

勤于思考，丰富经验，
精通技能，
做合格的软件人才。

不是三维

——软件项目的设计、开发与管理

何小朝 著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

不是三维

——软件项目的设计、开发与管理

何小朝 著

电子工业出版社

Publishing House of Electronics Industry

北京•BEIJING

内 容 简 介

本书从软件与三维实物的本质性不同出发研究软件生产方法论。今天，软件生产中的许多传统问题并没有得到质的改善，以“爆炸”形容技术发展的实质究竟是什么？先进管理模式下的管理现状如何？本书首先深入分析以上这些问题，指出软件的本质及其至今不能实现工业化生产的内在原因；接着，从软件设计、测试与管理三个方面，系统地介绍了作者多年实践中总结出来的有效实用方法；最后从软件本质出发，提出了全新的先进软件生产模式与方法——代码驱动生产（Code Driving Production），并指出这将是适合软件本质的新途径。

本书适合于从事软件工程研究的科研人员、管理人员与设计人员阅读，也可作为高等院校与计算机相关专业师生的教学参考读物。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目（CIP）数据

不是三维：软件项目的设计、开发与管理 / 何小朝著. 北京：电子工业出版社，2010.4
ISBN 978-7-121-10484-8

I. 不… II. 何… III. 软件开发—项目管理 IV. TP311.52

中国版本图书馆CIP数据核字（2009）第183285号

责任编辑：高洪霞

特约编辑：顾慧芳

印 刷：北京机工印刷厂

装 订：三河市鹏成印业有限公司

出版发行：电子工业出版社

北京市海淀区万寿路173信箱 邮编100036

开 本：720×1000 1/16 印张：19.25 字数：246.4千字

印 次：2010年4月第1次印刷

印 数：4000册 定价：39.80元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至zltts@phei.com.cn，盗版侵权举报请发邮件至dbqq@phei.com.cn。

服务热线：（010）88258888。

序 一

Preface 1

软件工程和传统意义上的工程有很大不同。软件产品是由人，而不是机器来加工的。在软件的设计中往往必须考虑实现方法，而实现中又会改变设计。这些因素决定了软件的设计，实现和管理不同于传统产品的设计，加工和管理。

本书作者从软件设计方法，实现方法和管理方法几个方面说明了现代软件工程中存在的一些方法性的问题及其原因。作者针对软件工程中的挑战总结了一些实用的方法，并提出了以实现为中心的“代码驱动软件生产”的观点，不同于以设计为中心的传统生产方法。

本书基于作者多年的实践经验和积累，包含了具体的技术内容，对于那些正在从事软件工程设计和管理的技术人员具有实用的参考价值和启发作用。

王敬文

加拿大Platform计算公司产品副总裁

2009年12月

Preface 1

Software engineering and traditional engineering are very different. Software products are engineered by people, not by machinery. The design of the software must be grounded by implementation methods, and the design is often revised during the implementation. These factors determine the differences between a software product and traditional product in its design, manufacturing and management.

The author of the book described the methodology challenges and their root causes in modern software engineering in terms of software design methodology, implementation methodology, and management practice. With these challenges in mind, the author has reviewed some practical approaches to software engineering. Finally, the author proposed an implementation centric approach, called “code driven software production”, which is quite different from the traditional design centric software production approach.

The book is based on many years of hands-on experience by the author with concrete technical content. It inspires thinking and is of practical reference value for technical people who are engaged in software engineering and management.



Jingwen Wang (王敬文)

Vice President, Products

Platform Computing Corporation, Canada

December 2009

Platform公司简介

Platform成立于1992年，是加拿大第四大软件公司。在全球设有包括北京、西安在内的26个研发、销售机构。作为世界知名的网格计算(Grid Computing)软件开发者和全球网格计算的领导者，Platform公司的网格计算产品和解决方案已成为包括GM、IBM、HP、AMD、SUN、SGI、EMC、Fujitsu、NASA、BMW、Toyota、Airbus、Boeing、Pratt & Whitney、JP Morgan、Shell Oil、Motorola、AT&T、Nortel和美国国防部等1 700多家世界顶尖级企业及60%的世界500强企业所使用。Platform还是国际上分布式计算方面的先驱与领军组织，也是“开放网格论坛”(Open Grid Forum)的发起人与组织者。

