“没有银子弹”读后感

所有软件活动包括根本任务——打造由抽象软件实体构成的复杂概念结构，次要任务——使用编程语言表达这些抽象实体，在空间和时间限制内将它们映射成机器语言。软件工程师花在次要任务的时间有可能较多，但是如果不提升根本任务的效率，缩短根本任务的时间才能给生产率带来数量级上的提高。

虽然我们希望寻求一种可以使软件成本像计算机硬件成本一样降低的方法，但是没有这种方法。进步是逐步取得的，没有一蹴而就的解决方法。相对于计算机硬件的过快发展，软件的发展速度要慢挺多。作者认为软件开发中困难的部分是规格化、设计和测试这些概念上的结构（数据集合、数据条目之间的关系、算法、功能调用等等），而不是对概念进行表达和对实现逼真程度进行验证。因此软件开发总是非常困难的，天生就没有银弹。

现代软件系统中有无法规避的内在特性，复杂度（规模、状态）、一致性（必须遵循各种接口）、可变性（遭受到持续的变更压力）和不可见性（不可见的和无法可视化的）。复杂度导致技术和管理的困难，很多复杂性来自保持与其他接口的一致，可变性包括很多功能是容易感受变更压力的部分，软件可以很容易地进行修改，无法可视化的固有特性则限制了个人的设计过程，也严重的阻碍了相互之间的交流。

以往解决次要困难的一些突破包括高级语言、分时、统一编程环境。现代解决次要困难的一些希望包括Ada和其他高级编程语言、面向对象编程、层次化类型、人工智能、“自动”编程、图形化编程、程序验证、环境和工具、工作站。但作者认为这些能起到的作用十分有限，并不能解决必要困难。解决根本问题的颇具前途的方法作者认为是购买和自行开发，需求精炼和快速原型。作者觉得现在的技术中最有希望的，并且解决了软件的根本而非次要问题的技术，是开发作为迭代需求过程的一部分——快速原型化系统的方法和工具。增量开发——增长，而非搭建系统，是在更低的层次调用程序、模块、子系统的占位符（伪程序）等，卓越的设计人员也是提高软件行业的核心。。