

高等院校信息类专业校企衔接创新实践系列教材

软件工程 导论

晏峰 著



中南大学出版社

www.csupress.com.cn

(277)

软件工程导论

晏 峰 著



中南大学出版社
www.csupress.com.cn

图书在版编目(CIP)数据

软件工程导论/晏峰著. —长沙:中南大学出版社,2016. 12
ISBN 978 - 7 - 5487 - 2290 - 8

I. 软... II. 晏... III. 软件工程 IV. TP311.5

中国版本图书馆 CIP 数据核字(2016)第 127813 号

软件工程导论

晏 峰 著

责任编辑 谢贵良

责任印制 易建国

出版发行 中南大学出版社

社址:长沙市麓山南路

邮编:410083

发行科电话:0731-88876770

传真:0731-88710482

印 装 长沙印通印刷有限公司

开 本 787×1092 1/16 印张 14 字数 345 千字

版 次 2016 年 12 月第 1 版 印次 2016 年 12 月第 1 次印刷

书 号 ISBN 978 - 7 - 5487 - 2290 - 8

定 价 35.00 元

图书出现印装问题,请与经销商调换

前 言

一直以来想写一本书，用作软件工程专业学生入门的参考书，希望本书能够让学生在开始时就能够明确学习的方向，能够理解软件项目并不是简单的编程，而是涉及到计算机科学、软件工程、系统工程学、管理学等多领域知识的集合体。同时作为一本入门的参考书，重点应该是介绍软件项目实际操作过程的方法论，而不是简单的理论堆砌或是大量的数学推导，这样做也许在学术上并不严谨，但是作为入门级参考书却能够降低学习的门槛。本书也不想写得太厚，这样在内容上不可避免地有所取舍。实际上本书的每个章节都可以对应到一本教材，所以本书仅仅只是入门，希望能够起到抛砖引玉的效果，引导学生主动地去看一些其他的资料。

本书可以分成三部分：第一部分也就是第1章，阐述了软件工程师的职业发展、素质要求和成长途径；第二部分从第2章开始到第5章，从软件开发模型开始，介绍了软件项目的系统分析方法和设计方法；第三部分包括第6章和第7章，重点介绍了软件质量管理和项目管理。这种章节的布局同样是希望学生能够在宏观上了解一个软件项目到底是如何开展的，而不仅仅是把软件开发就当成编程。

本书由晏峰执笔，吴章勇、任青松负责审校工作。在撰写过程中参考了百度百科、百度文库、维基百科、MBA文库等网络资料，同时也参考了计算机科学与技术、软件工程专业的部分教材，对此，深表谢意。

本书可以作为软件工程专业专业导论教材，也可以作为相关专业的参考书使用。

编 者

2016年11月30日

目 录

第 1 章 软件工程师的素养和成长	(1)
1.1 软件工程师职业规划	(1)
1.2 软件工程师职业能力	(3)
1.3 软件工程师的成长	(9)
第 2 章 软件开发模型	(18)
2.1 软件开发模型	(18)
2.2 敏捷开发	(22)
2.3 实践指导	(27)
第 3 章 系统分析与设计方法	(29)
3.1 结构化分析	(29)
3.2 结构化设计	(34)
3.3 面向对象分析	(38)
3.4 面向对象设计	(54)
3.5 数据库设计	(56)
第 4 章 软件体系结构(SA)	(65)
4.1 建模方法	(65)
4.2 常见软件体系结构风格	(66)
4.3 软件体系结构设计过程	(72)
4.4 实践指导	(73)
第 5 章 设计模式	(75)
5.1 面向对象的设计原则	(75)
5.2 创建型设计模式	(77)
5.3 结构型设计模式	(84)
5.4 行为型设计模式	(95)
5.5 实践指导	(112)

第6章 软件质量保证	(114)
6.1 质量与全面质量管理	(114)
6.2 软件质量模型	(118)
6.3 软件质量过程管理	(126)
6.4 软件测试	(138)
6.5 实践指导	(149)
第7章 项目管理	(151)
7.1 项目管理的基本概念	(151)
7.2 进度管理	(155)
7.3 成本管理	(165)
7.4 风险管理	(178)
7.5 团队建设	(185)
7.6 实践指导	(187)
附 录	(189)
一、系统流程图图例	(189)
二、数据流图图例	(190)
三、程序流程图图例	(191)
四、UML 语言	(192)
五、编程规范	(199)
六、GB/T 8567—2006 软件需求规格说明	(204)
七、GB/T 8567—2006 软件测试计划	(211)
参考文献	(216)

