摘 要

本论文尝试研究SpringMVC+Spring+Mybatis框架下的网站后台程序开发，在web应用的开发过程中,基于SpringMVC+Spring+Mybatis（俗称SSM）的框架组合凭借良好的性能和较快的开发效率，在现在成为主流的Web应用开发框架组合。SpringMVC是一个基于MVC的表现层框架，比如页面上的请求的响应等都是由SpringMVC框架在起作用。Spring框架主要负责的是业务层的功能，主要是起到了管理容器的作用，通过Spring整合SpringMVC和Mybatis，实现表现层与持久层之间的解耦合，降低了代码之间关系的复杂度，使业务逻辑更加清晰。Mybatis框架主要负责的是持久层，它支持定制的SQL语句，存储过程和高级映射，避免了JDBC代码和手动设置参数，使用过程中通过简单的XML文件的配置或注解来配置和映射原生信息，将接口和 Java 的 POJO映射成数据库中的记录，完成和数据库的相关操作。

关键词：SpringMVC Spring Mybatis Web框架

Abstract

This thesis attempts to research the development of website daemon under the framework of SpringMVC+Spring+MyBatis. In the process of web application development, the framework combination based on SpringMVC+Spring+MyBatis (commonly known as SSM) is based on good performance and fast development efficiency. It is now a mainstream web application development framework. SpringMVC is an MVC-based presentation layer framework, such as request response on a page, etc. The SpringMVC framework is at work. The Spring framework is mainly responsible for the business layer functions, mainly to play the role of the management container, through Spring integration SpringMVC and Mybatis, to achieve decoupling between the presentation layer and the persistence layer, reducing the complexity of the relationship between the code, so that Business logic is clearer. The Mybatis framework is mainly responsible for the persistence layer, which supports custom SQL statements, stored procedures and advanced mapping, avoids JDBC code and manually set parameters, and configures and maps native information through the configuration or annotation of simple XML files during use. Interfaces and Java POJOs are mapped into records in the database to perform related operations with the database.

**Key words** ：SpringMVC Spring Mybatis Web FrameWork

引 言

目前，国内外有很多流行的Web开发技术，现在应用最广泛的两类Web开发技术分别是微软（Microsoft）公司的．Net平台和Sun及IBM等公司提出的J2EE平台。J2EE主要面向使用Java程序设计语言进行基于Web的企业应用的开发部署。就服务端市场来讲，Java目前在技术先进性上有绝对优势。应用系统需要各个层面的框架(framework)，Java都有大量已完成或正在进行的开源项目帮助开发人员更好地完成自己的任务。诸如struts/JSF类的框架，.net没有对应物，而且Java在扬弃的同时保持着相对的稳定以及易于操作。因此很多企业都选择J2EE平台的结构设计和解决方案。传统的J2EE以EJB（EnterpriseJavaBean）容器为核心，EJB组件提供事务管理，处理用户请求，支持打包处理和资源管理等功能，但是EJB与EJB容器的高度耦合关系，使其可移植性差，同时EJB代码的复杂、维护的困难使其在现在阶段逐步被淘汰，因此轻量级的框架就应运而生。例如，Struts、Spring、mybatis等框架，降低了程序开发的复杂度，提高了开发时的调试效率，进而提高了Web网站的开发速度，节约了各个方面的成本。

