

The Tao of
Programming

编程之道



[美] 杰弗雷·詹姆斯 著
郭海 郭海 译



清华大学出版社

<http://www.tup.tsinghua.edu.cn>

译 者 序

如今程序设计类的图书可谓是层出不穷,其中绝大部分是以教程、使用手册和编程技巧为主,它们都是用来处理一些技术性问题的。

然而,怎样领悟程序设计的内涵和精髓呢?程序员编程时是一种怎样的思维方式?他们的工作、生活是个什么样子?他们在想些什么?作为一名经理如何同他们相处并充分发挥他们的积极性?涉及这些方面的书却寥寥无几。但没有并不代表不重要——随着我国程序设计事业从起步到发展,这些方面的内容显得越来越重要,我们越来越希望了解它们。

美国是个软件工业和程序设计水平处于世界领先地位的国家,我们是不是可以从那里借鉴点什么呢?您将要读到的《编程之道》一书便是出自美国的一位善于进行哲学性思考、有十多年工作经验的程序设计师——Geoffrey James 之手。

在本书中,作者以一种幽默、辛辣的眼光审视着发生在程序设计室里的各种各样的小故事,并运用东方的哲学体系进行深层次的思考 and 理解,即进行“道”式的思考 and 理解。简单的故事蕴含深奥的道理,是本

书的最大特色。

本书不仅语言优美、比喻生动,而且在结构上颇具匠心。

本书在美国出版时分为三册——《编程之道》、《编程之禅》和《计算机寓言——信息时代的启示》,经与作者协商,决定将中译本汇总为一本以《编程之道》的书名出版。

1. 《编程之道》

本书的构思新颖而奇特。作者作为一名计算机考古的业余爱好者,偶然发现了在程序设计业内传说的《编程之道》。这其中的内容,每个程序员都想知道,但谁也没有见到过,就有点像金庸的小说《射雕英雄传》中的《九阴真经》。于是作者便将它“翻译”成此书,公之于众。全书共分为九篇:

- 第一篇 静寂的空字
- 第二篇 古代的大师
- 第三篇 设计
- 第四篇 编码
- 第五篇 维护
- 第六篇 管理
- 第七篇 公司里的学问

第八篇 硬件和软件

第九篇 尾声

2. 《编程之禅》

在本书中,作者借用中国传统文化中的五行——“金、木、水、火、土”来代表与计算机相关的概念,并以此将多个富有禅宗色彩的故事有机地组织起来。全书共分为五篇:

第一篇 木

第二篇 火

第三篇 土

第四篇 金

第五篇 水

3. 《计算机寓言——信息时代的启示》

作者以四季——“春、夏、秋、冬”来组织本书的结构。全书共分为四篇:

第一篇 秋——虚幻与真实

第二篇 冬——战争与死亡

第三篇 春——真爱与新生

第四篇 夏——启示

我们将此书献给那些爱笑也爱思考的程序设计者,以及那些同程序设计者打交道的管理人员,当然他们也爱笑爱思考!

编程之禅



前 言

《编程之道》的出版在程序设计业内得到了普遍的接受,出版社邀请我翻译一些与之相关的文章,以作为那部著名的经典之作的补充。虽然,我申明我的能力难以胜任,但我最终还是被说服试一试。

本书是许多个月来研究和翻译的结果,它试图通过对那些传统著作的摘录,来包装一个复杂的主题。我不怀疑,将会有许多计算机考古专家对我的摘录标准提出意见。他们会问,“他为什么不把《UNIX 程序员、大象和妓女》的寓言包括进来?”“他竟敢忽视历史悠久的故事《图灵集市历险记》!”对于这些批评,我只能说,我已经尽了我最大的努力来选择有代表性的内容。

为了确定文中各个章节的年代,我利用了朝代系统。对于那些不熟悉这套划分年代的方法的人来说,可以按四个朝代(或者说“四代”)来划分。

第一个朝代,即所谓的“黄金时代”,要追溯到由玻璃管构建计算机的日子;许多现代的学者断言,这个年代具有神话色彩。第二个朝代开始于晶体管的发明,而结束于集成电路的出现。