第1章 软件工程师的素养和成长

1.1 软件工程师职业规划

计算机硬件技术的发展符合“摩尔定律”所述：当价格不变时，集成电路上可容纳的晶体管数目，约每隔 18 个月便会增加一倍，性能也将提升一倍。软件技术的发展同样符合此规律，每隔 1~2 年便有新的概念和新的技术出现，对软件开发者而言，所面临的的是一个知识和技能需要不断更新的挑战。

在软件行业流传着这样的说法：“程序员只能做到 35 岁”、“软件行业技术更新太快，学到的技术很快过时”、“在中国做技术没有前途，只有去做管理、销售……”这些说法当中第一条并非完全没有道理。35 岁是人类精力和体力发展的顶峰，过了这个年龄段，是否能够继续进行高强度的软件开发确实值得商榷。但后面两条就有失偏颇，技术发展是一个不断积累的过程，类似于爱因斯坦相对论这样划时代的发现在软件行业未必不会出现，但古典物理学在人们日常生活当中还是随处可见，在 IT 技术的发展中新的技术和新的概念总是与之前的技术和概念之间存在千丝万缕的关系，可能一项技术会过时，但是通过这项技术所积累的经验不会过时，所需要做的是修正已有经验中过时的部分。与技术提升相对应的是程序员自身价值和定位的变化，最开始的工作岗位并不一定是最合适的岗位。在整个职业生涯过程中，人们将通过不断的换岗、晋升和跳槽来找到适合自己的位置，实现自我的价值。

软件行业的岗位如果细分不下百种，简单的可以分为开发工程师（程序员）、测试工程师、售前工程师、实施工程师、售后工程师、运维工程师、系统分析师、系统架构师、开发经理、项目经理等，这些岗位又可以根据方向和职位的不同进行进一步划分，比如开发工程师（程序员）可以按照开发所使用的语言种类进行进一步划分，如 C 程序员、C++ 程序员、Java 程序员、C#程序员、PHP 程序员等（国外已经很少做这个划分）。每个语言种类都有在软件行业中相应的开发领域，差别在于需求量的大小和薪资的高低而已。了解软件行业的岗位分类有助于软件开发者进行职业生涯规划。

图 1-1 展示了一个软件编程爱好者的职业发展之路。在编程爱好者阶段还可以纠结于 C++ 还是 Java 的语言选择，但一旦选定职业化路径后，语言的偏好将让位给企业的需要和技术的发展，此时他成为一名开发工程师。之后有两条路可以选择，一条路是纯技术路线，成为资深专家→系统架构师；另一条路是技术+管理的路线，成为开发经理→项目经理。实际上随着职位的升高，管理素质的要求是在不断提高的。换言之就是他都将从团队底层的执行者逐步上升到团队的管理者或决策者。在这个过程中他将不断积累软件开发经验、项目管理经验和团队管理经验，路径的最后一步是成为 CTO。这一步取决于他的创新能力和机遇。

