

MIS 软件质量控制中的几点体会

任连仲 薛万国 (解放军总医院计算机室)

MIS 设计的方法学和软件工程方法学问题,很多教科书和众多的报刊文章已经讲得很详细,有关的原则和方法也讲得很多,似乎没有什么可再书的。其实不然,设计方法学也好,软件工程方法学也好,都是实践性很强的科学,在设计和运行过程中采用不同的控制方法和控制策略,效果会有极大的差别。为什么在等同的时间期间内,在类似的环境之下,有的系统运行成功,有的失败;有的系统能很快可以由用户驾驭,而有的系统则是设计者不能离开,总得跟着维护,甚至必须亲自上阵作这种或那种处理;有的试用一段时间之后即可遍地开花,而有的则只能独立树一枝,不能移植;甚至更有有的报告文章写得冠冕堂皇,通过专家鉴定,报了成果,但实际没能运行下去,成了软件“垃圾”。当然,形成这种结果的原因是多方面的,但我们认为根本原因在于对软件质量含义的认识深度和在软件研制过程中的控制方法的得当与否。笔者从几个系统的设计和实际运行情况的比较,从几个成功和失败的经验教训中总结出了几点体会。

我们必须承认这样一个事实:在我国当前情况下, MIS 的使用者对计算机知识基本上是外行型或初学型的,这就给 MIS 设计者提出两点需认真考虑的问题:一是他们提出的需求只能是粗略的,更深入的和潜在的需求,要设计者去深入分析和体会,对用户在设计完成之后又提出这样那样的修改要求产生反感将是无济于事的;二是不可能期求使用者对你的软件设计有多少理解和谅解以及在定制好的功能菜单以外还能自主完成某些操作。这就要求我们必须需求分析和功能设计上,在软件本身的质量上狠下功夫,向软件商品化方向努力。

在这个努力过程中,我们的体会之一是:对人——机界面友好要做全面理解。人——机界面它包含着功能、环境、视觉、屏幕、语言、响应速度、操作简便等诸多方面,设计者需对每一方面都作认真地分析和仔细琢磨才有可能设计出让使用者感到方便、舒适、得心应手的应用软

件。从这一角度上说,一套成功的应用软件是艺术和技术的结合,是“爱心”和细心的产物。MIS 软件工程师完全把自己圈在计算机技术圈子里是很不利的,把界面只理解为屏幕和操作是不全面的。要特别注意用户的操作习惯及要求,用户需要增加或修改一点功能,对我们来说实现起来可能要付出正常功能 10 倍的努力,但这是值得的,这一修改可能给用户带来 100 倍的好处。因为修改是一时的,使用是长期的。

体会之二是:在设计一套 MIS 软件之前必须作好基础准备。这个准备工作大致包括:

- 运行整体环境的论证,诸如数据的日增加量、从数据存储周期而得出的年增加量、与此相应的 I/O 速率、最大和最小的响应时间等。

- 输入方法的准备。输入方法关系到系统是专门人员使用还是任何人均可使用的问题,可否多种输入方法兼顾以保持较宽适应面的问题,更直接关系到输入速度和准确度的问题,还关系到内存开销等问题。

- 标准名称及代码的准备。例如在 HIS 中少不了疾病、手术、检查、化验、单位、籍贯等固定名称(或标准名称)及其编码。一般来说,输入时按名称输入,而存储的只是代码,这就要求名称字典要完善。

其中第一项准备不必再说,几乎每个技术人员都会作的。第二项,多数人也比较认真考虑,只不过是所选方法灵巧与否,适应面宽和窄以及输入效率高与低的问题,总不会造成系统完全不能使用。而第三项准备工作,一般技术人员也还理解,但工作是否作得深入,是否真的在这方面花了大力气,则往往决定着系统能否投入实用。比如说,疾病、手术有标准名称字典,但医生没有标准字典名称怎么办?在收费系统中,有药品、检查、化验的标准名称字典,医生在处方中使用了标准名称以外的名称怎么办?硬找对应的规范名,时间又不允许。搞系统设计的同志不能简单地把这些问题完全推给用户或管理部门,必须在系统设计时一并予以认真考虑,通过技术人员

