

# CHAPT1-4

## 软件工程师的成长

衣杨

[815583456@qq.com](mailto:815583456@qq.com)

Wechart: vivangSYSU

# Agenda

1. 个人能力衡量与发展
2. 软件工程师思维误区
3. 软件工程师职业发展
4. 技能的反面
5. 作业与讨论

# 1 个人能力衡量与发展

- 软件工程：开发、运营、维护
- 软件开发流程：把相关技术和过程统一到一个体系
  - 目的：提高软件开发、运营和维护效率，提高客户满意度、软件可靠性和可维护性
- 用足球 比喻：
  - 个人：体能、技术、意识、斗志
    - 传、接、盘带、射门、定位球、跑位，等等
  - 团队：阵型、配合、临场应变
    - 442, 433, 451
    - 南美风、欧洲风、小快灵、两翼齐飞、全攻全守



- 问：你是职业软件工程师么？
- 答：是
- 问：你觉得你‘职业’到哪一个程度？
- 答：嗯，我在一个能发工资的地方上班，靠我的软件技术挣钱，所以我相当的职业

- 问：像职业篮球队员那样职业？

- 答：差不多吧

球员	出场数	先发	分钟	命中率	三分命中率	罚球命中率	进攻	防守	篮板	助攻	抢断	盖帽	失误
凯文·马丁	80	80	32.5	0.436	0.383	0.888	0.4	2.9	3.3	2.5	1.0	0.1	23
路易斯·斯科拉	74	74	32.6	0.504	0.000	0.738	2.0	6.2	8.2	2.5	0.6	0.5	19
凯尔·洛瑞	75	71	34.2	0.426	0.376	0.765	1.2	2.9	4.1	6.7	1.3	0.3	21
姚明	5	5	18.2	0.486	0.000	0.938	1.4	4.0	5.4	0.8	0.0	1.6	14
蔡斯·巴丁格	78	22	22.3	0.425	0.325	0.855	0.7	2.9	3.6	1.6	0.5	0.2	03
康特尼·李	81	1	21.3	0.439	0.408	0.792	0.6	2.0	2.6	1.2	0.7	0.2	03

- 问：职业篮球队员都有很详细的记录说明，例如，途中说明了一个职业篮球运动员2010赛季的场上表现
- 问：你有类似的数据说明你的职业活动和成绩吗？
- 答：嗯。。没有。唯一的数据是，我的上场时间还是挺长的，而且经常打加时赛（加班）

- Individual contributor (IC)
- 通过交流、实践等，理解问题、需求和任务
- 提出多种解决方法，包括寻找以前的，做重复性的工作
- 与相关角色交流解决问题的提案，形成一个可行的方案
- 执行，完成代码，验证方案可行性和其它特性
- 与团队其它角色合作，在测试中完成方案，修复bug
- 显然，还有很多情况没有处理
  - 如果方案问题严重，考虑其它方案

- 积累软件开发相关的知识、提升技术技能（如对具体技术的掌握，动手能力）
  - 例如：对Java、C/C++、C#的掌握，诊断或提高效能的技术，对设备驱动程序（Device Driver）、内核调试器（Kernel Debugger）的掌握；对于某一开发平台的掌握
- 积累问题领域的知识和经验（例如：对游戏、医疗或金融行业的了解）
  - 随着经验的增长，一个工程师可以掌握更广泛、更深入的技术和问题领域的知识
- 对通用的软件设计思想和软件工程思想的理解
  - 什么是好的软件设计思想？什么是好的软件工程思想？
- 提升职业技能
  - 自我管理的能力，表达和交流的能力，与人合作的能力，按质按量完成任务的执行力
- 实际成果
  - 用户评价、市场占有率、对用户的价值，这些工作成果是最重要的评价标准