01

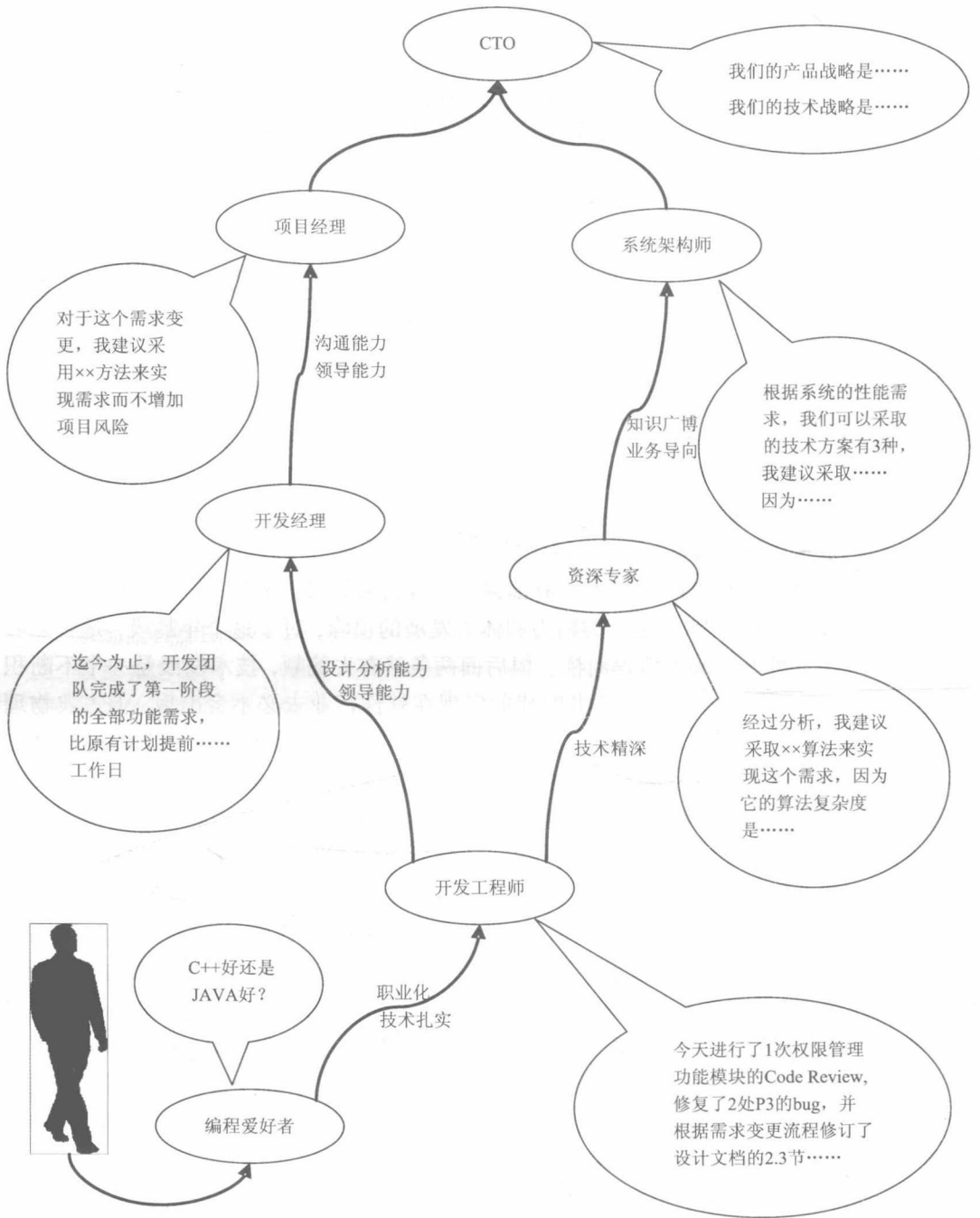


图 1-1 软件开发者职业之路

从另外一个角度来考察图 1-1, 可以发现随着职位的提升, 编程的代码量是递减的, 在最底层开发工程师这一级, 可能需要花费 80% 的时间在代码编写上, 但到了 CTO 这一级别, 代码可能仅仅会成为一种休闲的方式; 与代码量的递减相反的是系统性思考工作量的递增,

在开发工程师这一级仅仅需要思考如何去完成需求说明书所提出的功能性要求，而在 CTO 这一级需要考虑的是企业战略如何与信息技术的结合，这是一个思想从微观具体到宏观抽象的转变过程。

图 1-1 还隐含了一个命题，请注意从资深专家到系统架构师的路径上的这段文字——“知识广博、业务导向”。这段文字所希望表达的含义是在整个职业生涯的发展路径当中，需要充分考虑“跨专业领域”，跨专业领域所涵盖的范围非常广，所有非计算机学科专业或软件工程学科专业都属于此范畴。软件开发简单的可以分为计算机系统软件开发（如操作系统、数据库管理系统等）和应用软件系统开发（如财务管理系统、股票分析系统、游戏等），对应用软件系统而言，开发人员除了需要了解计算机软硬件方面的知识以外，还需要了解应用软件所处行业的专业知识、业务流程，只有这样才能够开发出符合“用户体验”的系统，才能够最终被用户所接受。

除了针对具体应用领域的知识以外，在软件工程师职业成长过程中，一些体现综合素质的知识也必须掌握，就像从开发经理走向项目经理，此时他不仅需要软件开发的知识和项目的知识，还需要具备基本的会计和财务分析能力，只有这样他才能够去控制项目的成本，获取更多的资源。

1.2 软件工程师职业能力

职业能力 (Occupational Ability) 是人们从事某种职业的多种能力的综合。职业能力所涵盖的范围非常广，因此在实践当中更关注的是其关键能力或职业核心能力。软件工程师的核心职业能力可以概括为：专业能力、团队协作能力、学习与创新能力、计划与执行能力、心理素质。上述能力指标又可以具体划分为多个二级能力指标，具体见表 1-1 软件工程师的核心职业能力指标。

表 1-1 软件工程师的核心职业能力指标

一级指标		二级指标	
I	专业能力	I-1	数学能力
		I-2	外语能力
		I-3	系统分析与建模能力
		I-4	系统编程能力
		I-5	系统测试能力
		I-6	系统部署能力
		I-7	文档写作能力

续表 1-1

一级指标		二级指标	
II	团队协作能力	II - 1	团队协作精神
		II - 2	沟通和表达能力
		II - 3	人际交往能力
		II - 4	绩效管理能力
III	学习与创新能力	III - 1	学习态度
		III - 2	学习方法
		III - 3	理解记忆能力
		III - 4	信息检索能力
		III - 5	发现问题能力
		III - 6	批判思维能力
		III - 7	直觉思维能力
IV	计划与执行能力	IV - 1	组织管理能力
		IV - 2	成本控制能力
		IV - 3	时间管理能力
		IV - 4	质量管理能力
		IV - 5	风险防控能力
V	心理素质	V - 1	成就导向
		V - 2	自我调节能力

