

基于 MVVM 架构的移动 Web 前端展示方案

封宇^{1,2}, 陈宁江¹

(1. 广西大学计算机与电子信息学院, 广西 南宁 530004;

2. 广西职业技术学院计算机技术与电子信息工程系, 广西 南宁 530226)

摘要:为解决移动终端用户个性化丰富体验的问题,本文分析当前基于 MVC 和 MVP 的移动 Web 开发框架,针对这 2 种模式在移动终端应用时展示逻辑没有完成与业务逻辑分离的问题,提出引入 MVVM 架构解决展示逻辑与业务逻辑完成分离的问题,为用户在移动终端上提供个性化体验。实例表明,使用基于 MVVM 架构设计的移动终端应用系统,能有效地实现展示逻辑与业务逻辑完成分离。

关键词:移动终端; MVVM; 个性化; 架构; 移动 Web

中图分类号:TP39 **文献标识码:**A **doi:** 10.3969/j.issn.1006-2475.2014.11.001

Scheme of Mobile Web Front-end Display Based on MVVM Framework

FENG Yu^{1,2}, CHEN Ning-jiang¹

(1. College of Computer and Electronic Information, Guangxi University, Nanning 530004, China;

2. Department of Computer Technology and Electronic Information Engineering,
Guangxi Vocational Technical College, Nanning 530226, China)

Abstract: To solve the problems of the mobile terminal user personalization rich experience, based on the MVC and MVP mobile Web development framework are analyzed, according to problems of show logic and business logic when using the MVC and the MVP to develop the mobile terminal applications, the MVVM architecture is proposed, and it can provide the personalized experience on the mobile terminals. Examples shows that using MVVM based architecture to design the mobile terminal applications can effectively achieve the complete separation of show logic and business logic.

Key words: mobile; MVVM; personalization; framework; mobile Web

0 引言

随着移动互联网技术的迅速发展,以及移动智能终端快速普及,人们获取信息的途径从 PC 端 Web 浏览逐渐转向移动智能终端的 Web 浏览。虽然获取信息的途径改变,但用户依然追求着个性化丰富体验的用户界面,如界面的个性化布局、多窗体协同工作等功能。为了适应用户的要求,开发框架一方面应具有“低耦合高内聚”^[1],另一方面要求业务逻辑层、数据逻辑层和 UI 逻辑层分离,使呈现界面和数据逻辑可以同步进行^[2]。目前,基于 MVC 和 MVP 的移动 Web

开发框架^[3]受自身架构的影响,并没有实现把 UI 逻辑层完全分离出来,因此无法实现用户个性化丰富体验的用户界面。MVVM 架构是微软 WPF 和 Silverlight 架构师 John Gossman 于 2005 年提出的^[4],充分实现最大程度的数据业务逻辑与界面分离,并在 PC 端开发 RIA 应用系统,为用户提供了丰富体验的用户界面^[5-7]。移动互联网中的移动终端由于内存空间小、运算速度慢、操作系统不统一、屏幕尺寸大小不一、网络间歇性等原因,WPF 和 Silverlight 并不适用于移动互联网,因此,本文借鉴 MVVM 架构在 WPF 和 Silverlight 上的运用策略,研究基于 MVVM 架构的

收稿日期:2014-08-08

基金项目:国家自然科学基金资助项目(61063012);广西自然科学基金资助项目(2012GXNSFAA053222);广西高校优秀人才资助计划([2011]40);广西科学研究与技术开发计划项目(桂科软 13180015)

作者简介:封宇(1982-),男,广西梧州人,广西大学计算机与电子信息学院硕士研究生,广西职业技术学院计算机技术与电子信息工程系讲师,研究方向:软件工程,移动应用;陈宁江(1975-),男,广西南宁人,教授,博士,研究方向:软件工程,网络分布式计算机。

移动 Web 前端展示系统,其具有扩展性、个性化功能等特性。

1 MVVM 架构

MVVM 架构^[8]分别由 Model(模型层)、View(视图层)和 ViewModel(视图模型)3 部分构成,其结构如图 1 所示。View 层根据要展示的应用类型和数据格式的不同生成不同窗体视图的动态组装模型,仅需负责数据的展现逻辑^[9]。ViewModel 实现展现逻辑,它只关心用户操作时的任务、操作流程、操作逻辑行为,操作运行控制^[10]。Model 层一方面实现业务逻辑,制定业务规则,另一方面对业务数据封装,实施业务制定的规则,保证数据的有效性和一致性^[11]。

