**项目管理员账号：admin 密码：asd123**

**项目技术开发文档**

**1.开发技术说明：**

本系统前端部分基于MVVM模式进行开发，采用B/S模式，后端部分基于Java的ssm框架进行开发。

前端部分：前端框架采用了比较流行的渐进式JavaScript框架Vue.js。使用Vue-Router和Vuex实现动态路由和全局状态管理,Ajax实现前后端通信，Element UI组件库使页面快速成型，项目前端通过栅格布局实现响应式，可适应PC端、平板端、手机端等不同屏幕大小尺寸的完美布局展示。

后端部分：采用ssm作为开发框架，同时集成MyBatis、Redis等相关技术。

**1.1 MVVM模式介绍：**

MVVM是Model-View-ViewModel的简写。它本质上就是MVC 的改进版。MVVM 就是将其中的View 的状态和行为抽象化，让我们将视图 UI 和业务逻辑分开。当然这些事 ViewModel 已经帮我们做了，它可以取出 Model 的数据同时帮忙处理 View 中由于需要展示内容而涉及的业务逻辑。微软的WPF带来了新的技术体验，如Silverlight、[音频](https://baike.baidu.com/item/%E9%9F%B3%E9%A2%91/1197465" \t "https://baike.baidu.com/item/MVVM/_blank)、[视频](https://baike.baidu.com/item/%E8%A7%86%E9%A2%91/321962" \t "https://baike.baidu.com/item/MVVM/_blank)、[3D](https://baike.baidu.com/item/3D/25017" \t "https://baike.baidu.com/item/MVVM/_blank)、[动画](https://baike.baidu.com/item/%E5%8A%A8%E7%94%BB/206564" \t "https://baike.baidu.com/item/MVVM/_blank)……，这导致了软件UI层更加细节化、可定制化。同时，在技术层面，WPF也带来了 诸如Binding、Dependency Property、Routed Events、Command、DataTemplate、ControlTemplate等新特性。MVVM（Model-View-ViewModel）框架的由来便是MVP（Model-View-Presenter）[模式](https://baike.baidu.com/item/%E6%A8%A1%E5%BC%8F/700029" \t "https://baike.baidu.com/item/MVVM/_blank)与WPF结合的应用方式时发展演变过来的一种新型架构[框架](https://baike.baidu.com/item/%E6%A1%86%E6%9E%B6/1212667" \t "https://baike.baidu.com/item/MVVM/_blank)。它立足于原有MVP框架并且把WPF的新特性糅合进去，以应对客户日益复杂的需求变化。

**1.2 MVVM模式优点：**

MVVM模式和MVC模式一样，主要目的是分离[视图](https://baike.baidu.com/item/%E8%A7%86%E5%9B%BE" \t "https://baike.baidu.com/item/MVVM/_blank)（View）和模型（Model），有几大优点

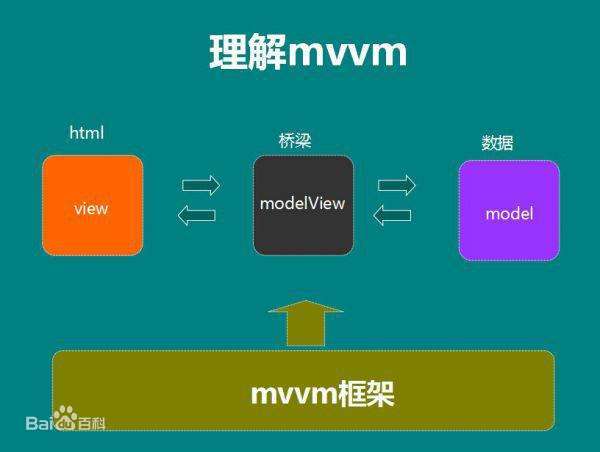
1.2.1. 低耦合。视图（View）可以独立于Model变化和修改，一个ViewModel可以绑定到不同的"View"上，当View变化的时候Model可以不变，当Model变化的时候View也可以不变。

1.2.2. 可重用性。你可以把一些视图逻辑放在一个ViewModel里面，让很多view重用这段视图逻辑。

1.2.3. 独立开发。开发人员可以专注于业务逻辑和数据的开发（ViewModel），设计人员可以专注于页面设计，使用Expression Blend可以很容易设计界面并生成xaml代码。