上述指标在进行职业能力评测的时候,需要根据软件工程师不同的职位或岗位要求进行权重再分配,也可以增减其中的一些指标。一级指标中的专业能力和团队协作能力是关键性指标,心理素质指标在大企业招聘过程中会成为性格测试的一部分。此外上述指标都可以通过恰当的训练进行提升。

1.2.1 专业能力

专业能力指标反映了软件工程师执业的基本任职要求,可分解为数学能力、外语能力、系统分析与建模能力、系统编程能力、系统测试能力、系统部署能力和文档写作能力等7个二级指标。

• 数学能力

在魏德林关于数学能力一书中指出:“数学能力是理解数学的(以及类似的)问题、符号、方法和证明本质的能力,是学会它们、在记忆中保持和再现它们的能力;是把它们同其他问题、符号、方法和证明结合起来的能力;也是在解数学的(或类似的)课题时应用它们的能力。”数学能力可以概括为抽象思维能力、逻辑推理与判断能力、空间想象能力、数学建模能力、数学运算能力、数据处理与数值计算能力、数学语言与符号表达能力等。对软件开发工

程师而言数学建模能力、空间想象能力、抽象思维能力、逻辑推理和判断能力是必须具备的基本素质。

• 外语能力

在软件开发领域,最新技术文献和专业技术文档一般都是采用英文写作。英语在软件外包和国际交流领域也是通用语言,因此作为软件工程师最少需要掌握一门外语(英语),并具备基本的听说读写能力,除此以外还应了解相应英文公文信函的书写格式。

• 系统分析与建模能力

系统分析是软件开发的第一步,在与用户沟通的基础上,提出用户需求模型,规划设计系统的实现框架和软件的逻辑结构,并选择合适的技术用于开发,因此软件工程师应熟悉软件开发的相关技术(如软件开发框架、开发语言、数据库技术等)。掌握常用系统建模工具的使用,并使用建模工具完成系统分析过程。

• 系统编程能力

系统编程是在系统分析或概要设计的基础上,对软件项目进行详细设计,并完成代码的编写工作,系统编程能力是软件工程师的基本职业素质。对软件工程师而言要求理解操作系统的概念,熟悉数据库操作,熟悉软件开发过程,熟练掌握软件开发模式和开发框架。最少掌握2门或以上常用开发语言。

• 系统测试能力

系统测试在软件生命周期中占有非常重要的地位,是软件质量的保证之一。软件测试分为程序调试、单元测试、集成测试和试运行几个步骤。软件工程师不仅需要有系统分析和编程的能力,同时也需要具备软件测试的能力,具体包括了解软件测试理论,掌握软件测试方法,能够熟练运用软件测试工具,能够编制软件测试计划和完成软件测试过程。

• 系统部署能力

最终完成的软件系统需要在最终用户环境中运行,软件开发很多时候会被要求参与最后的系统部署和试运行阶段,因此软件工程师对软件系统所运行环境,包括软件系统本身的结构、所运行的操作系统、数据库管理系统以及网络应有一定的了解。对操作系统和数据库管理系统应当掌握系统安装及基本参数配置的基本方法。对软件工程师更高的要求是掌握上述系统的调优方法。

• 文档写作能力

文档写作一直是被软件工程师所忽略的一个能力,但在正规项目开发流程当中,特别是在大型项目开发过程中,文档可能会占据软件工程师近三分之一的工作时间。文档是项目沟通交流和开发实施的基本依据,因此对软件工程师而言了解软件项目文档的分类,熟悉软件项目文档撰写的要求,掌握文档中各类文档、图形符号,能够独立编写完成软件项目文档是其基本职业能力之一。

1.2.2 团队协作能力

软件开发已经从“单兵模式”转向“兵团模式”。团队协作的范围可以从小组协作、部门内协作、组织内协助一直延伸到组织间协作,团队协作能力是软件工程师执业的一个基本条件,团队协作能力指标可以分解为团队协作精神、沟通和表达能力、人际交往能力、绩效管理能力等4个二级指标。

• 团队协作精神

团队协作精神包括大局意识、协作精神和服务精神。团队协作精神的核心是协同协作，要求在团队当中的个体利益和整体利益统一。应当注意的是团队协作并不是抹杀个性，一个优秀的团队是在充分发扬团队成员向善个性的基础上发展起来的，团队成员之间优势互补，各司其职。