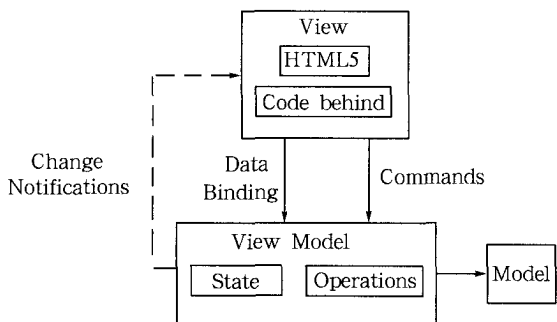


图1 标准 MVVM 架构

该架构运行策略为 Model 层的业务逻辑在服务器上实施,View 层与 ViewModel 层在用户端实施,ViewModel 监控着 View 的状态,并从 Model 根据用户需要获取数据,同时,把 View 的 DataContext 属性设置为 ViewModel 的实例,ViewModel 通过数据双向绑定或命令的方式与 View 控件的属性实现数据的更新,通过 ViewModel 解决 View 与 Model 之间数据的传递与交换,也保证 View 层和 Model 层的数据同步。

从上述得知,该架构展示逻辑与业务逻辑是完成分离,它们之间的数据传递都是通过 ViewModel 解决。对于目前移动互联网中的移动终端运算速度慢、内存小的特点,该架构非常适用于本文的移动终端 Web 应用开发,在用户的移动终端只负责数据的展示逻辑,大大减小移动终端运算的开销。由于展示逻辑与业务逻辑的分离,因此,在该架构具体实施时,可以为用户提供个性化界面设置的功能,用户操作界面布局时,不会影响业务逻辑的正常工作,使该架构的 View 层具有个性化。同时,服务器端的业务逻辑的改变或扩展,也不会影响展示逻辑,使该架构的 Model 层也具有扩展性。

2 基于 MVVM 架构的移动 Web 前端展示系统

2.1 运行策略

目前,MVVM 架构在 PC 端开发 RIA 应用系统得到很好的体现,该架构也适合移动终端应用开发使用,因此本文借鉴它的运行策略,在移动终端开发实施该架构,关键解决各层数据传递的策略。

本文借鉴思路:Model 层的业务逻辑仍然在服务器上实施,View 层仍然在用户移动终端实施。各层数据传递策略使用观察者模式和中介者模式解决,View 层作为观察者,Model 层作为被观察者,在它们两者之间加入一个中介者 ViewModel 层,帮助解决观察者与被观察者之间比较复杂的联系,以保证展示逻辑与业务逻辑的完全分离。

2.2 扩展 MVVM 架构

目前,在移动互联网中,用户使用的移动终端类型众多,未来也会涌现更多新的移动终端类型,同时,用户追求着个性化丰富体验^[12]。要求该系统应具有适应当前以及今后众多的移动终端类型的跨平台性,使用户移动终端提供获取第三方数据的扩展性,为用户移动终端提供个性化布局、多窗体协同工作等个性化功能。

标准的 MVVM 架构基本满足本系统的需求,但是,对于数据源来说,Model 层的业务逻辑仅局限于处理相应原生系统的 Data Service/Web Service 中的数据,而无法处理从其他系统或平台获取的第三方数据,导致信息孤岛,使用户体验有所欠缺,从而无法突出 MVVM 架构的低耦合性和高重用性的特点。因此,对标准 MVVM 架构进行扩展,在 Model 层添加对外部业务逻辑处理模块(External Business Logic),如图 2 所示,支持获取其他系统或平台第三方数据,提供在用户移动终端上进行展现。

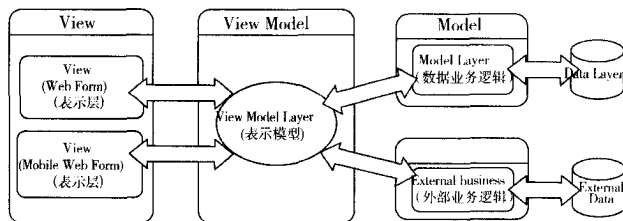


图2 扩展 MVVM 架构

为了使本系统易维护、易扩展,外部业务逻辑处理模块与 Model 层的业务逻辑相对独立。Model 层的业务逻辑只负责处理相应原生系统的 Data Service/

Web Service 中的数据,为 View Model 层提供原生系统服务器中的数据,而外部业务逻辑处理模块处理从第三方获取的数据。该模块业务处理的工作,依托服务器强大的计算能力,用户通过移动终端输入第三方数据所在的服务器 IP 地址,该模块根据 IP 地址,抓取第三方数据,并把第三方数据在服务器端处理加工,按 ViewModel 层的数据模型规则封装数据,再传送到 ViewModel 层,供用户移动终端展现。