# 软件开发的质量如何衡量

- 项目有多大
  - *LOC/FP (line of code/ function point)*
- 花了多少时间
  - *小时、天、月、年, 人数\*时间*
- 质量如何, 代码交付时有多少缺陷
  - *交付: 代码完成时交给测试人员; 软件发布时交给客户*
  - *rework 的理解*
- 是否按时交付
  - *实际用时, 估计用时, 平均时间, 时间方差*

任务	AI 的估计	AI 的实际用时	Bob 的估计	Bob 的实际用时
1, 2, 3	10 days	5, 10, 15	10 days	5, 10, 15
4, 5, 6 (3 months later)	7 days	1, 9, 11	8 days	7, 8, 9
		Average=7		Average=8
		StdDev=5.3		StdDev=1



- 软件项目的确需要创造性、一些意外、一些惊喜
- 更常见的是重复性的任务
- 瓦兹·汉弗雷：软件领域可以分为两个方面，一方面是艺术创新下的大爆发，另外一方面是坚持不懈的工程项目，包括软件改善、维护和测试等，这方面占了90-95%的比例
- 一个成熟的软件工程师应该降低任务交付时的标准差
  - 如果能长期的稳定的按时交付，内部和外部的顾客都会对你的工作有信心

# 团队对个人的期望

- 交流
- 说到做到
- 接受团队赋予的角色🤖并按期完成
- 权力投入团队活动：评复审，不是游离于团队之外
- 准备：开会之前，开启新功能、新项目之前，都做好准备工作
- 理性的工作：从实际数据出发、按流程
  - Chuck Close 我觉得灵感属于业余爱好者的，我们的职业人士只是每天持续工作。你今天继续昨天的工作，明天你继续今天的工作，最终你会有所成就

## 2 软件工程师思维误区

- 分析麻痹：只看不做
- 不分主次，想解决所有问题
- 过早优化：理想化
- 过早扩大化/泛化 (premature generalization)

画扇面	做软件工程团队项目
相声是一门说学逗唱的艺术。。。甲：我刚买了一把纸扇乙：哦，拿来看看，一把白纸扇。。。上面空空如也太可惜，拿不出手啊。如果能在上面画上画就更好了。我这几天也没什么事，我就给你免费画画	软件工程讲究的是需求分析，项目管理，开发，测试和维护甲：我觉得咱们团队项目做一个好用的小工具就好了，我已经做好了一个原型。乙：这想法固然好，但是我们这些编程高手，就做这么小的工具，未免拿不出手。我们要把它搞大！
甲：太好了，您能画什么？乙：画个美女图怎么样？美女出浴图或美女春游图都可以考虑甲：我激动ing。。。	甲：那我们做什么呢？乙：我们扩展一下，把所有相关的功能都实现了，一统天下，几种工具结合起来！甲：激动ing。。。
过了几天。。。甲问：我那美女画好了吗？乙：喔，美女，画好了！你看这美女的小脸蛋儿，眼睛稍微大了一点。。。但是，我不如给你改成张飞算了！都是人体，我可以很快重构一下，我画张飞最拿手了，过几天就好。	过了几周。。。甲：通过调研才发现，这么多工具都有自己独特的需求，不同需求互相冲突，不好协调，怎么办？甲：我们可以做成一个通用的工具，统一需求，解决用户从头到尾的问题
过了几天。。。甲问：我那张飞？乙：张飞？！对，张飞也画的差不多了，嗯，你看着张飞的胡子，这身躯。。。是粗了一点。。。要不咱们画成山水，这张飞。。。马上就可以变成一块怪石！甲：大热天的我等这扇子。。。我容易吗。。。	过了几周。。。甲：通用的工具听上去很好，但是太痛用了，不好掌握，我们到底实现那些具体的功能呢？乙：我们可以做成一个开放式的平台！这样所有人都是可以做一个插件，来实现这个平台的一些功能！而且别人还可以用我们这一个通用的框架开发任意的软件，你想想一框架，任意软件啊！我们都会是架构师啊！甲：我激动，不过我们软工课快要结束了，要交作业啊
过了几天。。。甲：我那山水？乙：啊，山水。。。我也画好了。你看那巨石，很巨大也很给力吧，构图有点那啥，我再改改。甲：您什么时候画好？这夏天都快过了！乙：嗯，我的山水画还是有些宋人风格的。。。假以时日。。。如果你着急用扇子，这样吧，我把扇面全图黑了，你再找人往上写字好了！	过了几周甲：项目发布时间到了，我们的平台还没有，工具还没连起来，怎么办？乙：咱们可以把项目开源到网上，另外也许有很多开源的朋友闲着没事，可以给我们的代码写一写注释等。这是我们对开源运动做的巨大贡献，我们虽然没有创造出产品，但是我们都输出了价值观，这可不是每个程序员都可以做得到的。甲：那期末怎么交差啊？乙：赶紧写PPT！

