

<<软件工艺>>

图书基本信息

书名：<<软件工艺>>

13位ISBN编号：9787115280688

10位ISBN编号：7115280681

出版时间：2013-1

出版时间：人民邮电出版社

作者：Pete McBreen

页数：240

字数：164000

译者：熊节

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<软件工艺>>

内容概要

《软件开发方法学精选系列：软件工艺》针对软件开发，提出了一些相当棘手和敏感的问题，并给出了颇具争议性的结论：从一个数百年来一直兴旺发达的系统——工艺学中获得启示，寻找答案。

《软件开发方法学精选系列：软件工艺》用5个部分共19章的篇幅，系统地阐述作者的观点，并试图回答一直困扰着软件行业的难题——我们应该如何重组软件构造的过程，使其能够如我们所愿地有效运转？

第1部分共4章，对传统的观点提出质疑——软件工程真的是解决软件开发问题的灵丹妙药吗？

第2部分共2章，这一部分提出了本书的观点，即以软件工艺的视角看待软件开发。

第3部分以7章的篇幅，从不同的角度全面地展现了软件工艺理论所带来的主要变化，以及如何实践这个观念。

第4部分共3章，对比了软件工艺与软件工程，并为各自适用的范畴重新划定了界限。

第5部分共3章，分别讨论软件开发中的权宜之计和长期问题。

本书荣获2002年度Jolt图书大奖。

阅读本书，有助于引发读者在软件开发问题上的独立思考，《软件开发方法学精选系列：软件工艺》适合软件行业的所有从业人员阅读参考。

作者简介

Pete

McBeen是一位独立顾问，对软件开发情有独钟。

尽管将很多时间用于写作、教学和顾问工作，但他仍然坚持每年至少在一个真实项目中亲手从事编程工作。

Pete特别善于为软件开发者面临的问题找到创造性的解决方案。

在过去的很多年中，他参与了各种正式与非正式的过程改进活动，所以他能够以超然的态度看待软件业普遍存在的问题，并敏锐地意识到：“软件开发理应有其乐趣。

否则，开发过程就是错的。

”Pete住在加拿大亚伯达省的小镇考昆，没有再回到大城市居住的计划。

<<软件工艺>>

书籍目录

第一部分 置疑软件工程

第1章 理解软件工程

软件工程的悖论

等待硬件开发时，软件开发者在干什么？

得到可用的硬件之后，软件开发者如何加快交付的速度？

传统开发过程的内蕴

软件工程的当代解读

“足够好”的软件-庶民的软件工程

软件工程适合你的项目吗？

第2章 软件工程的困境

“有组织的、可计量的”软件开发过程现实吗？

我们当然可以将软件开发中的某些部分自动化，不是吗？

“足够好”的软件开发方法的危害

谁能取代软件工程？

第3章 理解软件开发

软件资产

软件开发需要团队协作

软件开发的分工有用吗？

没有一劳永逸

寻找比“软件工程”更合用的隐喻

第4章 寻找一个比软件工程更好的隐喻

软件开发的工艺

与传统工艺学的比较

软件开发工艺的复兴

第二部分 软件工艺

第5章 重拾软件开发

工艺学致力于改善软件开发的现状

工艺学鼓励开发者编写优秀的软件

吹响号角

第6章 无须执照的工艺学

工艺是私人性的

同行认可和推荐是获得更好软件的办法

执照只是假象

执照是在向风车开战

工艺学关注个人

软件开发者不是太少，而是太多

第三部分 软件工艺隐含的意味

第7章 工艺学对系统的用户有何影响

<<软件工艺>>

软件容易拷贝，所以软件工艺能够有效
批量市场的难题
工匠与用户有一种不同的关系
但是，请记住：购买者很可能不是使用者
优秀的软件应该签上开发者的名字
为作品签名会使情况发生变化
工匠应当对作品负责
工匠需要挑剔的用户
更小、更坚固的软件更有利于用户
软件工艺带来协作式开发
第8章 顾客与工匠的关系
给我一个真实的交付日期
揭穿“足够好的软件”的谬论
另一种选择
不要只考虑出价最低的开发者
差劲的客户将很难吸引优秀的开发者
让软件工匠因为自己的作品而获得荣誉
要求开发者对作品负责
利用开发者之间的差异
雇佣优秀开发者组成的小团队
优秀的开发者究竟值多少？

但我们如何知道开发者有多优秀呢？

根据交付的成果来衡量开发者的水平
在选择工匠时，客户在成本和质量之间作出权衡
软件工匠的专业分工
客户与软件工匠有长期的联系
维护者是一个荣耀的身份
软件工艺有益于长期使用的软件
客户与软件工匠志趣相投
第9章 工匠的管理
软件工匠不是雇工
好的开发者比管理者更有价值
软件开发的实际过程无法详细定义
软件工匠与管理者的关系
以管理优秀的开发者为乐为荣
优秀的管理者理解项目的节奏
软件工匠喜欢创造软件
软件开发的根本从来没有改变过
家有一老，如有一宝
软件工艺要求全新的管理方式
软件工艺不是“有计划报废”
软件工匠坚持自己的要求
第10章 成为软件工匠
软件工艺拒绝精细的分工
过度的专业化会延误开发、导致错误