1.2.4. 可测试。界面素来是比较难于测试的，测试可以针对ViewModel来写。

**1.3 MVVM模式图解：**



**1.4 B/S体系结构介绍：**

B/S结构（Browser/Server，浏览器/服务器模式），是WEB兴起后的一种网络结构模式，[WEB](https://baike.baidu.com/item/WEB/150564" \t "https://baike.baidu.com/item/_blank)浏览器是客户端最主要的应用软件。这种模式统一了客户端，将系统功能实现的核心部分集中到[服务器](https://baike.baidu.com/item/%E6%9C%8D%E5%8A%A1%E5%99%A8/100571" \t "https://baike.baidu.com/item/_blank)上，简化了系统的开发、维护和使用。客户机上只要安装一个浏览器，如[Chrome](https://baike.baidu.com/item/Chrome/5633839" \t "https://baike.baidu.com/item/_blank)、[Safari](https://baike.baidu.com/item/Safari/597" \t "https://baike.baidu.com/item/_blank)、[Microsoft Edge](https://baike.baidu.com/item/Microsoft Edge/17511966" \t "https://baike.baidu.com/item/_blank)、[Netscape Navigator](https://baike.baidu.com/item/Netscape Navigator/1014148" \t "https://baike.baidu.com/item/_blank)或[Internet Explorer](https://baike.baidu.com/item/Internet Explorer/1537769" \t "https://baike.baidu.com/item/_blank)，服务器安装[SQL Server](https://baike.baidu.com/item/SQL Server/245994" \t "https://baike.baidu.com/item/_blank)、[Oracle](https://baike.baidu.com/item/Oracle/301207" \t "https://baike.baidu.com/item/_blank)、[MYSQL](https://baike.baidu.com/item/MYSQL/471251" \t "https://baike.baidu.com/item/_blank)等数据库。浏览器通过[Web Server](https://baike.baidu.com/item/Web Server/9306055" \t "https://baike.baidu.com/item/_blank)同数据库进行数据交互。

**1.5 B/S体系工作原理：**

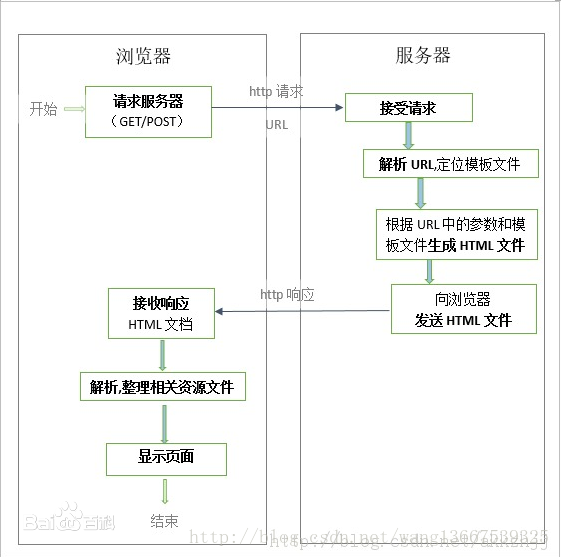
B/S架构采取浏览器请求，服务器响应的工作模式。

用户可以通过浏览器去访问Internet上由Web服务器产生的文本、数据、图片、动画、视频点播和声音等信息；

而每一个Web服务器又可以通过各种方式与数据库服务器连接，大量的数据实际存放在数据库服务器中；

从[Web服务](https://baike.baidu.com/item/Web%E6%9C%8D%E5%8A%A1/2837593" \t "https://baike.baidu.com/item/_blank)器上下载程序到本地来执行，在下载过程中若遇到与数据库有关的指令，由Web服务器交给数据库服务器来解释执行，并返回给Web服务器，Web服务器又返回给用户。在这种结构中，将许许多多的网连接到一块，形成一个巨大的网，即全球网。而各个企业可以在此结构的基础上建立自己的Internet。

在 B/S 模式中，用户是通过浏览器针对许多分布于网络上的服务器进行请求访问的，浏览器的请求通过服务器进行处理，并将处理结果以及相应的信息返回给浏览器，其他的数据加工、请求全部都是由Web Server完成的。通过该框架结构以及植入于操作系统内部的浏览器，该结构已经成为了当今软件应用的主流结构模式。

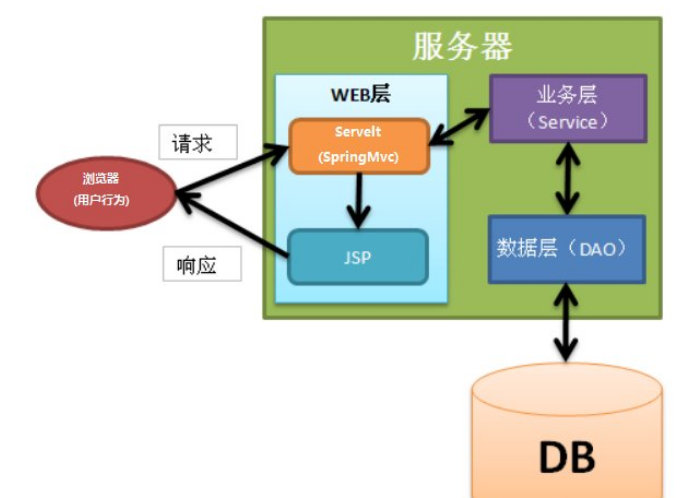


**1.6 ssm框架介绍：**

SSM（Spring+SpringMVC+MyBatis）框架集由Spring、MyBatis两个开源框架整合而成（SpringMVC是Spring中的部分内容）。常作为数据源较简单的web项目的框架。