标准 MVVM 架构与本文扩展 MVVM 架构详细的对比,如表 1 所示。

表 1 标准 MVVM 架构与扩展 MVVM 架构对比表

对比项	标准 MVVM 架构 (针对 PC 终端)	扩展 MVVM 架构 (针对移动终端)
Model 层	处理相应原生系统的数据	处理相应原生系统的数据,并增加对外部业务逻辑处理模块
Model	对服务器数据进行封装	对服务器与第三方数据进行封装
外部业务逻辑处理模块	无	对客户自发所需的第三方数据进行业务处理,并对数据封装
资源存储位置	服务器	服务器与客户端(离线存储必要数据)
View 层	屏幕大小,方向基本统一	适应移动终端屏幕各尺寸与各方向,个性化操作界面
支持平台	只支持 Windows	支持安卓、苹果等移动终端操作平台

2.3 系统的 MVVM 架构实现类图

根据 MVVM 架构思想和移动终端的特点进行设计,并把扩展模块融入到 MVVM 架构中,制定实现系统的 MVVM 架构类结构图,如图 3 所示。

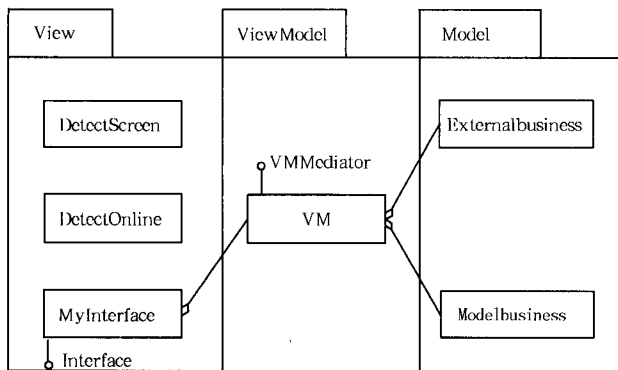


图 3 MVVM 架构实现类图

在图 3 可以看出,View 层中的类对应着展示逻辑,由于移动终端运算速度慢,用户移动终端只负责展示逻辑,为用户提供良好的体验。Model 层中的类是对应着业务逻辑,业务逻辑运算由服务器完成。展

示逻辑与业务逻辑是完全分离,它们之间并没有直接联系,而是通过 ViewModel 层中的 VM 类进行联系,保证数据的一致性。从而使得,用户在移动终端操作展示逻辑相关的类,或在移动终端界面进行个性化界面设置时,并没有影响业务逻辑。同时,业务逻辑的改变或运算时,用户移动终端的展示逻辑也不会发生变化。当服务器上数据发生变化时,业务逻辑通过 VM 类,通知用户移动终端的展示逻辑,展示逻辑会自动更新数据,VM 类起到中介桥梁作用。展示逻辑与业务逻辑之间的间接联系,可以使用观察者模式解决,而它们之间的间接联系比较复杂,可以使用中介者模式,把 VM 类作为它们之间的中介者。

3 应用实例

在广西职业技术学院的通识课学习平台中对上述方案进行了应用。该平台需要在用户的移动终端进行数据展示,并为用户提高个性化服务,提高用户学习兴趣,为用户提供良好的使用体验。该系统采用中介者^[16]和观察者设计模式^[17]实现 MVVM 架构中各层数据之间的传递,保证各层的数据同步,同时,View(视图层)与 Model(模型层)完成分离,提供用户可以制定个性化界面,用户在设置个性化界面时,只操作 View 层的逻辑,与 Model 层的逻辑无关。如图 4 所示。

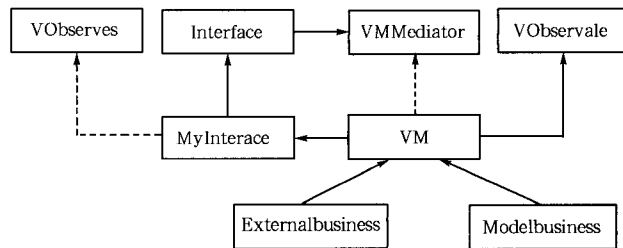


图 4 该系统核心类图

将用户界面定义为观察者 MyInterface 类,它实现 VObserves 接口,并继承 Interface 类。在用户界面的不同 View 与 ViewModel 之间传递参数时,定义一个公共的接口 VMediator,通过该接口,生成中介者 VM 类,用于实现不同 View 或不同 ViewModel 之间传递参数。同时,对 VM 类扩展,实现 VObservable 类,作为观察者模式中的被观察者,Externalbusiness 类与 Modelbusiness 类聚合在 VM 类中,观察者 MyInterface 类观察监控 VM 类。当 Externalbusiness 或 Modelbusiness 的状态有所改变时,立即通知所有 MyInterface 类,并自动更新数据。