目录

**第1章 云计算的概念和原理1**

1.1云计算的概念1

1.2云计算的原理2

1.3云计算的主要服务形式和典型应用2

1.3.1软件即服务(SaaS)3

1.3.2平台即服务(PaaS)3

1.3.3基础设施服务(IaaS)4

**第2章 经典运算供应商5**

2.1Google的云计算5

2.1.1两步认证机制5

2.2IBM6

2.2.1IBM“蓝云”计算平台6

2.2.2蓝云计算平台中的存储体系结构7

2.2.3IBM打出“组合拳”8

2.2.4IBM安全机制的分析9

2.3Amazon的弹性计算云10

2.3.1Amazon的云机制10

2.3.2Amazon的安全分析11

**第3章 云计算在校园网中的应用与研究12**

3.1资源共享12

3.2搭建“云”交互学习平台13

3.3建立基于云计算的安全校园网14

**第4章 云计算搭建16**

4.1Ubuntu系统的安装16

4.2Hadoop系统部署17

4.3分布式云计算实验19

**总 结21**

**参考文献22**

**致 谢23**

# 第1章 框架的概念和原理

## 1.1.1 SpringMVC的概念以及和Struts2对比

Spring MVC属于SpringFrameWork的后续产品，已经融合在Spring Web Flow里面,它属于Spring中的一部分，Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块。使用 Spring 可插入的 MVC 架构，从而在使用Spring进行WEB开发时，可以选择使用Spring的SpringMVC框架或集成其他MVC开发框架，如Struts2等，由于Struts2框架是类级别的拦截，每次请求就会创建一个Action，和Spring整合时Struts2的ActionBean注入作用域是原型模式prototype（否则会出现线程并发问题），然后通过setter，getter把request数据注入到属性，而SpringMVC是方法级别的拦截，一个方法对应一个Request上下文，所以方法直接基本上是独立的，独享request，response数据。而每个方法同时又何一个url对应，参数的传递是直接注入到方法中的，是方法所独有的。处理结果通过ModeMap返回给框架，SpringMVC实现了零配置，由于SpringMVC基于方法的拦截，有加载一次单例模式bean注入。而Struts2是类级别的拦截，每次请求对应实例一个新的Action，需要加载所有的属性值注入，所以，SpringMVC开发效率和性能高于Struts2。Spring MVC 分离了控制器、模型对象、过滤器以及处理程序对象的角色，这种分离让它们更容易进行定制。

SpringMVC架构图：



图1- 1 SpringMVC架构图

## 1.1.2 SpringMVC的工作流程

1. 用户发送请求至前端控制器DispatcherServlet
2. DispatcherServlet收到请求调用HandlerMapping处理器映射器。
3. 处理器映射器根据请求url找到具体的处理器，生成处理器对象及处理器拦截器(如果有则生成)一并返回给DispatcherServlet。
4. DispatcherServlet通过HandlerAdapter处理器适配器调用处理器
5. 执行处理器(Controller，也叫后端控制器)。
6. Controller执行完成返回ModelAndView
7. HandlerAdapter将controller执行结果ModelAndView返回给DispatcherServlet
8. DispatcherServlet将ModelAndView传给ViewReslover视图解析器
9. ViewReslover解析后返回具体View
10. DispatcherServlet对View进行渲染视图（即将模型数据填充至视图中）。
11. DispatcherServlet响应用户

## SpringMVC组件说明

1.DispatcherServlet：前端控制器

用户请求到达前端控制器，它就相当于mvc模式中的c，dispatcherServlet是整个流程控制的中心，由它调用其它组件处理用户的请求，dispatcherServlet的存在降低了组件之间的耦合性。

2.HandlerMapping：处理器映射器

HandlerMapping负责根据用户请求找到Handler即处理器，springmvc提供了不同的映射器实现不同的映射方式，例如：配置文件方式，实现接口方式，注解方式等。

3.Handler：处理器

Handler 是继DispatcherServlet前端控制器的后端控制器，在DispatcherServlet的控制下Handler对具体的用户请求进行处理。

4.HandlAdapter：处理器适配器

通过HandlerAdapter对处理器进行执行，这是适配器模式的应用，通过扩展适配器可以对更多类型的处理器进行执行。

5.View Resolver：视图解析器

View Resolver负责将处理结果生成View视图，View Resolver首先根据逻辑视图名解析成物理视图名即具体的页面地址，再生成View视图对象，最后对View进行渲染将处理结果通过页面展示给用户。 springmvc框架提供了很多的View视图类型，包括：jstlView、freemarkerView、pdfView等。

## 1.2.1 Spring的概念

Spring是一个开源的Java/Java EE全功能栈的应用程序框架，以Apache License 2.0开源许可协议的形式发布，也有.NET平台上的移植版本。该框架基于 Expert One-on-One Java EE Design and Development（一书中的代码），最初由Rod Johnson和Juergen Hoeller等开发它提供了一个简易的开发方式，这种开发方式，将避免那些可能致使底层代码变得繁杂混乱的大量的属性文件和帮助类，并且面向接口的编程思想贯穿整个系统应用。