1.6.1 Spring  
　　Spring就像是整个项目中装配bean的大工厂，在配置文件中可以指定使用特定的参数去调用实体类的构造方法来实例化对象。也可以称之为项目中的粘合剂。  
　　Spring的核心思想是IoC（控制反转），即不再需要程序员去显式地`new`一个对象，而是让Spring框架帮你来完成这一切。  
　　1.6.2 SpringMVC  
　　SpringMVC在项目中拦截用户请求，它的核心Servlet即DispatcherServlet承担中介或是前台这样的职责，将用户请求通过HandlerMapping去匹配Controller，Controller就是具体对应请求所执行的操作。SpringMVC相当于SSH框架中struts。  
　　1.6.3 mybatis  
　　mybatis是对jdbc的封装，它让数据库底层操作变的透明。mybatis的操作都是围绕一个sqlSessionFactory实例展开的。mybatis通过配置文件关联到各实体类的Mapper文件，Mapper文件中配置了每个类对数据库所需进行的sql语句映射。在每次与数据库交互时，通过sqlSessionFactory拿到一个sqlSession，再执行sql命令。

页面发送请求给控制器，控制器调用业务层处理逻辑，逻辑层向持久层发送请求，持久层与数据库交互，后将结果返回给业务层，业务层将处理逻辑发送给控制器，控制器再调用视图展现数据。



**1.7 ssm框架各层关系：**

DAO层、Service层这两个层次都可以单独开发，互相的耦合度很低，完全可以独立进行，这样的一种模式在开发大项目的过程中尤其有优势。

Controller，View层因为耦合度比较高，因而要结合在一起开发，但是也可以看作一个整体独立于前两个层进行开发。这样，在层与层之前只需要知道接口的定义，调用接口即可完成所需要的逻辑单元应用，一切显得非常清晰简单。

**1.8 JavaScript 运行模式：**

JavaScript是一种属于网络的高级脚本语言,已经被广泛用于Web应用开发,常用来为网页添加各式各样的动态功能,为用户提供更流畅美观的浏览效果。通常JavaScript脚本是通过嵌入在HTML中来实现自身的功能的。

1.8.1是一种解释性脚本语言（代码不进行[预编译](https://baike.baidu.com/item/%E9%A2%84%E7%BC%96%E8%AF%91" \t "https://baike.baidu.com/item/_blank)）。

1.8.2主要用来向[HTML](https://baike.baidu.com/item/HTML" \t "https://baike.baidu.com/item/_blank)（[标准通用标记语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E9%80%9A%E7%94%A8%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80" \t "https://baike.baidu.com/item/_blank)下的一个应用）页面添加交互行为。

1.8.3可以直接嵌入HTML页面，但写成单独的[js](https://baike.baidu.com/item/js/10687961" \t "https://baike.baidu.com/item/_blank)文件有利于结构和行为的[分离](https://baike.baidu.com/item/%E5%88%86%E7%A6%BB" \t "https://baike.baidu.com/item/_blank)。

1.8.4跨平台特性，在绝大多数浏览器的支持下，可以在多种平台下运行（如[Windows](https://baike.baidu.com/item/Windows" \t "https://baike.baidu.com/item/_blank)、[Linux](https://baike.baidu.com/item/Linux" \t "https://baike.baidu.com/item/_blank)、[Mac](https://baike.baidu.com/item/Mac/173" \t "https://baike.baidu.com/item/_blank)、[Android](https://baike.baidu.com/item/Android/60243" \t "https://baike.baidu.com/item/_blank)、[iOS](https://baike.baidu.com/item/iOS/45705" \t "https://baike.baidu.com/item/_blank)等）。

1.8.5 JavaScript脚本语言同其他语言一样，有它自身的基本数据类型，表达式和[算术运算符](https://baike.baidu.com/item/%E7%AE%97%E6%9C%AF%E8%BF%90%E7%AE%97%E7%AC%A6/9324947" \t "https://baike.baidu.com/item/_blank)及程序的基本程序框架。JavaScript提供了四种基本的数据类型和两种特殊数据类型用来处理数据和文字。而变量提供存放信息的地方，表达式则可以完成较复杂的信息处理。

**1.9 Vue.js 介绍：**

Vue (读音 /vjuː/，类似于 view) 是一套用于构建[用户界面](https://baike.baidu.com/item/%E7%94%A8%E6%88%B7%E7%95%8C%E9%9D%A2/6582461" \t "https://baike.baidu.com/item/_blank)的渐进式[JavaScript](https://baike.baidu.com/item/JavaScript/321142" \t "https://baike.baidu.com/item/_blank)框架。与其它大型框架不同的是，Vue被设计为可以自底向上逐层应用。Vue 的核心库只关注视图层，不仅易于上手，还便于与第三方库或既有项目整合。另一方面，当与现代化的工具链以及各种支持[类库](https://baike.baidu.com/item/%E7%B1%BB%E5%BA%93/3351433" \t "https://baike.baidu.com/item/_blank)结合使用时，Vue 也完全能够为复杂的单页应用（[SPA](https://baike.baidu.com/item/SPA/17536313" \t "https://baike.baidu.com/item/_blank)）提供驱动。