现代计算机的历史,开始于第三个朝代,它由主机和控制主机的军机大臣所统治。第四个朝代开始于对“集成教”的镇压,“集成教”对业已建立的秩序的反抗,被狂

热的“蓝色兵团”残酷地摧毁了。具有讽刺意义的是，正是这次镇压，导致编程之禅扩散到了外部世界。

除了组成该书主体的古老的素材以外，我还有幸得到了 Babbage 博士和 Yu 博士的帮助，他们分别为该书撰写了引言和序。我希望他们对该书的贡献能够或多或少地弥补我作为编者的不才。

Geoffrey James

1988 年 1 月于洛杉矶

根据古老的“五行学说”，本书由五部分组成：

木	火	土	金	水
大师：忍者	行者	隐者	大力神	阿幸
媒体：编年史	民间故事	说教	公案	俳句
硬件：键盘	显示器	CPU	打印机	磁盘
方向：东	南	中	西	北
软件：编辑程序	格式化程序	调试程序	解释程序	编译程序
感觉：触觉	视觉	嗅觉	听觉	味觉
阶段：界面	设计	编码	排错	测试
动物：龙	凤凰	牛	兔	蛇
系统：VMS	CP-6	OS/VS	MS-DOS	Unix

这套分类系统包罗万象，要想在该书的篇幅内把它阐释清楚是不大可能的。然而，在沉思于这些千变万化的对应关系之后，您或许会发现其中那令人惊异的洞察力。

序

毫无疑问,古老的编程艺术在西方人的脑海里通常会被误解,现代的观点则认为这是一种工程学、机械论和唯物论。

许多人认为,编程仅是为了达到一个目的而使用的一种手段,并且,一个程序(及程序员)只能依据其赚钱的能力来评价。

这些原始的错觉,来自于对编程的真正意图的一种很深的误解。

高超的程序员并非致力于表面的成功,而是在人机之间寻找一种神秘的存在。

根据禅的解释,在硬件、软件、固件、界面和理解之间没有分界——相反,所有这些都结合进了一个和谐的整体。

只有当程序员最终学会了将那种“自我的存在”的虚假感觉抛在一边,此种境界才可能达到,而那种“自我的存在”的感觉往往伴随在我们大多数人的生活之中。

这是计算机所带来的——禅师程序员的行为方式。

有人说,掌握了禅机的程序员也就掌握了生命。这样的程序员是以一种永不疲倦的和童稚的欢乐来看待这个世界的。

觉悟了的程序员,走在大街上也能感觉到两旁房子

和大楼里面的电脑。

觉悟了的程序员,能够感到和听到调制过的数据穿过电话线时那种电流脉冲的持续不断的嗡嗡声。

觉悟了的程序员已经与宇宙合为一体。

我以前的学生 Geoffrey 能够完成这样一个工作,将失落的编程之禅的经典著作再现于世,作为一名教师,我感到由衷的满足。

真心希望,这本书能够重新确立禅在完善的程序员教育中的重要地位。

C.P.Yu 博士

莲花大学超越机器学院

西藏拉萨

引 言

当 James 先生要求我给他的这本书写一篇引言的时候,除了讲述一下关于我自己在程序维护这个神秘的领域中的个人经历,我实在想不到更好的办法了。程序的维护是程序设计艺术中很少被人理解的领域。

有的读者肯定会坚持说,在禅的深奥教义和程序维护的卑微技艺之间,并没有什么共同点。但正如一位大师所言:“方法和途径(也就是道)存在于所有的程序之中,甚至在电子游戏里。”因此,长期被忽视的程序维护艺术必定有其禅的方面,这点是确凿无疑的,虽然那对于未经过训练的头脑来说或许不会立刻显现出来。