<<软件工艺>>

软件工匠建造能够理解的系统
工艺学需要献身精神
如何成为软件工匠？

学徒是比学校教育更有效的学习方式
技师是工艺学传统的关键
工艺学传统已经延续多年
第11章 工艺的掌握
软件工艺大师是什么样子？

善用你的老员工
“掌握技艺”意味着使用稳定的技术
软件工匠不会仅仅因为工具“最新最好”而使用它
软件工程对COBOL的谋杀
技艺需要花时间去掌握
“掌握”意味着承担起传递工艺的责任
工匠挑选学徒和技师
第12章 学徒开发者
我们必须扭转开发者培训质量下滑的局面
大学文凭与项目开发无关
会编程不等于会开发软件
如果必须送初学者去培训，选择好的培训课程
工艺的掌握，学徒比培训更有效
成为学徒是重要的一步
为了降低对工作的影响，工匠慎选学徒
重要的是学，不是教
学徒不是学校
活到老学到老
学徒审查师傅的作品，并从中学习
学徒的角色
从低风险的任务开始
晋升到产品开发
因为能力而晋升
学徒不是廉价劳动力
学徒期是时间和精力投资
学徒如何成为技师
第13章 技师开发者
技师在工艺学传统中的位置
技师开发者
技师很少单独工作
技师关注应用程序的交付
技师在软件工艺中扮演关键角色
第四部分 重新定位软件工程
第14章 软件工程项目
软件工程的目的是大型系统项目
软件工程需要专业分工
软件工程项目依旧使用瀑布过程

<<软件工艺>>

编程是一项刻板的工作
软件开发不是软件工程项目的瓶颈
形形色色的软件工程项目
敏捷方法代替缜密的软件工程
第15章 “软件工程” 隐喻的危害
无法以低成本实施软件工程
鱼与熊掌可以兼得？

相信估算软件工程项目的确需要很长的时间
软件工程鼓励 “科学管理”
软件工程轻视不精确的讨论
软件工厂：软件的生产线
跨项目复用极难实现
冒险的 “长时间复用”
“标准软件开发过程” 的迷思
传统的分工无助于软件开发
“最佳实践” 是 “科学管理” 的遗毒
最佳实践使人墨守成规
最佳实践阻碍了过程革新
软件工程强迫我们忽视个人
软件开发者不是可替换的资源
伪造一个 “理想的开发过程”
开发过程，不嫌其多
抛弃软件工程的瀑布式过程
瀑布方法需要大型团队来实施
小型团队绝不要尝试软件工程
第16章 学习软件工程的经验
尺度和复杂度
做软件，不容易
应用程序需要良好的结构
变化的代价很高——如果你不允许变化的话
交流至关重要
文档总是错的
用增量式开发来控制风险
精确的估算很难得到
借用这些经验
第五部分 星期一的早上
第17章 经验——项目成功的指示灯
根据声望选择软件工匠
信任工匠的推荐
最后，开始大范围搜索
根据声望和作品来评价工匠
考察工匠的作品
工匠的试演
由软件工匠来组建开发团队
根据个人了解和推荐挑选团队成员
年富力强的开发团队

<<软件工艺>>

为低预算团队担心

通力协作

使用增量式开发

尽早解决问题

任何人都能学会协作式开发

回避极端技术

经验的价值

他们去年在哪里？

奖励优秀开发者

想要人才，就得付高薪

我们付得起那么多钱吗？

做好吃惊的准备

第18章 为测试和维护而设计

是软件应用，不是软件项目

应用程序只会退休，不会结束

维护团队理应拒绝丑陋的软件

可维护软件需要有自动测试

使应用程序能够被测试

为维护而设计

创建可维护软件需要经验丰富的开发者

可维护软件能够生存多年

长寿的应用程序需要长寿的开发工具

开放源码，软件工艺的最爱

Java对项目的健康有害

可维护软件需要稳定的基础设施

优秀的软件是全球性的

保证软件的全局性

拒绝“有计划报废”

优秀的软件需要优秀的用户界面

能够安全使用的软件

可维护软件易于诊断

外包的危害

外包忽视了软件开发的本质

在外包中坚持软件工艺

借助外来的工匠

维护是软件生命中最重要的一部分

提高维护者的地位

维护者当受赏

并非所有软件都必须可维护

“为测试和维护设计”不能一蹴而就

第19章 活到老，学到老

创造学习的环境

用内部研讨创造学习环境

邀请所有人参加讲座

学习时间是一种投资

<<软件工艺>>

掌握软件开发的技艺
鼓励参加用户组和技术会议
慎选培训课程
课前联系
课后跟踪
亡羊补牢
鼓励员工活跃于开发者社群中
鼓励出席技术会议
鼓励开发者担任讲师
鼓励开发者写书
沉思的实践者

.....

<<软件工艺>>

编辑推荐

《软件工艺》曾获有软件行业奥斯卡之称的Jolt图书大奖，由知名技术作者、CSDN技术主编孟岩，资深技术编辑韩磊倾力推荐，知名IT技术专家熊节主笔翻译。

《软件工艺》是颠覆软件工程、开发回归人本，值得深思回味的经典力作。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>