**1.10 Vue.js 主要功能：**

Vue.js是一套构建用户界面的渐进式框架。与其他重量级框架不同的是，Vue采用自底向上增量开发的设计。Vue 的核心库只关注视图层，并且非常容易学习，非常容易与其它库或已有项目整合。另一方面，Vue 完全有能力驱动采用单文件组件和Vue生态系统支持的库开发的复杂单页应用。

Vue.js 的目标是通过尽可能简单的 API 实现响应的[数据绑定](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E7%BB%91%E5%AE%9A/2884939" \t "https://baike.baidu.com/item/_blank)和组合的视图组件。

Vue.js 自身不是一个全能框架——它只聚焦于视图层。因此它非常容易学习，非常容易与其它库或已有项目整合。另一方面，在与相关工具和支持库一起使用时，Vue.js 也能驱动复杂的单页应用。

**1.11 Vue.js 语言特点：**

易用：在有[HTML](https://baike.baidu.com/item/HTML/97049" \t "https://baike.baidu.com/item/_blank)，[CSS](https://baike.baidu.com/item/CSS/5457" \t "https://baike.baidu.com/item/_blank)，[JavaScript](https://baike.baidu.com/item/JavaScript/321142" \t "https://baike.baidu.com/item/_blank)的基础上，快速上手；Vue.js 的 API 是参考了[AngularJS](https://baike.baidu.com/item/AngularJS" \t "https://baike.baidu.com/item/_blank)、KnockoutJS、Ractive.js、Rivets.js；Vue.js 的 API 的对于其他框架的参考不仅是参考，其中也包含了许多 Vue.js 的独特功能。

灵活：简单小巧的核心，渐进式技术栈，足以应付任何规模的应用。

性能：20kb min+gzip 运行大小、超快虚拟 [DOM](https://baike.baidu.com/item/DOM/50288" \t "https://baike.baidu.com/item/_blank) 、最省心的优化。

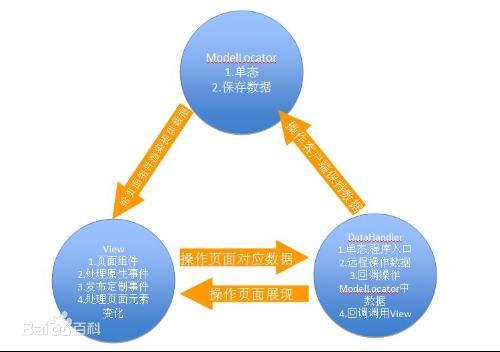
**1.12 Ajax 介绍：**

Ajax即Asynchronous Javascript And XML（异步JavaScript和[XML](https://baike.baidu.com/item/XML/86251" \t "https://baike.baidu.com/item/ajax/_blank)）在 2005年被Jesse James Garrett提出的新术语，用来描述一种使用现有技术集合的‘新’方法，包括:[HTML](https://baike.baidu.com/item/HTML/97049" \t "https://baike.baidu.com/item/ajax/_blank)或[XHTML](https://baike.baidu.com/item/XHTML/316621" \t "https://baike.baidu.com/item/ajax/_blank),CSS,[JavaScript](https://baike.baidu.com/item/JavaScript/321142" \t "https://baike.baidu.com/item/ajax/_blank),[DOM](https://baike.baidu.com/item/DOM/50288" \t "https://baike.baidu.com/item/ajax/_blank),XML,[XSLT](https://baike.baidu.com/item/XSLT/1330564" \t "https://baike.baidu.com/item/ajax/_blank), 以及最重要的[XMLHttpRequest](https://baike.baidu.com/item/XMLHttpRequest/6788735" \t "https://baike.baidu.com/item/ajax/_blank)。使用Ajax技术网页应用能够快速地将增量更新呈现在[用户界面](https://baike.baidu.com/item/%E7%94%A8%E6%88%B7%E7%95%8C%E9%9D%A2/6582461" \t "https://baike.baidu.com/item/ajax/_blank)上，而不需要重载（刷新）整个页面，这使得程序能够更快地回应用户的操作。

