体则已经被“封装”到框架中，理解业务已经大部分变成理解框架。

(2) 架构设计与搭建能力：这个能力是架构师的基本功。从架构蓝图、到设计方案，再到架构实现，甚至架构维护，要求架构师像开发软件代码一样，将架构从设想变为现实。与编写软件代码不同，架构师的工作，很大一部分不是由他自己完成的，而是由他提出方案、指挥别人实现的。从这一点开始，他已经不是一个士兵，而是“领导”了。难就难在，一个没有“头衔”的领导如何“服众”。所以，他必然要经历从起初的“压不住”，到“心服口服”的成长过程。

(3) 架构关键需求的把握能力：关键需求是指与架构设计有关的那些需求，它们通常不会出现在用户的需求文档中，甚至在系统交付之前，用户从来都没有提出过。例如，软件系统的灵活性、可扩展性、实时性、并发性、鲁棒性等。有经验的架构师早早就会“预见”到，这些需求的实现，最终是“逃不过”去的“一关”。这一关是什么、在哪里？如何在系统架构设计的蓝图阶段就考虑好解决方案，这个能力即是“远见”，更是“教训和经验”的积累。

(4) 架构设计的组织实施管理能力：架构的规划和设计，不像代码编写，自己调试成功以后，就可以交给别人使用。架构设计需要架构师完成规划和设计后，带领开发团队，共同去实现架构蓝图和设计方案。这包括：架构设计需要让开发团队理解、执行设计方