　 计算机软件的可靠度和可维护性

　随着软件工程以及开发的复杂程度和系统完善程度的加深，逐步形成了相对较为科学的方式，对软件的可靠性、可维护性以及可懂性予以保障，同时生产率也逐步提高，效率提高的同时必然降低了成本。文章便主要研究分析了有关软件的维护方法，并针对其中的一些问题提出了相对具有建设性的建议，希望可以为软件可维护性的进一步发展有所助力。   
　　【关键词】可维护性 软件 方法   
　　作为硬件同用户之间的接口界面，计算机软件具有十分重要的作用。在进行系统设计的过程中，设计者会综合考虑用户的使用以及软件同硬件之间的结合，使得系统具有整体效用，通过全局性的把控，满足各类用户的需求。在整个生命周期中，软件技术的各个阶段所采用的技术以及管理都具有先进性，每个阶段在完成后都需要针对性的进行严格的审查、管理，只有在上一阶段工作合格之后才能进行下个环节的开发，这就确保了软件开发的质量，尤其在可维护性上得到了有效提高。   
　　1 维护性概述   
　　软件的周期主要有两个重要的阶段，一个是开发阶段另一个则是运行阶段，在运行过程中会发现实际上系统是存在很多的错误和漏洞的，软件的开发是一项大投资，在生产率提高、成本降低的同时，人们希望获得高品质的软件，从而快速的实现软件的应用和扩张。因此就需要在软件投入使用运行后，由软件人员进行进一步的修改，针对系统中存在的错误和漏洞进行修正，这就是我们常说的软件维护。   
　　常见的软件维护有三种：首先是纠正维护，主要针对软件中的错误进行修改；其次为完善维护，主要用于对软件的性能和功能予以扩大和提升；最后则是适应维护，该类工作则是为了提高软件对外部环境的适应性。   
　　2 一般维护方法   
　　2.1 模块化提升以及质量技术提升   
　　在开发软件时，对软件可维护性提高最有效的方式之一便是对开发质量的提高，这不但会对成本予以降低，同时也能够提高技术的有效性。功能模块的独立性是该种方式的优势，若某一模块需要更改，那么只需要对该模块进行修正，而不是整体性修改，不会对其他模块造成影响；若程序需要增添新的功能，那么只需要对该功能进行新模块的添加即可；而在测试重复以及程序时更加方便，很容易发现序列错误，并对其进行修正，从而提高了运行效率。   
　　2.2 品质的创建以及优先级的创建   
　　维护性对于程序来说应当是能够理解、修改、移植、使用并且可靠性以及效率较高的。上述要求的实现，需要付出相当大的代价，并且即便是付出了也不一定具有可行性。一些软件在质量特性上具有一定的互补性，例如可修改性、可理解性以及可理解和测试性等。但是在其他的特质上同样具有一定的矛盾性，例如可移植性同效率之间或者效率同可变化性之间。所以，软件在满足可维护性要求的同时应当根据程序的使用和环境的要求而做出适当的调整。   
　　2.3 设计语言的选择   
　　根据程序可维护性，选择程序设计语言，其影响是极大的。低层次的语言就是机器语言和汇编语言，这非常难以理解和掌握，也更难以对其进行维护。高级语言更容易理解，具有更好的可维护性。不管是什么语言，程序编制出来都很容易理解和修改，但存在指令数量可能会少一个数量级，而语言编制数量级要多一个，其开发速度会快多倍。   
　　3 可维护性加强措施   
　　在进行软件的维护过程中，会遇到诸多的问题，例如软件开发过程中人员变动较大，开发商的变动性也较大；由于文档资料的缺失，使得开发过程中对他人的开发体系无法全面的掌握；程序问题或者稳当文件的不恰当，软件结构出现问题，难以理解，无法对不恰当的地方予以修改，或者即便修改也容易出现问题。   
　　在企业计算机技术应用中，领域管理应用较为广泛，程序深入程度更胜一筹。管理程序主要可以从两方面进行分析，一个是信息化管理，一个则是工业控制。引入自我维护是软件设计执行上的重大进步，该项功能提高了软件自身的活力。但是在设计上，该项自动维护功能还是具有一定难度的，其运行会占用系统资源。但是计算机技术在发展，并且时至今日，计算机资源已经十分充盈，自我维护功能所占用的量完全可以为系统所承受，不会影响系统运行，综合考虑，其功能的应用是值得的。而在其他软件的开发中，将该种自我维护技术加以应用，全面结合系统的个性、共性，那么对于软件技术的发展将会是一种更高层次的进步。   
　　4 结束语   
　　通过上述内容可以分析出，目前国民经济中软件产业占据了十分广泛的空间，无论是生产还是生活学习中，到处都有计算机软件的存在，计算机信息系统的灵魂便是计算机软件，这是计算机技术得以发展应用的基础，并且软件技术开始重视信息系统的新型发展方向，逐步的实现集成化、多元化，使得结构分布更为广泛。在实践中完成一系列的创新，可维护性是软件在这一阶段主要形成的重要方面，所以，在进行软件开发的各个环节中都必须对可维护性进行把控，使之能够贯穿始终。将软件整个周期进行全面的掌握，能够有效提高软件所具有的可维护性，同时也能够促进软件开发的效率，对维护人员的维护工作也具有一定的意义。