**1.13 Ajax 应用：**

该技术在 1998 年前后得到了应用。允许客户端脚本发送[HTTP](https://baike.baidu.com/item/HTTP" \t "https://baike.baidu.com/item/ajax/_blank)请求（XMLHTTP）的第一个组件由Outlook Web Access小组写成。该组件原属于微软 Exchange Server，并且迅速地成为了 Internet Explorer 4.0 的一部分。部分观察家认为，Outlook Web Access 是第一个应用了 Ajax 技术的成功的商业应用程序，并成为包括Oddpost 的网络邮件产品在内的许多产品的领头羊。但是，2005 年初，许多事件使得 Ajax 被大众所接受。Google 在它著名的交互应用程序中使用了异步通讯，如[Google](https://baike.baidu.com/item/Google" \t "https://baike.baidu.com/item/ajax/_blank)、Google 地图、Google 搜索建议、[Gmail](https://baike.baidu.com/item/Gmail" \t "https://baike.baidu.com/item/ajax/_blank)等。Ajax 这个词由《Ajax: A New Approach to Web Applications》一文所创，该文的迅速传播加强了人们使用该项技术的意识。另外，对Mozilla/Gecko 的支持使得该技术走向成熟，变得更为易用。

Ajax 前景非常乐观，可以提高系统性能，优化用户界面。Ajax 现有直接框架 AjaxPro，可以引入 AjaxPro.2.dll 文件，可以直接在前台页面 JavaScript 调用后台页面的方法。但此框架与[表单验证](https://baike.baidu.com/item/%E8%A1%A8%E5%8D%95%E9%AA%8C%E8%AF%81" \t "https://baike.baidu.com/item/ajax/_blank)有冲突。另外[微软](https://baike.baidu.com/item/%E5%BE%AE%E8%BD%AF" \t "https://baike.baidu.com/item/ajax/_blank)也引入了 Ajax 组件，需要添加AjaxControlToolkit.dll 文件，可以在控件列表中出现相关控件。



**1.14 element UI介绍：**

Element，一套为开发者、设计师和产品经理准备的基于 Vue 2.0 的组件库，提供了配套设计资源，帮助你的网站快速成型。由饿了么公司前端团队开源。

**1.15 element UI 特点：**

1.15.1一致性 Consistency

与现实生活一致：与现实生活的流程、逻辑保持一致，遵循用户习惯的语言和概念；

在界面中一致：所有的元素和结构需保持一致，比如：设计样式、图标和文本、元素的位置等。

1.15.2反馈 Feedback

控制反馈：通过界面样式和交互动效让用户可以清晰的感知自己的操作；

页面反馈：操作后，通过页面元素的变化清晰地展现当前状态。

1.15.3效率 Efficiency

简化流程：设计简洁直观的操作流程；

清晰明确：语言表达清晰且表意明确，让用户快速理解进而作出决策；

帮助用户识别：界面简单直白，让用户快速识别而非回忆，减少用户记忆负担。

1.15.4可控 Controllability

用户决策：根据场景可给予用户操作建议或安全提示，但不能代替用户进行决策；

结果可控：用户可以自由的进行操作，包括撤销、回退和终止当前操作等。

**1.16 MyBatis 介绍**

MyBatis 是一款优秀的持久层框架，它支持定制化 SQL、存储过程以及高级映射。MyBatis 避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集。MyBatis 可以使用简单的 XML 或注解来配置和映射原生信息，将接口和 Java 的 POJOs(Plain Ordinary Java Object,普通的 Java对象)映射成数据库中的记录。

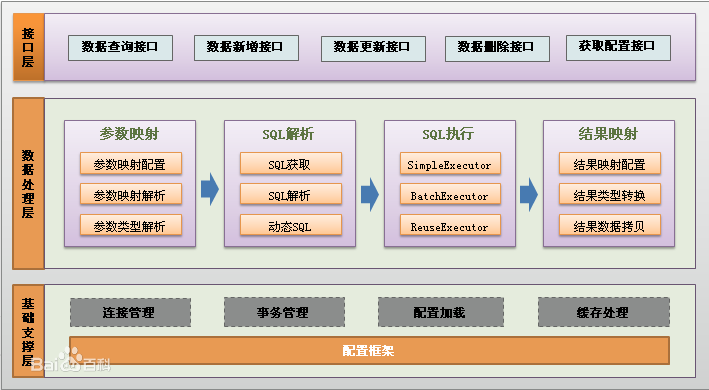
**1.17 MyBatis 功能架构**

我们把Mybatis的功能架构分为三层：

1.17.1 API接口层：提供给外部使用的接口API，开发人员通过这些本地API来操纵数据库。接口层一接收到调用请求就会调用数据处理层来完成具体的数据处理。

1.17.2 数据处理层：负责具体的SQL查找、SQL解析、SQL执行和执行结果映射处理等。它主要的目的是根据调用的请求完成一次数据库操作。

1.17.3 基础支撑层：负责最基础的功能支撑，包括连接管理、事务管理、配置加载和缓存处理，这些都是共用的东西，将他们抽取出来作为最基础的组件。为上层的数据处理层提供最基础的支撑。