## 1.2.2 Spring中包含的关键特性：

1.强大的基于JavaBeans的采用控制反转（Inversion of Control，IoC）原则的配置管理，使得应用程序的组建更加快捷简易。

2.一个可用于Java EE等运行环境的核心Bean工厂。

3.数据库事务的一般化抽象层，允许声明式（Declarative）事务管理器，简化事务的划分使之与底层无关。

4.内建的针对JTA和单个JDBC数据源的一般化策略，使Spring的事务支持不要求Java EE环境，这与一般的JTA或者EJB CMT相反。

5.JDBC 抽象层提供了有针对性的异常等级（不再从SQL异常中提取原始代码），简化了错误处理，大大减少了程序员的编码量。再次利用JDBC时，你无需再写出另一个'终止'（finally）模块。并且面向JDBC的异常与Spring通用数据访问对象（Data Access Object）异常等级相一致。

6.以资源容器，DAO实现和事务策略等形式与Hibernate，JDO和MyBatis、SQL Maps集成。利用众多的翻转控制方便特性来全面支持，解决了许多典型的Hibernate集成问题。所有这些全部遵从 Spring 通用事务处理和通用数据访问对象异常等级规范。

7.灵活的基于核心 Spring 功能的MVC网页应用程序框架。开发者通过策略接口将拥有对该框架的高度控制，因而该框架将适应于多种呈现（View）技术，例如JSP、FreeMarker、Velocity、Thymeleaf 等。值得注意的是，Spring 中间层可以轻易地结合于任何基于MVC框架的网页层，例如Struts、WebWork或Tapestry。