序 二

Preface 2

无论技术发展多快，与电子、机械产品的规范化生产相比，软件开发的实际状况却一直不尽人意，被称为“黑箱技术”。这本书从软件与三维实物的本质性不同这个全新角度出发，从技术与管理两个层面，深入分析现状，研究软件生产方法论，并给出相应的实用方法与工具介绍。

这里，我想先谈谈软件生产的现状。今天，软件的应用已经覆盖各行各业，几乎无处不在，无论什么领域，一定或多或少地存在着软件设计与开发工作，本书作者称为“软件生产”。应该说，与很多年前相比，软件技术的发展更新速度飞快，成果显著！在开发工具、支撑环境、分析设计方法、软件工程理论与软件管理方法等各个方面都有众多的革新内容。

在实践中不难发现，技术进展虽然使软件开发的效率与质量大大提高，但要使其实现如传统工业那样的标准化与工业化，仍然感到困惑、困难，甚至是无从下手。很多组织虽然不断引进新的技术与方法，但软件生产中的许多传统问题却依然存在并且没有得到质的改善；在软件生产管理中，开发者与管理者的不协调比任何其他行业都明显；在软件生产理论研究中，研究者与实践者的脱节也比任何其他领域都严重。

在软件开发、设计与管理的实践中，人们总是自然地产生对更优方法的需求与渴望，于是对不断出现的新生理论、管理模式、方法与工具

感到兴奋异常，继而学习、使用，却又一次次产生困惑与疑问！这究竟是怎么回事呢？

作者何小朝曾参与过很多项目与产品开发，他本人也接触过一些国际先进的软件组织与著名的资深专家，并与如KDE创始设计师这样的专家一起工作过，有许多经验、体会与感触。作者曾告诉我，在多年的工作与生活中，他深深感到：“大千世界，卧虎藏龙”，平凡世界中隐藏着无数的“大侠”！因而他希望能将自己的体会、观点与同行们共享，希望能与大家在许多方面进行讨论，也希望能为大家提供支持与帮助。

书中，作者分析了经过多年的飞速发展后软件生产依然存在的主要问题，指出软件业飞速发展的本质以及软件生产至今不能实现如机械电子产品那样有效工业化的内在原因，并总结了软件界的两个矛盾：管理者与开发者的矛盾；研究者与实践者的矛盾。在其方法论篇，作者先结合多年经验体系化地介绍了长年实践中总结的实用软件生产方法，其中包括：软件设计方法论、软件测试方法论、软件管理方法论等，旨在对现有的生产实践作出实际的指导与帮助；最后则从软件本质出发，提出了新的现代软件生产模式、理论与方法——代码驱动生产（Code Driving Production），并指出这将是适合软件本质特征的新途径。

本书立意新颖，发人深省，其内容丰富、全面而又实用，我愿意向各位推荐这本书，希望能为广大读者带来帮助与启发。

承继成

北京大学教授，欧亚科学院院士

2009年12月

前言

Preface

20世纪初，伟大的物理学家爱因斯坦提出了具有划时代意义的相对论，其全新的时空理念直到今天都很难被人们以常规的形象思维所理解，而爱因斯坦认为，人们不能正确理解世界本质的根本原因是由于人类的理解能力无法超脱三维空间范围的限制，正如我们无法想象在四维或多维空间的一个球体的形状一样。

现代软件行业的概念、方法、工具出现速度之快是非常惊人的，几乎所有的同行每天都在不停地学习新的东西。笔者相信其中有相当一部分人是期望能从新的方法中找到解决该行业多年需要解决的众多难题的好途径，如设计的规范化问题，管理的有效性问题的，开发的重复性问题等。

然而，笔者也相信，当我的同行们怀着期待的心情对这些新的理论与方法不断学习与尝试后，他们发现——虽然大量可视化开发工具与众多应用支撑环境的出现，使现代应用软件开发基本摆脱了过去那种将最大精力放在界面、通信及大量数据管理的状态，而能集中力量开发自己的核心业务内容；然而，往往在一个具体的项目或具体的企业中，又有众多的原本令人兴奋不已的新方法、新理论，甚至是那些成功者的宝贵经验，却是那样难以真正起到预期的作用，甚至在有些情况下毫无作用。因此，这些可能主要存在于新的设计分析方法与管理方法和具体应用结合等方面。