**1.18 MyBatis 框架架构**

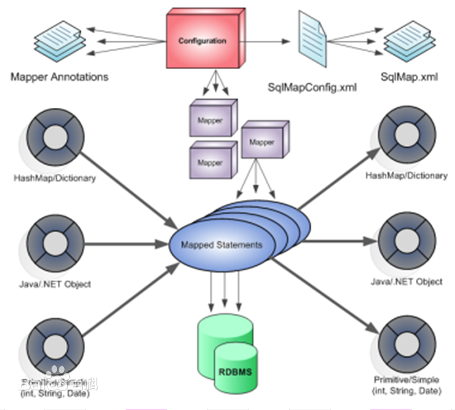
框架架构讲解：

1.18.1加载配置：配置来源于两个地方，一处是配置文件，一处是Java代码的注解，将SQL的配置信息加载成为一个个MappedStatement对象（包括了传入参数映射配置、执行的SQL语句、结果映射配置），存储在内存中。

1.18.2 SQL解析：当API接口层接收到调用请求时，会接收到传入SQL的ID和传入对象（可以是Map、JavaBean或者基本数据类型），Mybatis会根据SQL的ID找到对应的MappedStatement，然后根据传入参数对象对MappedStatement进行解析，解析后可以得到最终要执行的SQL语句和参数。

1.18.3 SQL执行：将最终得到的SQL和参数拿到数据库进行执行，得到操作数据库的结果。

1.18.4结果映射：将操作数据库的结果按照映射的配置进行转换，可以转换成HashMap、JavaBean或者基本数据类型，并将最终结果返回。



**1.19 redis 介绍**

Redis（Remote Dictionary Server )，即远程字典服务，是一个开源的使用ANSI [C语言](https://baike.baidu.com/item/C%E8%AF%AD%E8%A8%80" \t "https://baike.baidu.com/item/Redis/_blank)编写、支持网络、可基于内存亦可持久化的日志型、Key-Value[数据库](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%BA%93/103728" \t "https://baike.baidu.com/item/Redis/_blank)，并提供多种语言的API。从2010年3月15日起，Redis的开发工作由VMware主持。从2013年5月开始，Redis的开发由[Pivotal](https://baike.baidu.com/item/Pivotal" \t "https://baike.baidu.com/item/Redis/_blank)赞助。

**1.20 redis 存储说明**

redis使用了两种[文件格式](https://baike.baidu.com/item/%E6%96%87%E4%BB%B6%E6%A0%BC%E5%BC%8F" \t "https://baike.baidu.com/item/Redis/_blank)：全量数据和增量请求。

全量数据格式是把内存中的数据写入磁盘，便于下次读取文件进行加载；

增量请求文件则是把内存中的数据序列化为操作请求，用于读取文件进行replay得到数据，序列化的操作包括SET、RPUSH、SADD、ZADD。

redis的存储分为内存存储、[磁盘存储](https://baike.baidu.com/item/%E7%A3%81%E7%9B%98%E5%AD%98%E5%82%A8" \t "https://baike.baidu.com/item/Redis/_blank)和log文件三部分，配置文件中有三个参数对其进行配置。

save seconds updates，save配置，指出在多长时间内，有多少次更新操作，就将[数据同步](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%90%8C%E6%AD%A5" \t "https://baike.baidu.com/item/Redis/_blank)到数据文件。这个可以多个条件配合，比如默认配置文件中的设置，就设置了三个条件。

appendonly yes/no ，appendonly配置，指出是否在每次更新操作后进行日志记录，如果不开启，可能会在断电时导致一段时间内的数据丢失。因为redis本身同步数据文件是按上面的save条件来同步的，所以有的数据会在一段时间内只存在于内存中。

appendfsync no/always/everysec，appendfsync配置，no表示等[操作系统](https://baike.baidu.com/item/%E6%93%8D%E4%BD%9C%E7%B3%BB%E7%BB%9F" \t "https://baike.baidu.com/item/Redis/_blank)进行[数据缓存](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E7%BC%93%E5%AD%98" \t "https://baike.baidu.com/item/Redis/_blank)同步到磁盘，always表示每次更新操作后手动调用fsync()将数据写到磁盘，everysec表示每秒同步一次。

**1.21 css3 介绍**

层叠样式表(英文全称：Cascading Style Sheets)是一种用来表现[HTML](https://baike.baidu.com/item/HTML" \t "https://baike.baidu.com/item/CSS/_blank)（[标准通用标记语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E9%80%9A%E7%94%A8%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80/6805073" \t "https://baike.baidu.com/item/CSS/_blank)的一个应用）或[XML](https://baike.baidu.com/item/XML" \t "https://baike.baidu.com/item/CSS/_blank)（标准通用标记语言的一个子集）等文件样式的计算机语言。CSS不仅可以静态地修饰网页，还可以配合各种脚本语言动态地对网页各元素进行格式化。