我的故事开始于我从大学计算机专业毕业后的几个星期。我大学毕业后的目标是为一家研究和开发机构工作,最好是编译程序或操作系统设计的部门。我最后终于找到一家愿意雇佣我的机构,但条件是我必须在一段时间内,通过完成程序维护的工作来“学会这套系统”。

我当然会对这个建议产生抵触心理。我花了五年的大学时间,竟然是为了浪费时间来解决其他一些程序员的错误!然而,因为他们曾经答应日后让我干感兴趣的工作,于是我便接受了,并在心中提醒自己,即使这份工作干不出名堂,我还能找到其他的工作。

当我第二个礼拜去报到的时候，我被领着去与那个程序维护组的师父见面。人事部总管带着我快步地穿过开发中心黑暗的走廊，最后她指着长长的大厅通道尽头处的一扇门说：“他在那里面。”说完便转身走了，好像有些心神不定。

我向门口走去，眼睛朝里面窥探，我看见一个男人正坐在终端前工作，但他背对着我，所以我不知道他有多大年纪，长得什么样。我只有靠咳嗽一声来表明我的存在，这时，那位大师甚至没有回头瞅一眼便说：“请坐。”

越过他的肩膀，我瞥见了那些难以理解的屏幕显示，随着他纤长的手指在键盘上飞舞，它们一闪一闪地出现在他的终端上。最后，他满意地咕哝了一声，退出系统，然后转过身来面对着我。

我所看见的让我吃了一惊，因为他看上去不像那种应该是禅师的人，他的脸平淡乏味，几乎是丑陋的，他的头发如同一圈烦恼的光环。但人们首先注意的是他的眼睛，透过厚厚的眼镜片射出淡淡的蓝光。

他从头到脚地打量了我一番，然后点点头，就好像确认一个个人观点，“你就是那个新来的？”他酸酸地问。

“是的。”我回答。我假装充满激情，把我的经历和在大学的成绩向他作了快速的汇报。

这位师父礼貌地听着，然后说：“那很好，但你以前做过程序的维护工作吗？”

我坦白地告诉他我没有做过。

师父长长地叹了口气，“那好，我们应该做我们能做

的事。”他说。然后他从一个架子上取下一本庞大的程序清单,随意地翻开它,并递给我。接着他问:“你怎样对付这个?”

我眼睛盯着这本清单,那上面是汇编语言,还搀和着一些奇怪的宏语言。每隔十几行语句,控制权便转移到某个神秘的子程序,而且,即使这个程序具有任何结构性,我也看不出来。“这是什么程序?”我问。

师父把那本清单从我的膝盖上拿了过去,“这是《世代大师编码藏经》,”他说,“当你学会了从迷魂阵中把错误的代码抓出来时,那就是你出师的时候了。”然后,他合上清单,把它放回到书架上。

我不久便意识到,程序的维护远比我想象的要难得多。我起初试图学会那本《编码藏经》中所用的汇编语言,但令我烦恼的是,我发现那套汇编语言从未按规矩提供过说明文档,现存的仅有一些笔记,是一位多年前就死去或离开公司的硬件开发者写的。

这本《编码藏经》不能给我什么帮助了。虽然其中偶尔有一些注释,但这些注释和这种汇编语言一样模糊,除了一些涉及到有关原始硬件构造的使人干着急的资料以外,别无它物。

当我向那位师父抱怨这一切的时候,他礼貌地听着,并在我们两人沉默了许久之后,回答我说:

“你正在试图理解某种不可能被你的理性思维所理解的东西,”他说,“这样所导致的结果必然是失败。你必须先清空你的大脑,只有那时,你才能开始领悟《编码藏

经》。”

接着，师父开始慢慢地向我阐释那本《编码藏经》里复杂的逻辑。当我聆听他那平静而温和的声音时，我终于开始觉察到一点光芒的闪烁，这是隐藏在《编码藏经》中辉煌的和永恒的灵光。“那些大师对‘良好的程序设计惯例与规则’一无所知，”师父说，“他们努力去理解普遍意义上的计算机思维的内部活动，他们还需要什么说明文档呢？那些程序是极限的表达。”