• 沟通和表达能力

在团队内部以及与团队外部为完成共同的工作目标，需要进行良好的协作，良好协作的基础是良好的沟通，良好的沟通首先需要有正确的心态，其次需要掌握正确的沟通方法和沟通技巧，最后需要因时、因地、因人进行正确的表达，能够利用语言、肢体以及其他媒介无歧义的表达所需要传递的思想和意识，不怯场。在沟通过程中必要的让步有时是必需的，也就是通常所说的“以退为进”。

• 人际交往能力

人际交往能力就是指人际关系处理能力，可以分为人际感受能力、人事记忆力、人际理解力、人际想像力、风度和表达力、合作能力与协调能力等，简单地说就是能够妥善处理组织内外关系，包括与周围环境建立广泛联系和对外界信息的吸收、转化，以及正确处理上下左右关系，使自己能够成为一个受欢迎的人。需要注意的是好的人际关系并不代表没有原则，做“滥好人”。

• 绩效管理能力

目前团队考核一般都是采用绩效考核的方式，团队绩效不能简单认为是个人绩效的总和，按照“木桶理论”，团队绩效的优劣不仅仅取决于团队成员的平均绩效，而可能是团队中最差成员的表现，因此要在团队中立足，需要不断提高或者提升个人的绩效水平。绩效管理分为自我绩效管理和团队绩效管理两个方面，个人绩效管理主要指是否能够按时、按质、按量完成团队所分派的任务指标；团队绩效管理则包括指标制定、分解、落实、检查和激励等方面，作为团队项目负责人更需要关注的是团队绩效管理水平的提高。

1.2.3 学习与创新能力

软件行业的特征是不断发展、不断创新，软件工程师必须适应这种特征，只有通过不断学习和创新才能够延长软件工程师的职业生涯里程，才能够为软件工程师向更高层次发展打下基础，学习与创新能力指标可以分解为学习态度、学习方法、理解记忆能力、信息检索能力、发现问题能力、批判思维能力、直觉思维能力等7个二级指标。

• 学习态度

学习态度决定了学习的成效，事实上也决定了工作的成效，每个人都有惰性，这是需要客观面对的。软件工程师从行业发展的需要，必须要有积极向上的学习态度，可以从以下几方面分析：是否有接受新事物的愿望？是否有学习新事物的冲动？是否有良好的时间管理观念？是否有付诸实施的勇气？是否有坚持不懈的毅力？需要经常问问自己，为什么要读书？为什么要学习？我与他人的差距到底在哪里？

• 学习方法

明确了学习态度，之后就需要一个好的学习方法，好的方法事半功倍，差的方法事倍功半，没有那种“放之四海而皆准”的学习方法，适合别人的方法未必就适合自己，就像现在的

“成功学”教程，别人按那个方法做可能能够成功，而自己来用就未必能行。好的学习方法是目标导向，要根据自身的特点和不同的目标制定不同的学习方法和学习计划，对软件工程师而言只能提几点建议：“学中做、做中学”、学会使用工具书(包括网络信息检索)、学会阅读的方法(快速阅读和精读)、学会从前人的项目中吸取经验、学会归纳总结和反思、学会知识归零。

• 理解记忆能力

记忆能力是人的天赋，记忆力的好坏对学习、工作的影响较大，但是记忆力是可以训练和通过有效的方法进行弥补的，最简单的就是在日常工作当中的工作笔记本，重要的东西可以先用笔记下来，通过之后的反复阅读和背诵可以强化记忆。对软件工程师而言理解后的记忆比强记(死记硬背)更重要，特别是对软件专业领域的知识，原因很简单，软件行业是应用性行业，是直接面向问题的，如果不能理解其中的精髓，就如同赵括的“纸上谈兵”，而且理解后的记忆效果是远优于强记的效果。如果理解后再进行实际项目的运用，这样会更好地加深记忆。

• 信息检索能力

不论是学习、做项目、做研究还是要搞创新，都要进行信息检索(狭义上可以称为文献检索)，通过信息检索可以获得知识参考、了解发展动态和可能出现的问题、提供有用的思路和方法。软件工程师尤其需要具备信息检索的能力，包括了解信息检索的方式、方法和渠道，掌握信息检索工具的使用，检索结果的分析、归纳、整理和应用等。

• 发现问题能力

发现问题是创新的第一步，实际上也是软件项目的第一步，是软件项目系统调优的第一步。发现问题源于认真的观察和思考，凡事需要多问一个为什么，有什么可以改进的地方。在观察的过程当中最为忌讳的是存在“事实就是如此”的思想，一旦有了这种思想就不会进一步的去思考“为什么会这样”的问题；发现问题的第二步是记录问题，很多时候可能想到了有问题、有改进之处，但是因为这样或那样的原因无法立即实施改进的步骤，这个时候记录问题就非常重要；发现问题的最后一步是实施改进，实施改进有时候是需要勇气和毅力的，大多数人在这个时候选择了放弃。