软件开发工具不断更新，软件工程师们现在可以很轻松地构建复杂的图形用户界面（多年以前，图形用户界面是软件开发中工作量最大的部分之一），很方便地对数据进行管理，进行调试；同时，一些新软件开发支撑环境与体系架构的出现，如C/S结构、J2EE、DCOM、CORBA等，一方面使程序员可以摆脱RPC、Socket等底层的方法，轻松地开发网络化的信息系统；另一方面使复杂系统的构建效率与质量大大提高。近年来，软件行业在这些方面的发展的确产生了巨大的成效，并为大多数人所接受。

然而，软件行业在其他方面的革新内容对软件业的实际贡献，以及在软件管理、设计、开发人员中的普及与实用程度，都远不如开发工具与环境的新内容那么明显与直接。

软件分析设计方法的革新主要体现在面向对象的技术与思想，数据与方法的封装、继承，多态及复用，使设计师们可以更加形象地用软件来描述客观事物及其相互关系与作用，这应该是一种非常伟大的创造，也在软件行业起到巨大的推动作用；然而，另外，虽然面向对象技术与思想的优势已为同行们所公认，但在实际工程中，除了大量使用面向对象的开发工具与语言以外，真正采用面向对象的思想来分析、设计系统、描述设计结果以完成实际项目的做法却远不如想象中那么普及。这是因为某些工程太简单而没有必要用到它，还是因为它太难以实用化，或者还有其他什么原因呢？

软件工程理论不断在构造新的理想：软件的生产能像生产一部汽车，生产一台电视机或盖一座大楼一样，有标准的零件，标准的结构，标准的流程，标准的生产线，标准的组装……理论家们充满了创造性的构想与信心，而大批一线的软件工程师们却对这种理想产生怀疑。

软件企业管理模式的革新也是大部分企业迫切的需求，先进的CMM出现了，其理论不断被人们所了解，大家都对那些已经获得某种CMM级别的企业产生了深切的崇拜与向往，并异口同声的宣布，“我们也要做！”但我们同时又发现，大量的软件企业在把CMM与自己企业的实际情况有机结合的道路上却遇到了令人头痛的困难与疑惑，这种困难与疑惑好像不仅仅是为接受新机制而“痛一时、利一生”那么简单。

问题出在哪里呢？为什么在软件工业化的道路上要达到像电子、机械产品的程度会那么困难？多年来，作者也一直思考这个问题。

本书认为，“对一事物解决方法的制定要基于正确了解该事物的本质。”那么，可能软件这个事物与传统其他产品的特征还是有着巨大的区别，这可能不只是量的区别，而是质的区别。

或许我们应该从这个角度重新审视一下过去采用与传统工业化类比的思想所得来的灵感、理想与方法。

因此，本书从软件与三维实物的本质性不同出发研究软件生产方法论，指出软件的本质是不能实现工业化生产的内在原因，提出了全新的先进软件生产模式与方法——代码驱动生产的概念，并称这将是适合软件本质的新途径。

这里，要首先感谢家人的支持！感谢长年在日本工作的好友郭果明与施文彪为我提供的一些信息，同时，电子工业出版社的袁金敏编辑和顾慧芳编辑也为本书的出版提出了许多宝贵的建议，在此一并衷心感谢！

何小朝

电子邮箱: [ageneral_he@hotmail](mailto:ageneral_he@hotmail.com)

2009年12月

目 录

第一部分 现状与困惑

第1章

困难与困惑

1.1 软件项目实施的困难	2	1.1.4 需求何时确定	4
1.1.1 漫长的维护期	2	1.1.5 令人头痛的边缘问题	4
1.1.2 什么时候开始编码 ——生产过程控制	2	1.1.6 鲁棒性	5
1.1.3 开发周期的精确估算	3	1.2 飞速发展？ 困惑	6