## ■ Linux 刚写的时候

- I am doing a (free) operation system (just a hobby, won' t be big and professional like gun) for 386(486) AT clones...
- 管理学大师彼得. 德鲁克的忠告
- Those entrepreneurs who start out with the idea that they will make it big- and in a hurry- can be guaranteed failure.

# 3 软件工程师的职业发展

- 临时的寄托或工作 (temporary work)
- 工作 (job)
  - 赚钱养家
- 职业 (profession)
  - 对30、35岁以后都有打算
- 投身的事业 (commitment/vocation)
  - 长期承诺，遇到困难不退缩，一直坚持到完成
- 理想的呼唤 (calling)
  - 通过软件改变世界

# 专和精的关系

## ■ 对比：乐手/作曲家



## ■ 全栈工程师



- 职业发展
  - 考级之路
- 职业成长



- Steve McConnell 创立的 Construx Software 为员工提供一套成长路径



# Steve McConnell 版本

- 知识：相关知识分为十大领域
- 能力，四个阶段
  - 入门（introductory），熟练（competency），带头人（leadership），大师（Mastery）
- 工程师分为8个级别（8-15）
  - 例如要到12级，工程师必须在3个领域达到带头人
  - 例如要达到工程的呢个管理（知识领域）熟练（能力）水平，要做到
    - 阅读：4-6篇经典文献的深入分析
    - 工作经验：参与并完成6个具体项目
    - 课程：参加3个专门课程

# 职业成长-大公司版本

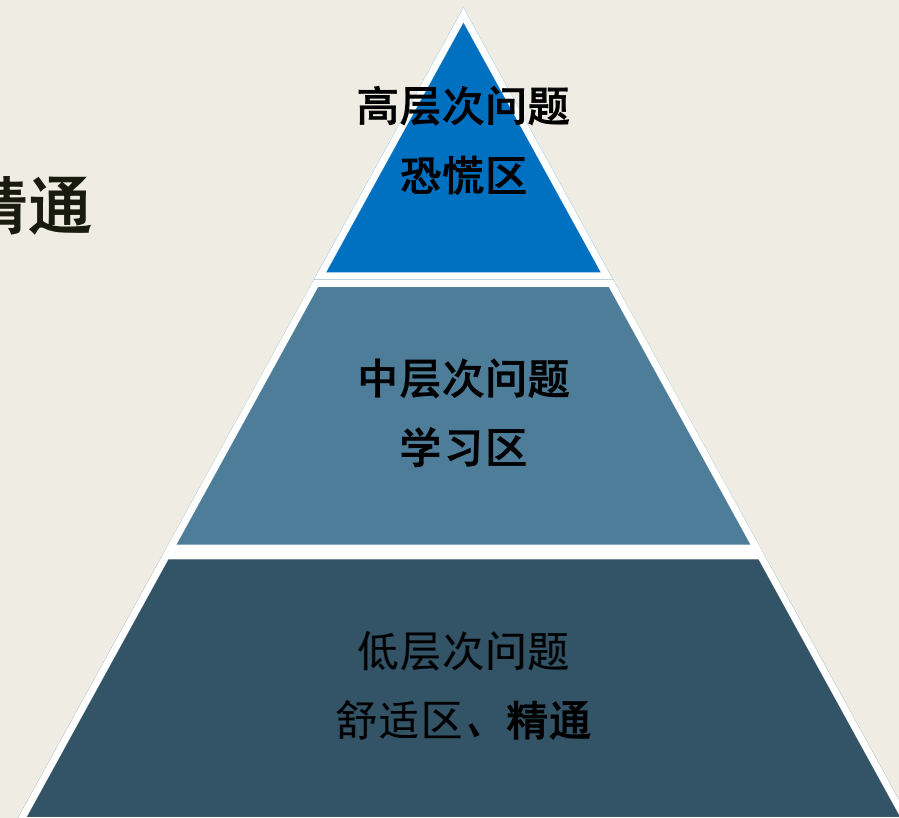
等级	要求
SDE（初级软件开发工程师）	入门。在学校里学到了一些技能，尚未在实践中得到充分锻炼
SDE II（中级软件开发工程师）	独立。可以写别人交给你的任何东西，不明白时知道去问谁
Senior SDE（高级软件开发工程师）	小组领导。影响着3-12名工程师，或者是他们的行政领导；或者是他们的技术带头人
Principal SDE（首席软件开发工程师）	团队领导。影响着10人以上的一个大团队，成为影响团队成败的关键人物
更高的职位有：Partner SDE、Distinguished Engineer、Technical Fellow	影响力扩大到整个机构，甚至工业界