• 批判思维能力

思维是人类的本能，当遇到问题时，会不自觉的开始思考，因为知识阅历、思考方式、思考方法的不同，会得到不同的结论。在思考过程中很多时候人们会基于“本能自信”，也就是认定自己的知识和经验是正确的、所了解的信息材料是准确和全面的、所掌握的方法是正确的。但事实上，知识和经验中可能存在错误的或过时的成分，所了解的信息材料中可能有不准确、不全面的成分，所运用的方法中也可能有不恰当或错误的方法，思维过程会受到各种不利因素的干扰。因此，我们的思维活动并不是必然科学合理的，得出的结论不是必然正确的，而是存在产生错误的可能性。因此，需要对我们的思维过程进行审查，发现并排除可能存在的错误，这就需要进行批判性思维。培养批判性思维能力，就是要从对自己思维的“本能自信”，转向对自己思维的“自觉质疑”和“仔细审查”。

• 直觉思维能力

直觉思维也称非逻辑思维，它是一种没有完整的分析过程与逻辑程序，依靠灵感或顿悟迅速理解并作出判断和结论的思维。这是一种直接的领悟性的思维，具有直接性、敏捷性、

01 简缩性、跳跃性等特点，可以认为它是逻辑思维的凝聚或简缩。直觉思维是创造性思维活跃的一种表现，是发明创造的先导，在创造发明的过程中具有重要的地位。很多时候，软件工程师过度强调了系统性和逻辑性思考，而忽视了对直觉性思考的训练，但软件行业的创新需要软件工程师具备直觉思维能力。

1.2.4 计划与执行能力

计划执行是指企业在经营和发展过程中制定战略目标，由企业各组成部分对战略目标进行分解，制定切实可行的行动计划，有效地协调与运用各种资源，执行并达到目标与任务要求的过程。在这个过程中所表现的能力就是计划执行能力，可以分成三个部分，分解制定计划，协调运用资源，执行并达到目标。在软件项目团队中，项目负责人负责计划制定和资源协调，项目成员负责计划执行。计划与执行能力可以分解为：组织管理能力、成本控制能力、时间管理能力、质量管理能力、风险防控能力等6个二级指标。

• 组织管理能力

组织管理能力是指为了有效地实现目标，灵活地运用各种方法，把各种力量合理地组织和有效地协调起来的能力，组织管理能力是一个人的知识、素质等基础条件的外在综合表现，是作为项目团队负责人的必备能力之一。组织管理包括设计组织架构、明确责权关系、制订工作计划、管理工作进度、充分授权、沟通协调内外部资源、资源分配等。

• 成本控制能力

软件项目的失败很多时候是因为成本控制失败，软件项目成本按大类可以分为固定成本和可变成本两类，主要包括人力成本、实施成本、材料成本、其他直接成本以及分摊成本等，成本控制的核心目标是保证各项工作在其费用预算范围之内完成，软件项目成本控制工作从资源计划开始、通过资源计划进行费用预估和成本预算工作，之后在项目执行过程中实施成本控制。成本控制过程要求随时监控成本的执行情况，根据监控情况对成本进行及时有效的调整并告知利益相关方，当成本不可控时，及时终止项目。

• 时间管理能力

时间管理水平的高低很大程度上决定了软件项目的进度执行效率，也决定了软件工程师的学习效率，时间具有无弹性、无积蓄的特征，软件工程师每日工作时间分配到具体任务上的时候是呈现碎片化特征，在固定任务与突发事件处理之间存在着不可调和的矛盾。在具体任务时间分配上软件工程师可以考虑使用ABC分类法或者是6点优先工作制，此类方法同样适用于软件项目进度管理。

• 质量管理能力

质量是产品的生命所在，软件质量的优劣同样是软件项目成败的关键因素之一，在软件项目执行过程中的各阶段产品质量直接影响项目进度和成本控制，软件项目的质量保证并不是仅仅依靠测试来完成的，测试仅仅只是软件产品质量保证的一小部分，软件产品质量包括代码质量、文档质量、沟通质量、实施质量等，因此软件项目的整个生命周期都需要实施全面质量管理，对软件工程师而言首先需要具备全面质量管理的思想，了解和掌握全面质量管理的方法和评估工具的使用，并在软件项目全生命周期中贯彻执行。

• 风险防控能力

软件项目的整个生命周期当中事实上存在有很多风险因素，这些风险因素严重的可以影

响整个软件项目的生存,最直观的比如技术风险、进度控制风险、成本控制风险、质量控制风险、知识产权风险等。风险防控包括事前、事中和事后三个阶段,事前需要充分预估项目可能存在的风险,并提出相应的应对措施,在软件开发瀑布模型中的第一环节可行性分析其中的重要部分就是对项目可能存在的风险进行预估;事中需要及时检查诸如项目进度、成本、质量等风险点,并对已发生的风险提出改正措施,评估后执行;事后需要总结在整个软件项目周期的所发生的风险事件,为下一个项目提前做好准备。