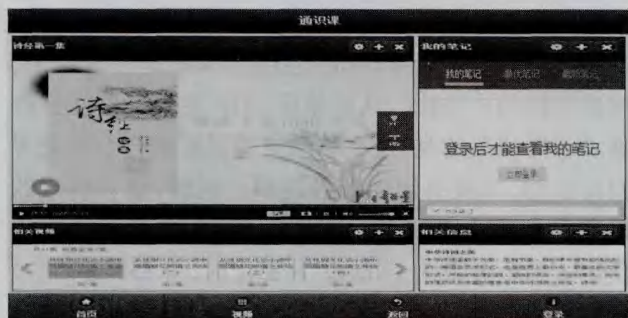


图5 Android平台运行效果图



图6 IOS平台运行效果图

学习平台运用本系统作为移动终端前端显示时, 基于IOS和Android平台的移动终端都得到良好的支持。在图5中, 每个模块的右上角有3个功能按钮, 分别为: 配置、添加和删除。移动终端的用户可以根据个人喜好, 调整各个模块的布局, 并对模块进行管理配置展示的内容, 可以对模块进行添加或删除。在图6中, 该前端展示系统, 根据移动终端屏幕尺寸、屏幕方向进行展示页面的最优化调整, 并没有影响用户体验。用户根据个人喜好, 调整了显示页面布局, 并把模块的功能按钮, 以文字方式替换。该系统在移动终端上只负责数据的展示逻辑, 符合移动终端的特点, 良好地体现了该系统的用户个性化操作功能。用户还可以通过该系统的兴趣信息模块对感兴趣的信息进行拓展, 在该模块中输入相关的网址或关键字, 系统会自动抓取信息, 并在用户的移动终端展示。

4 结束语

本文根据移动终端的特点, 论述了MVVM架构

在移动终端开发中的优缺点, 并对该架构的缺点进行改进, 扩展MVVM架构。开发基于MVVM的移动Web前端展示系统, 并在我院的通识课学习平台上使用, 取得了良好的效果。下一步笔者将加强该系统对复杂数据结构的处理效率等问题, 提高系统性能。

参考文献:

- [1] 李茂, 贾宏, 李艳明, 等. 云数据管理的新数据模型[J]. 计算机工程与设计, 2012, 33(10): 3870-3876.
- [2] 程国雄, 胡世清. 基于SilverLight的RIA系统架构与设计模式研究[J]. 计算机工程与设计, 2010, 31(8): 1706-1709, 1713.
- [3] Addy O. Learning Java Script Design Patterns[M]. Britain. O'Reilly Media, 2012.
- [4] Raffaele Garofalo. Building Enterprise Applications with Windows R R Presentation Foundation and the Mode View View-Model Pattern[M]. Targtted News Service, 2011.
- [5] 张姝, 周林, 付立新. 基于富客户端技术的心语墙系统的设计与实现[J]. 计算机科学, 2012, 39(10): 94-96.
- [6] 张艺宝. 基于富客户端的门户网站的开发与实现[D]. 长春: 吉林大学, 2012.
- [7] 陆玉刚, 邱知, 游先祥, 等. 基于SilverLight和REST的富网络地理信息系统框架设计[J]. 地球信息科学学报, 2012, 14(2): 192-198.
- [8] 陈明, 李猛坤, 张强. 一种基于扩展MVVM模式的SaaS面向服务计算模型[J]. 微电子学与计算机, 2010, 27(8): 27-30.
- [9] 刘立. MVVM模式分析与应用[J]. 微型电脑应用, 2012, 28(12): 57-60.
- [10] 罗智勇, 罗娟, 赖德军. RIA技术方案下的插件式WebGIS系统架构设计[J]. 测绘科学, 2012, 37(6): 160-162.
- [11] 蒲哲, 朱名日. 基于WPF MVVM的甘蔗种植管理系统[J]. 计算机与现代化, 2014(2): 110-114, 177.
- [12] 黄有福. 基于智能移动终端的学习平台的设计与实现[D]. 广州: 华南理工大学, 2012.
- [13] Jay Bryant, Mike Jones. Pro HTML5 Performance[M]. Springer Press, 2012.
- [14] 林闯, 董扬威, 单志广. 基于DTN的空间网络互联服务研究综述[J]. 计算机研究与发展, 2014, 51(5): 931-943.
- [15] 吴良波, 金连甫. 离线应用本地数据存储设计与实现[J]. 计算机工程与设计, 2010, 31(6): 1236-1239.
- [16] 路游, 纪连恩. 一种群组行为模拟绘制系统软件框架设计方法[J]. 计算机技术与发展, 2013, 23(4): 79-82, 86.
- [17] 尹禄高, 刘旺开, 沈为群. 基于组件与设计模式的测控系统[J]. 计算机测量与控制, 2010, 18(2): 360-362.