之手解决这些问题。也有的系统靠强硬的管理手段强行统一,使系统投入运行。但系统移到另外一家,又怎么办?到头来,还是不能作为通用产品推广。

我们在设计收费系统过程中,也有人提出就是要求医生写标准名称。但我们估计到,上百名门诊医生,每天开出数千张处方和检查单据,一律要求写出的药品和化验名称一字不差地完全符合“标准”名称将是不现实的。为了增强系统的适应性,放宽对处方和检查单据名称书写的要求,我们花了大力气搜集和整理了同一项目的各种别名和代号,并抽取出标准名中的关键词作为检索词。效果相当好,即使不熟悉收款业务的新上岗也能迅速完成划价。

在这一系统设计中单为这一种项目字典所花的时间就超过了系统的编程时间。

我们在这里所强调的是,建立一个实用信息系统,基础准备工作是大量的,工作是艰苦细微的。看起来这似乎不是技术工作,但它确实是一个优质实用的信息系统必不可少的,其作用丝毫不低于软件本身质量的作用。

我们的体会之三是把住两头——即总体设计、详细设计和软件测试。要想使一套软件交付使用之后,技术人员能很快脱开“保姆”地位,减少软件维护量,缩短软件试用期,就必须从根本上采取措施以确保软件质量。达到这一目标的关键步骤之一是:

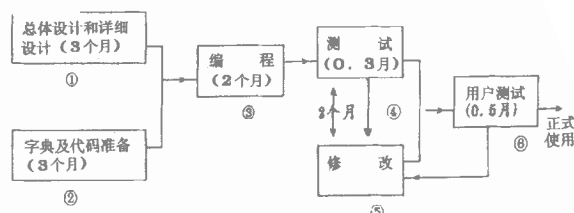
写好总体设计和详细设计,在设计没有完全审定之前决不能匆忙进入编程。总体设计和详细设计应包括如下方面:支撑环境设计、功能设计、工具选择、处理方法设计、安全措施设计、数据结构设计、数据项定义。在这个过程中技术人员最容易犯两方面的毛病:一是注重主要功能——如录/改/查的设计,而忽视维护功能和辅助功能设计;二是有个概要设计就急于编程。什么各种辅助功能、处理方法、安全措施,乃至数据结构和数据项的定义等等都在编程过程边走边想,而不肯花时间写出文字设计,先是自己反复推敲,再请有经验同志或技术主管反复审查。结果是急于求成而常常不成,或勉强用上之后必须在较长一段时间内跟着维护,脱离不开。一旦脱离便运行停顿。而在多次修补之后,连设计者自己也视它“不成样子”,于是又考虑改版了。事实上,写设计文档的过程就是综合考虑各种需求进行全面综合设计的过程,没有这一步,你的设计很可能是不完善的,还可能被

后续发现的问题所推翻。

应该抓住的另一头是软件工程中的重要一环——软件测试。软件测试也是一种技术,它要求设计得提供出全套设计文件和使用说明,测试者根据系统功能和使用要求设计出测试内容和测试方法以及测试用的数据,还要故意作错误的操作以检查软件的自身保护能力。测试——修改——再测试,有可能要几个反复。

技术测试之后最好还要再有一个用户测试过程。

我们以一个成功运行的收费系统为例。该系统达到了各类人员均能使用;实际试用一周以后设计者可以基本离开;试用期内基本没有发现软件上的问题。这套系统的研制大体遵守了上述过程和要求,那么各个环节占用时间比例如何呢?



整个软件研制时间大约 10.5 个人月,其中:

- 总体设计和详细设计月约占 2 / 5
- 名称字典准备约占 1 / 3
- 编程约占 1 / 5
- 测试约占 1 / 4

从图中看出编程时间只占 1 / 5, 似乎不可想像,但确实如此。设计者自己体会到,研制一套优质的应用软件,必须从设计到稳定运行的全过程着眼,从最终的效果着眼,把握住每个环节,特别是①②和④⑤这两头的环节。只要这样认真作了,软件人员常常头痛的一件事——软件维护将可缓解。我们认为,维护工作量将和设计质量和编程质量成反比。在一定程度上讲,维护工作量是可以控制的。

参考文献:

- (1) 人——计算机界面设计 [美] A. 苏其列夫著, 西安电子科技大学出版社。
- (2) 信息系统分析与设计 陈禹编 电子工业出版社。
- (3) 软件质量的鉴定 [美] B.W. 博姆等著 科学普及出版社。