

分层技术在计算机软件开发中的应用

闻丹丹

河南翔宇医疗设备股份有限公司 河南 郑州 450000

摘要：计算机软件能够转变社会大众的行为习惯与思维动态，创新信息传播的形态及发展趋势，使人们通过计算机软件获得更好、更优质的现代化服务。然而在计算机软件开发与应用的过程中，传统的计算机开发技术已经难以适应新时代的发展史诉求，无法通过科技创新的方式，助力我国社会主义市场经济的健康发展。而将分层技术有机地融入计算机软件开发中，可以切实地提升计算机开发的实效性和有效性，帮助我国更好地利用信息化、数字化技术，推动社会经济的发展进程。

关键词：计算机工程；软件开发；分层技术

1 分层技术应用于计算机软件开发核心概念

层次关系不仅是电脑软件的物理含义，也是电脑领域的特殊含义，它的含义要比物理意义更加复杂。电脑软件发展的层次观念就是将各种不同的处理方法排列在不同的概念层次，再由整体组成一套体系，每一层次都是平等的。在一定程度上，软件开发体系结构可以从一定程度上反映出应用于软件开发中的分层技术。随着计算机的不断发展，小规模数据库也得到了广泛的应用。

由于分层技术在计算机软件开发中的诸多优点，所以计算机软件技术的发展逐步转向了层次技术。计算机软件的研制目的是为了使软件的品质达到用户的要求。在电脑软件的发展中，使用者可以利用已经过严格测试的元件来建立电脑软件，从而逐渐缩短工作时间，提升效能。这个系统也保证了所有的软件都是高品质和高品质的^[1]。通过采用多个层次的构件，实现了软件与软件之间的相互联系，进而进一步改善软件系统的效能。在通用的电脑软件系统中，没有非常严密的软件架构。多层次电脑软件会透过大量的分析，引发电脑软件与软件的相互依存。他们基本上是在很多层面上装配起来的，只是在微观层面上，层级间的关系是内在的。广泛的应用层次化技术可以极大地促进整个软件系统的抽象和从一个复杂的部分到一个软件的设计。此外，若将多层次技术应用于电脑软件开发，可使之具有较好的稳定性及可扩展性

2 分层技术的发展及优势

2.1 分层技术发展历程

分层技术是应用于现在软件开发行业中比较广泛的技术，一般来说，很多计算机在出厂以后，硬件都是配置好的，在以后的更新换代后，基本都是依靠软件更新，所以软件有着不可替代的作用。这就要求软件的开

发要更精确，在计算机快速发展的今天，分层技术也变更了好几代发展模式。从上个世纪八十年代开始，在编构小型化数据库的时候就已经出现了分层技术，到后来的客户端计算机的广泛普及，分层技术又发展成了双层结构，单一的结构不能满足用户的实际需求，逐渐被双层技术所取代，再到后来双层的结构也无法应对未来的发展趋势，三层技术又取代了前两层的结构^[2]。

2.2 分层技术应用优势

软件开发的模式有很多种，比如，耳熟能详的边做边改模式、瀑布模式、迭代模式还有螺旋模式等，据不完全统计大约有十一种。分层技术相对于这十一种开发技术来说优点最为明显，也是标志着日后软件开发模式的行业标杆，就像CAD在工程制图领域的地位。软件开发的目的是要人机交互变得简单方便，试想一个软件开发出来，在使用过程中BUG频出，对于谁来说都是一个不好的体验。分层技术就可以完全适配现在的开发模式，满足用户的软件使用需求。它不仅能够扩展相关的计算机软件，还能够分析其中的源代码升级和改造计算机软件，相当于从底层的重新构建。这不仅对于开发人员起到一个辅助作用，更能够提高开发人员的开发效率，软件的运行也可以通过这种开发模式得到很好的保障，减少BUG的出现，使用得当还可以改善计算机的生态模式系统。一个软件从开发到应用的期间有长有短，特别是对于软件开发中的算法而言，运用分层技术能够有效加快算法的速度，缩短开发时间，提高开发效率。

3 计算机软件分层技术的特点

在软件开发中广泛使用的分层技术具有独特的优势。计算机可以根据其功能特性分解软件开发中最为繁杂的部分^[3]。这使得创建计算机软件变得容易，并且便于软件升级和扩展。软件开发分层技术可以在很大程度上

提升软件开发的效率,大大缩减软件开发的时间和改进软件的性能,保障软件的开发工作能够完满地运行,并且可以基于不同的功能水平进行分类和改进。但是,通过代码的重用和易于理解的开发模型也可以保证软件质量。因此,在计算机软件的开发中使用多层技术非常有效,该系统可用于定义功能和与标准接口和端口的不间断通信。在较深的层次上,分层技术开发软件具有以下四个特征:

3.1 分层技术允许将计算机软件扩展到基于功能模块的简单系统,并将复杂软件分解为简单系统。

3.2 计算机软件开发的层次结构可以提高开发效果,缩短开发时间,在更深层次上提升计算机在软件领域以及自动化领域开发的质量。

3.3 分层技术具有双重结构这一显著的优势。在计算机的性能和功能层面上,它能够在很大程度上改善计算机的总体性能,从而改善开发商的工作量,因为软件开发是计算机本身的一个基本部分。

