# 课程个人作业

**————《没有银弹》读后感**

当我了解到《人月神话》作为软件工程两大圣经之一而被诸多软件工程师奉为圭臬之时，我便做好了怀着崇敬的心情进行拜读的心理准备。事实上，我必须承认:在逐字逐句地通读一遍全文后，我仍忘了大半。大部分词语我本看得懂，但是组合成句之后却让我一头雾水，充斥着各种专业术语的部分令我难以融入并参与作者的思考。我必须为自己开脱，可能有以下几点原因：

1. 也是主要原因。我没有任何完整的项目经验和管理经验（即便是如同短学期那样的mini项目也是不够系统全面的）。
2. 中文译本的翻译使得有些表述佶屈聱牙，令人莫名其妙不得其法。一些英文含义实际上是很难直译成我们所理解的中文意思，使得我觉得晦涩难懂。

虽然没有共鸣，但是“硬骨头”也让我啃掉了些许粉屑，使我得以勉强进行总结：

（1）软件的活动包括什么？

必要任务：打造抽象软件实体的复杂概念结构。

次要人物：用编程表达这些抽象实体，在空间和时间的限制下映射成机器语言。

（2）软件系统无法规避的内在特性？

复杂度，一致性，可变性和不可见性。

（3）解决次要问题的突破？

高级语言，分时，统一编程环境。

（4）银弹的希望在哪？

Ada和其他高级编程语言；

面向对象编程；

人工智能；

专家系统；

“自动”编程；

图形化编程；

程序验证；

环境和工具；

工作站；

(5) 针对概念上根本问题的颇具前途的方法

购买和自行开发；

需求精炼和快速原型；

增量开发——增长，而非搭建系统。

挑选和培养卓越概念设计人员；

而本章内容对我印象最深的启发在于：

增量开发中提到的：

***光是大脑本身，就比任何对它的描述都要复杂，比任何的模拟仿 真都要强大，它的多样性、自我保护和自我更新能力异常丰富和有力。其中的秘密就是逐步 发育成长，而不是一次性搭建。 所以，我们的软件系统也必须如此。很多年前，Harlan Mill 建议所有软件系统都应该 - 116 - 以增量的方式开发 11。即，首先系统应该能够运行，即使未完成任何有用功能，只能正确调 用一系列伪子系统。接着，系统一点一点被充实，子系统轮流被开发，或者是在更低的层次 调用程序、模块、子系统的占位符（伪程序）等。***

“没有银弹”的观点依旧令我保持怀疑——即便如此，难道也没有铜弹，钢弹，铅弹吗？为了探寻作者对于解决“根本问题”没有银弹的观点是一个怀疑论者还是悲观主义者，我继续往下读了作者的《再论没有银弹》。我对此章有两点感悟：

1. 如果开发的次要人物少于整个工作的9/10，即使不占用任何时间，也不会给生产率带来数量级的提高，所以必须解决开发的根本问题。
2. 可重用和可交互的构件开发是解决软件根本困难的一种方法。

作者在本章提到

***《人月神话》一文被大量地引用，很少存在异议；相比之下，《没有银弹》却引发了众 多的辩论，编辑收到了很多文章和信件，至今还在延续 。***

自1987年作者在《没有银弹》上断言:”***在未来十年之内，不会有任何单一的软件工程上的突破，能够让程序设计的生产力得到一个数量级上的提升。***”但事实上在2000年的《再论没有银弹》中，作者对当初的观点有了让步。

而在40周年纪念版的发行又过去的2018年；当提起文档不再是微缩胶片的2018年，《没有银弹》中已存在着部分过时的观点和例证，但仍不失为一章具有里程碑和启发式意义的论文。