第2章

技术方法的抉择——软件设计师的困惑

2.1 引言	8	2.5 疑惑 方法的抉择	36
2.2 软件设计师的理想	9	2.6 方法到底解决了什么	38
2.2.1 什么是软件设计师	10	2.6.1 通用函数库、类库与 框架的复用	39
2.2.2 技术人员的理想	12	2.6.2 分层体系使生产集中于 应用本身	41
2.3 似有曙光 方法论	13	2.6.3 面向对象有机描述客观 目标	43
2.3.1 技术进步带来希望	13	2.6.4 “平台无关”是以“时间 换空间”	45
2.3.2 软件技术方法综述	14	2.6.5 其他本质探讨	46
2.4 现实	26	2.7 常用有效技术方法简介	47
2.4.1 新技术手段的使用状况 ..	27		
2.4.2 困难没有得到本质解决 ..	33		
2.4.3 许多人这样看	34		
2.4.4 痛一时、利一生	35		

2.7.1 分析与设计的组织	48	2.7.4 文档的生成	54
2.7.2 开发环境的选择	49	2.7.5 其他环节	55
2.7.3 编程	53	2.8 本章小结	56

第3章

如何有效地服务——技术管理者的烦恼

3.1 软件项目管理现状	57	3.2.3 过程控制问题	64
3.1.1 管理过多依赖个人经验 ...	58	3.2.4 测试的问题	66
3.1.2 管理规则很不成熟	59	3.2.5 需求制定的问题	66
3.1.3 管理组与开发组不协调 ...	60	3.3 管理者的希望与困惑 CMM ..	67
3.1.4 无通用规则可循，各有 特色	61	3.3.1 CMM简介	67
3.1.5 新方法的现实情况	61	3.3.2 PSP/TSP	69
3.2 软件项目管理的几个具体问题 ..	62	3.3.3 CMM应用现状	70
3.2.1 文档的管理问题	62	3.3.4 CMM的困惑	70
3.2.2 设计文档的作用到底 是什么	63	3.4 究竟怎么回事	71
		3.5 本章小结	72

第二部分 软件的本质

第4章

理想与现实——科学与工程的碰撞

4.1 软件工程的理想 生产线与 复用	75	4.2.2 软件生产产业的主体是 什么	76
4.2 工程实践 与理想模式的 距离	76	4.2.3 软件生产工程实践现状 ...	77
4.2.1 软件是数字化社会的 主体	76	4.3 飞速发展 的本质	78
		4.4 本章小结	79

第5章

不是三维——软件的本质

5.1 三维空间 人类形象思维的 局限	80	5.3.2 二维可视化设计语言能力 分析	83
5.2 传统产品简析	81	5.3.3 软件无法完全可视化 表达	85
5.2.1 传统产品——三维实体 ...	81	5.3.4 软件产品的核心—— 代码	85
5.2.2 传统产品的核心——设计 输出	81	5.3.5 一个有趣的问题	86
5.2.3 传统产品生产模式—— 生产线	82	5.4 软件的本质 不是三维	87
5.3 软件产品简析	82	5.5 本章小结	88
5.3.1 软件是一种特殊的产品 ...	82		

第三部分 软件生产方法论——实践与研究

第6章

设计与开发

6.1 引言	91	6.3.2 系统整体结构的软件 描述	123
6.1.1 构件化生产与自动化的 探讨	92	6.3.3 整体代码框架搭建	127
6.1.2 实用技术方法涉及的 内容	94	6.3.4 系统公共库设计	128
6.2 应用系统架构设计	95	6.3.5 接口设计	129
6.2.1 什么是架构	95	6.4 代码编制与调试	131
6.2.2 几种常见架构模式	96	6.4.1 程序结构搭建	131
6.2.3 实践中的系统架构设计 ...	107	6.4.2 多线程/多进程序序设计 ...	134
6.3 系统分析与设计	122	6.4.3 编辑	138
6.3.1 软件分析与设计的内容 ...	122	6.4.4 编译	140
		6.4.5 调试	142

6.4.6 类型安全	154	6.6 设计的结束与输出	169
6.4.7 跨平台程序设计	155	6.6.1 实效的制度化, 而非教条 的制度化	169
6.4.8 开源软件代码	158	6.6.2 强化设计结束准则	169
6.5 面向对象 正确理解与使用	160	6.6.3 设计阶段的输出文档	170
6.5.1 何时使用面向对象	160	6.7 本章小结	171
6.5.2 如何使用面向对象	162		