3.4 从计算机软件系统的角度上来说,分层技术同计算机的硬件紧密相关,这有助于提高计算机软件的性能。

4 分层技术在计算机软件开发中的应用分析

多层技术是从单层技术演变而来,为了提升计算机软件的开发水平,软件开发人员开始逐步使用双层技术、三层技术等分层开发技术,在缩短计算机软件开发时间的同时,使软件开发更具效率^[1]。在分层技术的帮助下,软件功能得到了拓展,并且也为软件未来更多功能的嵌入提供了技术支持。

4.1 双层技术的应用

在早期软件开发工作中,单层技术的应用十分普遍。然而,随着社会对软件功能要求的不断提升,单层技术由于技术层面本身的限制,已经无法迎合时代发展要求。因此,为了提升计算机软件开发水平,推出了双层技术,取得了良好的效果。双层技术的应用,可以使软件的运行更加稳定,提升了计算机的运行速度。将双层技术开发的软件应用到客户端与服务端,可增强服务器分析、处理用户信息请求的速度,提升服务器的工作效率。双层技术在客户端软件中的应用,可以使界面设计更加人性化,逻辑处理效率得到进一步提升。然而,随着连接用户数量的不断提升,服务端的负载会增加,从而导致软件的启动运行速度下降,如果荷载进一步增加,将会引发报错问题。为了提升双层技术运行的稳定性,需要对用户数量进行限制,并更新硬件设备提升服务器的性能,具体为:(1)缩减同一时间连接服务器端的用户数量,使服务器中的用户数量控制在要求范围,

避免服务器荷载过大,影响服务器的服务质量。(2)提升服务器的性能,增加服务器的容量,确保服务器的性能可以得到最大限度地发挥。尽管双层技术存在诸多优势,但是也存在一些弊端,需要使用三层技术弥补双层技术的不足^[2]。

4.2 三层技术的应用

三层技术属于双层技术的延伸与拓展,简单来说就是双层技术的优化和升级,在双层技术上进行了服务器端口的添加,在极大程度上,促进了计算机系统工作效率和能力的提升。服务器层次应用过程中发挥了处理和分担用户端逻辑关系的功能,其应用过程中可提供用户需求的界面,有利于彰显人机交互的应用优势。分层结构主要由数据层、处理层的界面层三种结构组成,可对用户实际需求进行广泛收集,并将其向处理层发送,这属于界面层的主要功能,业务处理层能够依据用户实际需求,开展用户需求的定向分析,并将分析后的最终数据向数据层发放,数据层依靠申请形式对信息进行查询分析,并锁定相关信息向处理层中投放,将信息整理完成后,由界面层向用户提供反馈。这三个层面的有机分工可缩减信息传输时间,提升信息数据分析效率,促进端口工作效益的提升,三层技术相对二层技术来说拥有明显优势,但是在实际开发过程中,必须依据软件系统的需求进行分析并合理选择应用技术层次,实现节约成本投入的目的。

4.3 四层技术的应用

从整体比较来说,四级分层技术与三层分层技术的相似度很高,这主要是以为四层分层技术原本就是基于三层分层技术作出的深入研究和最终优化成果,其中经历了对三层分层技术的多次调整、改进与升级,填补了以前存在的漏洞。应用四层分层技术的第一步还是整合处理数据的操作,完成数据处理后将结果传递到Web层,与以往的数据系统相比,Web层的功能性更强、更透彻。

Web层中对于数据的处理非常高效,在接收数据的第一时间就要进行详细分析,通过分析判断其与计算机系统的适配性,基于这些数据分析结果和计算机系统之间的联系,精准的找出有数据差异的地方,从而采取一系列的调整、优化、修改措施处理差异性数据。

由于Web层具有深度且有效融合多个不同层次的功能,同时还能保证各个层次原本的独立性,普通接触都是在层面接口处完成的,这种接触模式在实践过程中出现不兼容现象的概率很高,因此有效解决信息数据差异性问题的最好方式就是通过Web层功能推动信息传递^[3]。

4.4 层技术的应用

当前,在某些领域或者专门的条件下,计算机软件开发人员通过四层技术将数据层分为集成层与资源层的五层结构,进而提高开发系统的运行成效,同样能够基于现实效应,以达到特殊装置的运行标准,虽然对五层技术架构实行了较为精细的分工,却也无法消除该技术在当前时期无法进行有效推广的问题,所以只有在部分专门的行业获得运用。

结束语

为充分满足用户使用需求,为用户提供更为优质的软件服务,有效应对复杂的网络问题与运行环境。因此,企业机构、开发人员必须提高对计算机软件开发分

层技术的应用力度,根据软件开发需求灵活运用各项具体技术,充分发挥技术优势。同时,还需加强技术研发力度,对分层技术体系与理论基础进行完善补充、优化创新。

参考文献

- [1]姬晓鹏.分层技术在计算机软件开发中的应用研究[J].电子测试,2020(18).
- [2]黎乾坤.关于分层技术在计算机软件中的应用研究[J].科技传播,2020,12(14).
- [3]苏越.分层技术在计算机软件开发中的应用研究[J].信息与电脑(理论版),2020,32(14).