1.2.5 心理素质

心理素质是以生理素质为基础的,在实践活动中通过主体与客体的相互作用,而逐步发展和形成的心理潜能、能量、特点、品质与行为的综合。心理素质涵盖的范围非常广,比如自信心、责任感、坚韧性、主动性等,这里选取了对软件开发工程师核心的两个指标——成就导向和自我调节能力。

• 成就导向

成就导向是指为自己及所管理的组织设立目标、提高工作效率和绩效的动机与愿望。个人希望出色地完成工作,愿意从事具有挑战性的任务。这种人在工作中有强烈地表现自己能力的愿望,不断地为自己设立更高的标准,努力不懈地追求事业上的进步。软件行业是一个充满挑战性的行业,新概念、新技术、新方法层出不穷,对软件工程师而言时时存在不进则退的危机感,只有具备成就导向才能够不断地提升自身素养,在激烈竞争的环境当中生存下来。

• 自我调节能力

自我调节是个体认知发展从不平衡到平衡状态的一种动力机制。自我调节有广义和狭义之分。广义的自我调节,指人们给自己制定行为标准,用自己能够控制的奖赏或惩罚来加强、维护或改变自己行为的过程。狭义的自我调节,实际上指自我强化,即当人们达到了自己制定的标准时,用自己能够控制的奖赏来加强和维持自己的行为的过程。软件工程师所面临的压力非常多,项目进度的压力、新技术的压力、人际关系的压力、社会期望的压力等,面对诸多压力软件工程师必须要有良好的心态和冷静的思考,这些都需要软件工程师通过自我调节来达成。

1.3 软件工程师的成长

软件工程师的成长是一个渐进的过程。在开启软件工程师职业生涯之路时,首先需要明确兴趣和职业的关系。兴趣代表的是一种个人的喜好,可以随着时间的推移而改变,或者因为一件微不足道的小事儿改变,兴趣改变的代价是不显著或者对个人没有太大的影响。职业所代表的是一种专业,随着时间的推移职业的经验是不断累积的,职业改变的代价是显著的,对个人有较大影响的,最少更换一个职业意味着需要重新进行学习与新职业相关的知识和技能。有兴趣不代表一定要从事与兴趣相关的职业,同样的从事与兴趣无关的职业也是可行的,当然从事与兴趣相关的职业时能够获得的事业成就感更强,同样的当职业受到打击的时候其事业挫折感也会更强。

其次需要明确的是职业能力是可以训练的,是可以通过后天的学习、实践来弥补先天不

01 足的,就如沟通和表达能力,可能有的人天生就是表演家,有的人天生就言语木讷,但并非没有办法来进行改变,只要他愿意敞开心扉,愿意接受相关的训练,就如同“士兵突击”中的战士许三多一样,是能够改变自己能力不足的,区别在于不同人的付出和收入比不同而已。

最后需要正确的评估自己的能力,过高的评估会导致不切实际的目标制定,最后的结果是放弃,过低的评估会设立低水平的目标,导致自满情绪的产生,一旦碰到困难后,最后的结果同样是放弃。

在正确的评估自己的能力之后需要制定成长的目标和计划,目标、计划的制订应符合SMART原则(Specific:具体的;Measurable:可以衡量的;Attainable:可以达到的;Relevant:和岗位发展目标相关的;Time-based:明确的截止期限),不建议制订时间跨度过长的成长目标和计划,因为技术和需求的变化在软件行业是非常快速的,过长时间的目标、计划几乎不可能跟上这种变化的速度;同样不建议制订过细的计划,具体细致到每天或者每个小时,这种计划在实施过程中同样是没有意义的,因为你无法预计到每天、每个小时到底会发生什么,你可以做的是确定在当天根据你的目标和计划你准备完成哪些任务。一般目标计划建议以三年为一个周期,在三年时间内再制定年度计划、季度计划和月计划,具体过程如图1-2所示。在每一个计划的终止时间都应该设置检查点,检查是否按预期达成目标和完成计划,对已经达到的总结经验,对未达到的检讨其发生的原因并提出改进措施,之后根据当前阶段所完成的目标情况,修订下一阶段的目标,再制定完成目标的下一阶段工作计划。