CSS 能够对网页中元素位置的排版进行像素级精确控制，支持几乎所有的字体字号样式，拥有对网页对象和模型样式编辑的能力。

**1.22 css3 工作原理**

CSS是一种定义样式结构如字体、颜色、位置等的语言，被用于描述网页上的信息格式化和显示的方式。CSS样式可以直接存储于HTML网页或者单独的样式单文件。无论哪一种方式，样式单包含将样式应用到指定类型的元素的规则。外部使用时，样式单规则被放置在一个带有文件扩展名\_css的外部样式单文档中。

样式规则是可应用于网页中元素，如文本段落或链接的格式化指令。样式规则由一个或多个样式属性及其值组成。内部样式单直接放在网页中，外部样式单保存在独立的文档中，网页通过一个特殊标签链接外部样式单。

名称CSS中的“层叠（cascading）”表示样式单规则应用于HTML文档元素的方式。具体地说，CSS样式单中的样式形成一个层次结构，更具体的样式覆盖通用样式。样式规则的优先级由CSS根据这个层次结构决定，从而实现级联效果。

**1.23 Maven 介绍**

Maven项目对象模型(POM)，可以通过一小段描述信息来管理项目的构建，报告和[文档](https://baike.baidu.com/item/%E6%96%87%E6%A1%A3/1009768" \t "https://baike.baidu.com/item/Maven/_blank)的[项目管理工具](https://baike.baidu.com/item/%E9%A1%B9%E7%9B%AE%E7%AE%A1%E7%90%86%E5%B7%A5%E5%85%B7/6854630" \t "https://baike.baidu.com/item/Maven/_blank)软件。

Maven 除了以程序构建能力为特色之外，还提供高级项目管理工具。由于 Maven 的缺省构建规则有较高的可重用性，所以常常用两三行 Maven 构建脚本就可以构建简单的项目。由于 Maven 的面向项目的方法，许多 Apache Jakarta 项目发文时使用 Maven，而且公司项目采用 Maven 的比例在持续增长。

Maven这个单词来自于意第绪语（犹太语），意为知识的积累，最初在Jakata Turbine项目中用来简化构建过程。当时有一些项目（有各自Ant build文件），仅有细微的差别，而JAR文件都由[CVS](https://baike.baidu.com/item/CVS" \t "https://baike.baidu.com/item/Maven/_blank)来维护。于是希望有一种标准化的方式构建项目，一个清晰的方式定义项目的组成，一个容易的方式发布项目的信息，以及一种简单的方式在多个项目中共享[JARs](https://baike.baidu.com/item/JARs" \t "https://baike.baidu.com/item/Maven/_blank)。

**1.24 Maven 特点**

那么，Maven 和 Ant 有什么不同呢？在回答这个问题以前，首先要强调一点：Maven 和 Ant 针对构建问题的两个不同方面。Ant 为 Java 技术开发项目提供跨平台构建任务。Maven 本身描述项目的高级方面，它从 Ant 借用了绝大多数构建任务。因此，由于 Maven 和 [Ant](https://baike.baidu.com/item/Ant/785685" \t "https://baike.baidu.com/item/Maven/_blank)代表两个差异很大的工具，所以接下来只说明这两个工具的等同组件之间的区别，如下表所示。

|  |  |  |
| --- | --- | --- |
|  | Maven | Ant |
| 标准构建文件 | project.xml 和 maven.xml | build.xml |
| 特性处理顺序 | ${maven.home}/bin/driver.properties  ${project.home}/project.properties  ${project.home}/build.properties  ${user.home}/build.properties  通过 -D 命令行选项定义的系统特性  最后一个定义起决定作用。 | 通过 -D 命令行选项定义的系统特性  由 任务装入的特性  第一个定义最先被处理。 |
| 构建规则 | 构建规则更为动态（类似于编程语言）；它们是基于 Jelly 的可执行 XML。 | 构建规则或多或少是静态的，除非使用<script>任务 |
| 扩展语言 | 插件是用 Jelly（XML）编写的。 | 插件是用 Java 语言编写的。 |
| 构建规则可扩展性 | 通过定义 <preGoal> 和 <postGoal> 使构建 goal 可扩展。 | 构建规则不易扩展；可通过使用 <script> 任务模拟 <preGoal> 和 <postGoal> 所起的作用。 |

Maven是一个[项目管理工具](https://baike.baidu.com/item/%E9%A1%B9%E7%9B%AE%E7%AE%A1%E7%90%86%E5%B7%A5%E5%85%B7" \t "https://baike.baidu.com/item/Maven/_blank)，它包含了一个项目对象模型 (Project Object Model)，一组标准集合，一个[项目生命周期](https://baike.baidu.com/item/%E9%A1%B9%E7%9B%AE%E7%94%9F%E5%91%BD%E5%91%A8%E6%9C%9F" \t "https://baike.baidu.com/item/Maven/_blank)(Project Lifecycle)，一个依赖管理系统(Dependency Management System)，和用来运行定义在生命周期阶段(phase)中[插件](https://baike.baidu.com/item/%E6%8F%92%E4%BB%B6" \t "https://baike.baidu.com/item/Maven/_blank)(plugin)目标(goal)的逻辑。当你使用Maven的时候，你用一个明确定义的项目对象模型来描述你的项目，然后Maven可以应用横切的逻辑，这些逻辑来自一组共享的（或者自定义的）插件。