第 7 章

软件测试——质量保证的重要环节

7.1 测试技术综述	172	7.2.5 测试时间估算	183
7.1.1 测试的阶段	173	7.2.6 测试人员的素质	184
7.1.2 测试的种类	173	7.3 对测试标准的探讨	186
7.1.3 常用的测试工具	173	7.3.1 软件bug的分布特征	186
7.1.4 测试步骤	175	7.3.2 测试标准的制定原则	187
7.2 软件测试中几个实用问题	177	7.3.3 测试人员的困惑——测试 标准如何实施?	188
7.2.1 手工测试	177	7.4 软件测试的本质探讨与工业化 生产	189
7.2.2 自动化测试	180	7.5 本章小结	191
7.2.3 测试用例的制定	180		
7.2.4 关于测试驱动开发	182		

第 8 章

软件项目管理方法论

8.1 软件项目管理的产生	192	8.4 软件项目管理的实用方法	194
8.2 软件项目管理的作用与地位 为研发服务	193	8.4.1 软件规模划分	195
8.3 软件项目管理的内容	194	8.4.2 软件类型划分	198
		8.4.3 软件项目实施的阶段 划分	198

8.4.4 软件项目的最初需求	8.5 软件生产过程中的实用工具 ...215
确定200	8.5.1 软件配置管理系统216
8.4.5 软件项目实施的资源	8.5.2 软件缺陷追踪系统216
确定201	8.6 国外软件项目管理简析219
8.4.6 软件生产过程202	8.6.1 欧美国家软件项目管理
8.4.7 软件生产过程控制208	模式219
8.4.8 设计与编程的分界点209	8.6.2 最为特色的模式——日本
8.4.9 测试与研发的交流213	软件项目管理222
8.4.10 文档的生成与管理214	8.7 本章小结223
8.4.11 用软件进行软件项目	
管理215	

第9章

用软件管理软件

9.1 引言224	9.3 实用系统设计229
9.2 管理系统功能分析225	9.3.1 系统组成与总体构架229
9.2.1 问题的提出225	9.3.2 软件开发的计划与进度
9.2.2 CMM规范226	管理231
9.2.3 策略226	9.3.3 评分系统231
9.2.4 实现管理的要素227	9.3.4 界面示例231
9.2.5 关键方法227	9.4 管理系统再分析232

第10章

新一代软件生产方法论探索——代码驱动生产

10.1 原则与依据236	10.1.3 软件生产方法论应包括
10.1.1 以质为本，以实践为	现有成果237
依据236	10.2 软件生产技术回顾237
10.1.2 不能以类推法制定软件	10.2.1 概述237
生产理论框架236	

10.2.2 传统软件工业化生产 技术分析.....	239	10.6.2 代码驱动需求分析.....	261
10.3 先进软件生产方法分析.....	246	10.6.3 代码驱动设计开发.....	263
10.3.1 设计思维过程分析.....	246	10.6.4 代码驱动测试.....	271
10.3.2 先进软件生产方法应 有的特征.....	248	10.7 软件过程 代码框架 上的 并行协作.....	275
10.4 代码驱动生产的定义与内容.....	250	10.7.1 软件过程的两个要素.....	275
10.4.1 软件“工业化”生产的 新含义.....	251	10.7.2 软件生产平台——“代码 框架”.....	275
10.4.2 代码驱动生产的要素.....	251	10.7.3 并行协作的软件过程.....	276
10.4.3 代码驱动生产的组成 内容.....	255	10.7.4 生产过程迭代再探.....	281
10.5 软件生产模型 代码为核心 的星形模型.....	256	10.7.5 代码核心与文档弱化.....	282
10.5.1 传统软件生产顺序模型.....	256	10.8 其他讨论.....	283
10.5.2 以代码为核心的星形 模型.....	257	10.9 本章小结.....	284
10.6 代码驱动生产运行机制（Code Driving Production）.....	260	后记.....	285
10.6.1 代码驱动生产运行机制 综述.....	260	参考文献.....	289

第一部分 现状与困惑

第1章 困难与困惑

第2章 技术方法的抉择——软件设计师的困惑

第3章 如何有效地服务——技术管理者的烦恼