# 职业成长-自我评估

基本需求	基本技术	扩展技术	进一步的扩展技术
把数据放到数据空中满足增删改查的需求	数据库技术（关系数据库的基本原理和操作）	大容量的数据库操作、并行、备份等技术	关系数据库模型、数据挖掘、商业智能
有网页满足一般用户的查询需求	网页服务技术（ASP.NET、PHP等），数据绑定及控件	用户界面的设计，对不同浏览器的支持	用户心理，用户交互的原则在不同设备和不同场景下的应用
能不断实现新的功能	编程语言和开发工具（Java、C#、Python）	程序的效能分析，软件的重用，面向对象的理论等	能改进软件工具，或构建新的语言提高解决问题的效率
软件团队能按时高质量完成任务	每日构建，版本管理，单元测试，项目管理	需求分析，敏捷开发等高级软件工程的技术	软件团队的绩效评估，团队的培训和发展
要有一定的安全性	数据库安全，网站的安全	计算机网络与数据通讯，操作系统的知识，数据加密解密	密码学，各种病毒工作原理
能满足业务的需求	对业务领域有基本的了解	进一步了解业务领域知识	对业务领域有深入了解，能观察行业发展的趋势

## 4 技能的反面

- 低层次问题（变成自动操作）：舒适区，精通
- 中层次问题（花脑力解决）：学习区
- 高层次问题（无暇顾及）：恐慌区



- 选择合适的去学习，不断构建自己的舒适区，拓展学习区，最后在某些领域达到精通，循序渐进

# 5练习与讨论

## I. 你选择哪个医生

- a. 刚看病，谨慎
- b. 富有创新意识
- c. 熟练，边做边讲故事
- d. 无文凭和资质，号称秘方治百病

- 事实上，很多项目就是 a b, 个别的 d
- 你选择哪种医生？
- 医生、药剂师等都有职业资格证书，软件工程师需要正式的证书吗？

## II 各式各样的工程师

- <http://blog.jobbole.com/21881/>

- <http://blog.jobbole.com/12176/>

### **III 我们经常看到一些专家讨论，他们有自知之明吗？**

- **<http://www.psmag.com/health-and-behavior/confident-idots-92793>**

**IV 程序员经常学习、交流、提高自己，最终我们的核心竞争力是什么？打字快就能写程序快吗？**

- **<http://www.zhihu.com/question/27180582>**
- **<http://news.cnblogs.com/n/509554>**
- **<http://www.tuicool.com/articles/r3852177.html>**