然而，即使我慢慢开始有所领会，我还是感觉到自己像一只在琥珀中挣扎的小飞虫。师父所讲的与我曾学过的东西截然不同，我的理性思维很难接受。但师父总是耐心地一遍遍解释说，我必须不靠理性思维去推理，而要用潜意识来领悟《编码藏经》的内涵。

经过数月的指点后，我感到很自信了，便想试试我的第一块修补程序。为了给师父一个惊喜，我偷偷地工作。我写了一段能重新运行几行语句的补丁，把程序重新进行汇编，并把新程序释放到生产系统中去。

第二天早上，我来晚了一会儿。让我感到奇怪的是，开发中心的主任和人事部总管正在师父的办公室里。当我走进大厅后，人事部总管看见了我就把门关上了。我听见了很大的说话声，但听不清说什么。

等到那两位来访者离开后，我走进师父的办公室。“什么事？”我问。

“你的修补程序，昨天晚上六点整进入了生产系统。它现在已经被删除了。”

“那么？”

“你仍然可以继续工作。”师父说。

最后，我终于明白，所有用我的理性思维去理解《编码藏经》的努力都是徒劳的，这使我很绝望。师父觉察到了我思想上的这个变化，他开始向我传授天机。他教给我冥想和查错的技巧，他说这些技巧是从计算机时代之初，由技术支持部一代一代地传下来的。

我听着听着，开始意识到一个关于我以前的编程体验的特大事实。在大学里，我一直认为程序员的工作主要是控制硬件和软件的活动，编程的最高艺术境界是，为了完成一项任务或达到一个目标而成功地运用高超的编程技巧。但程序维护是不同于程序开发的。要想维护一个程序，就要把程序看作是一棵渐渐成长的植物，拔苗助长是毫无益处的。事实上，这样的行为更容易导致植物的死亡。一个程序必须被细心地滋养。程序员在对程序进行改动之前，必须对每一个逻辑关系非常熟悉，必须对程序的意图有很深刻的理解力。这种理解力不是在一夜之间就可以得来的，需要长时间的积累。

好几个个月以后，我终于能够给《编码藏经》成功地设计修补程序了，但设计之前要经过长时间的冥想，还要把那本程序清单打开着支在我的桌子上。我还发现，如果我工作的时候点燃一支香，口中不停地重复师父教给我的口诀：“Null - So - Stix - Etx - Eot”（ASCII 码表中头五个字符 NUL、SOH、STX、ETX、EOT 的发音——译注），我更容易集中注意力。师父说，那个口诀意味着宇宙的“五原”。

不久我发现我不再关心是否能为工作成绩而得到奖励了,在我自身和我所维护的程序之间也看不见任何分离了。我像一个一生都活在阴影里的人一样,开始理解编程之禅,这是一种隐于程序设计背后的难以言喻和难以形容的力量,就像投射影子的太阳一样。

从那毫无意义的“自我的存在”中摆脱出来后,我开始发觉,那些伟大的编程语句过去对我来说似乎模糊不清,只是因为我还没有彻悟,无法理解它们。我现在知道了为什么过去的那些程序员从不为他们的程序提供文档,因为英语语言的描述本身更让人迷惑,而非使人受到启发。

一天,我发现自己正在解决的一个问题涉及到《编码藏经》中最复杂的部分——错误分析例程。我在对此一无所知的情况下,给出了一个修补程序,它能通过对硬件中断区的内容进行检查来测定错误的情况,使程序能继续正确执行。

那天下午,师父第一次走进了我的工作间,他把手放在我的肩膀上,眼睛朝下看着我,“现在该是你出师的时候了。”他说。

这些就是我第一次接触禅式程序设计的经历。虽然从那以后我被指派参加过很多项目设计,但我从未忘记过我的第一位师父给我的教诲。

想象一下我的惊奇——当我在《编程之禅》中发现了如此之多的我师父非常喜爱的言语时。我终于看到了那些隐于他那难忘的学说背后的古代传统。