Maven 有一个生命周期，当你运行 mvn install 的时候被调用。这条命令告诉 Maven 执行一系列的有序的步骤，直到到达你指定的生命周期。遍历生命周期旅途中的一个影响就是，Maven 运行了许多默认的[插件](https://baike.baidu.com/item/%E6%8F%92%E4%BB%B6" \t "https://baike.baidu.com/item/Maven/_blank)目标，这些目标完成了像编译和创建一个 JAR 文件这样的工作。

此外，Maven能够很方便的帮你管理项目报告，生成站点，管理JAR文件，等等。

**1.25 Tomcat 特点**

Tomcat是Apache 软件基金会（Apache Software Foundation）的Jakarta 项目中的一个核心项目，由[Apache](https://baike.baidu.com/item/Apache/6265" \t "https://baike.baidu.com/item/tomcat/_blank)、Sun 和其他一些公司及个人共同开发而成。由于有了Sun 的参与和支持，最新的Servlet 和JSP 规范总是能在Tomcat 中得到体现，Tomcat 5支持最新的Servlet 2.4 和JSP 2.0 规范。因为Tomcat 技术先进、性能稳定，而且免费，因而深受Java 爱好者的喜爱并得到了部分软件开发商的认可，成为比较流行的Web 应用服务器。

Tomcat 服务器是一个免费的开放源代码的Web 应用服务器，属于轻量级应用[服务器](https://baike.baidu.com/item/%E6%9C%8D%E5%8A%A1%E5%99%A8" \t "https://baike.baidu.com/item/tomcat/_blank)，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试JSP 程序的首选。对于一个初学者来说，可以这样认为，当在一台机器上配置好Apache 服务器，可利用它响应[HTML](https://baike.baidu.com/item/HTML" \t "https://baike.baidu.com/item/tomcat/_blank)（[标准通用标记语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E9%80%9A%E7%94%A8%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80/6805073" \t "https://baike.baidu.com/item/tomcat/_blank)下的一个应用）页面的访问请求。实际上Tomcat是Apache 服务器的扩展，但运行时它是独立运行的，所以当你运行tomcat 时，它实际上作为一个与Apache 独立的进程单独运行的。

诀窍是，当配置正确时，Apache 为HTML页面服务，而Tomcat 实际上运行JSP 页面和Servlet。另外，Tomcat和[IIS](https://baike.baidu.com/item/IIS" \t "https://baike.baidu.com/item/tomcat/_blank)等Web服务器一样，具有处理HTML页面的功能，另外它还是一个Servlet和JSP容器，独立的Servlet容器是Tomcat的默认模式。不过，Tomcat处理静态[HTML](https://baike.baidu.com/item/HTML" \t "https://baike.baidu.com/item/tomcat/_blank)的能力不如Apache服务器。

**1.26 Tomcat 安全启动**

Tomcat是一个世界上广泛使用的支持jsp和servlets的Web服务器。它在java上运行时能够很好地运行并支持Web应用部署。会因为设置不当，造成灾难性的后果。在Tomcat默认安装，Tomcat作为一个系统服务运行，如果没有将其作为系统服务运行，几乎所有Web服务器管理员都是缺省地将其以Administrator权限运行。这两种方式都允许Java运行时访问Windows系统下任意文件夹中的任何文件。缺省情况下，Java运行时授予安全权限。当 Tomcat以系统管理员身份或作为系统服务运行时，Java运行取得了[系统用户](https://baike.baidu.com/item/%E7%B3%BB%E7%BB%9F%E7%94%A8%E6%88%B7" \t "https://baike.baidu.com/item/tomcat/_blank)或系统管理员所具有的全部权限。这样一来，Java运行时就取得了所有文件夹中所有文件的全部权限。并且Servlets(JSP在运行过程中要转换成Servlets)取得了同样的权限。所以Java代码可以调用Java SDK中的文件API、列出文件夹中的全部文件、删除任何文件，最大的危险在于以系统权限运行一个程序。

1. **软件开发环境及开发工具：**

操作系统：Windows 10、Windows 7、Windows 8

开发语言：Java

使用框架：ssm

前端技术：JavaScript、VUE.js（2.X）、css3

开发工具：IDEA（2020版）/MyEclipse（10）/Eclipse、Visual Studio Code

数据库：MySQL 5.7.26（版本号）

数据库管理工具：phpstudy/Navicat

JDK版本：Java sdk8

Maven:apache-maven 3.8.3-bin

Tomcat:apache-tomcat-7.0.88

1. **硬件环境：**

CPU：1.60GHz及以上

内存：4GB及以上

硬盘：500MB以上

显卡：图像级显卡以上