在软件工程师成长过程当中了解可能的学习途径、学习内容、掌握正确的学习方法是有必要的。

1.3.1 学习途径

学习途径主要有正规学习、短期进修、研讨会或技术交流和自学四种。每种方式各有其优劣,在具体选择时应根据实际情况进行选择。

- 正规学习

软件工程师可以在正规院校、培训机构参与学习。其特点是时间周期较长,一般在半年以上,强调所学习知识的系统性和完整性,在进入软件行业之前,建议先进行正规学习。

- 短期进修

短期进修一般在半年以内,短的可能只有1周左右。形式有很多种,有实体开班的,也有网络学习的,一般短期进修所关注的是某一个具体问题的解决,比如一项新的技术。短期进修比较适合已经在岗的软件工程师。

- 研讨会或者技术交流

这个时间要比短期进修的时间更短,可能是一个小时或者是一天,研讨会或技术交流可以是针对某个专题展开,也可以是不同技术之间的一个碰撞,通过这种方式所获取的信息量是非常大的,应当记录一些感兴趣的问题,之后再通过其他方式进行进一步学习。

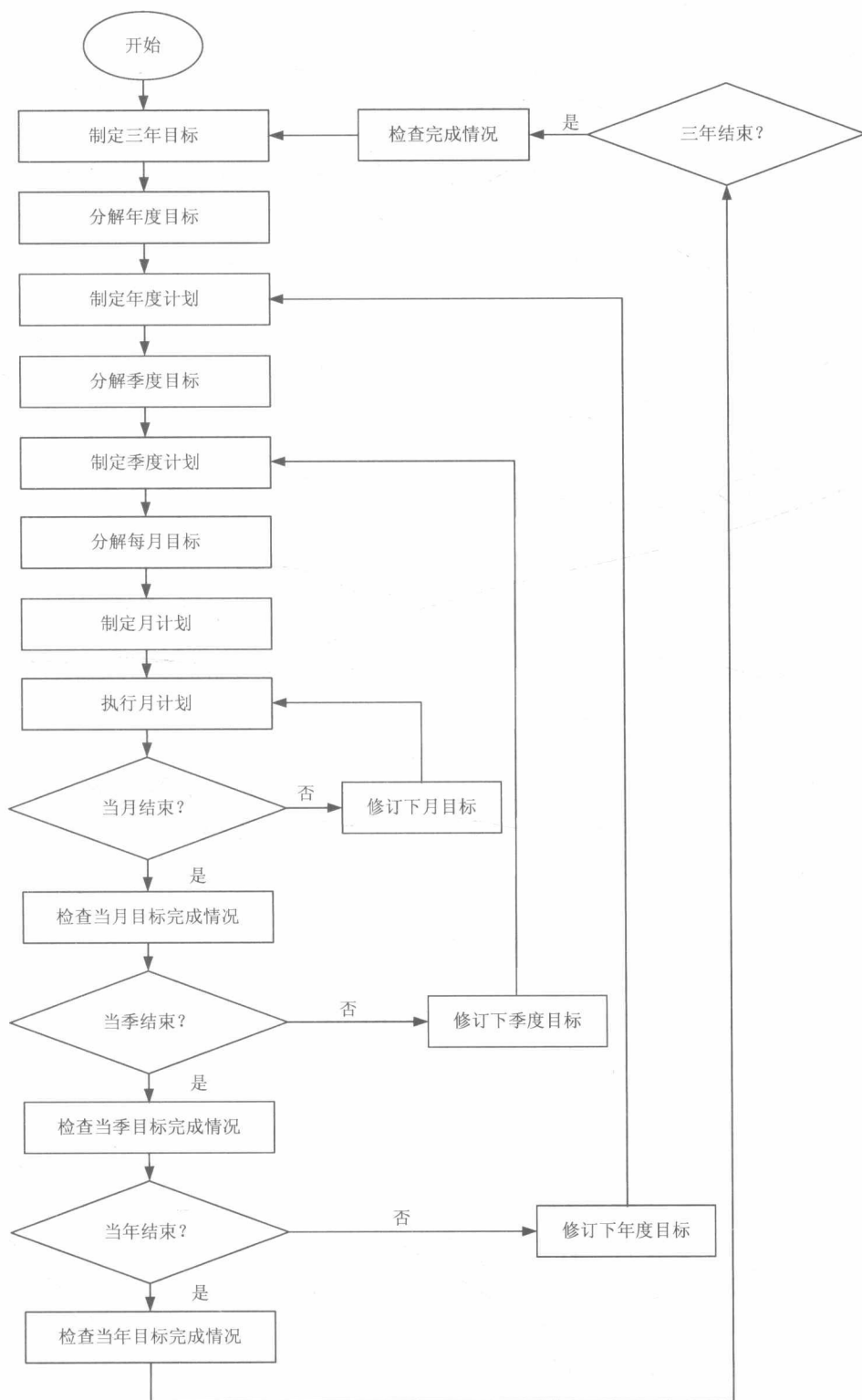


图 1-2 成长目标、计划制订与实施