8.提供诸如事务管理等服务的AOP框架。

## 1.2.3 Spring的IOC操作说明

1.把对象的创建交给spring进行管理

2.ioc操作两部分：

（1）ioc的配置文件方式

（2）ioc的注解方式

### 1.2.3.1 ioc底层原理

1.ioc底层原理使用技术

（1）xml配置文件

（2）dom4j解决xml

（3）工厂设计模式

（4）反射技术

2.图示ioc底层原理

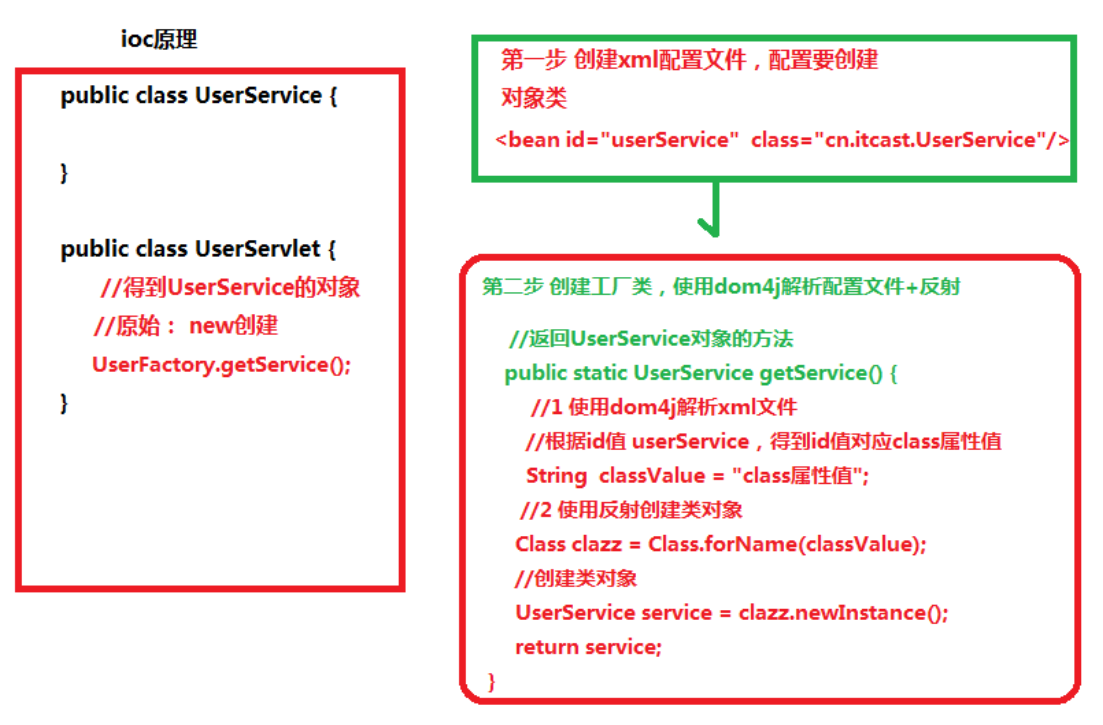


图1-2-3 ioc底层原理图

## 1.2.4 Spring的Bean管理（xml方式）

### 1.2.4.1 Bean的三种实例化方式

1.在spring里面通过配置文件创建对象

2.Bean实例化三种方式实现

第一种 使用类的无参数构造创建（重点）

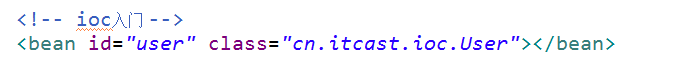
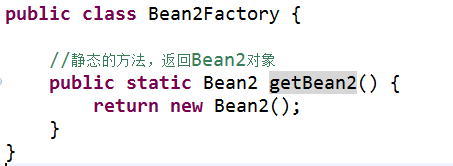
****

图1-2-4-1 无参构造创建bean方式图

第二种 使用静态工厂创建

1. 创建静态的方法，返回类对象

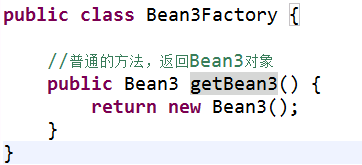
****

****

图1-2-4-2 静态工厂创建bean方式图

第三种 使用实例工厂创建

（1）创建不是静态的方法，返回类对象

****

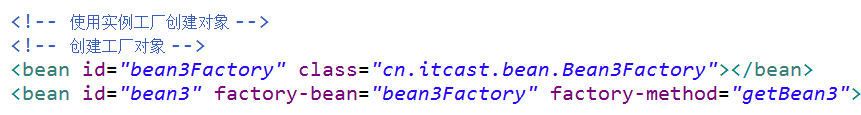
****

图1-2-4-3 实例工厂创建bean方式图

## 1.2.5 Bean属性注入介绍

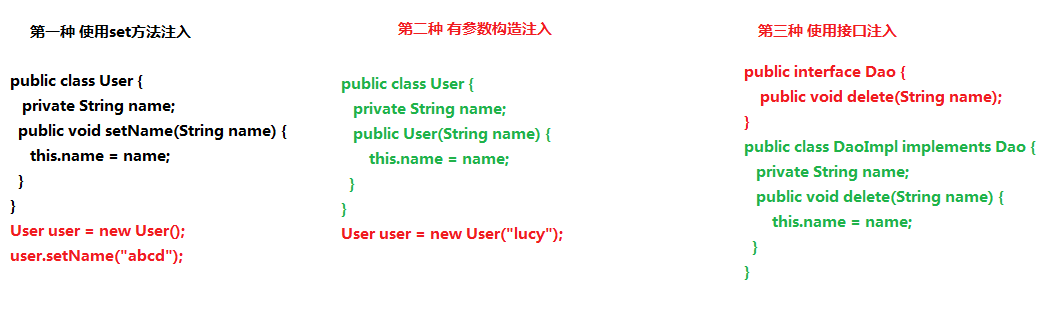
1.创建对象时候，向类里面属性里面设置值

2.属性注入的方式介绍（三种方式）

（1）使用set方法注入

（2）使用有参数构造注入

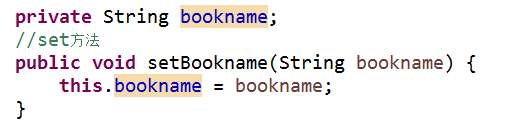
（3）使用接口注入

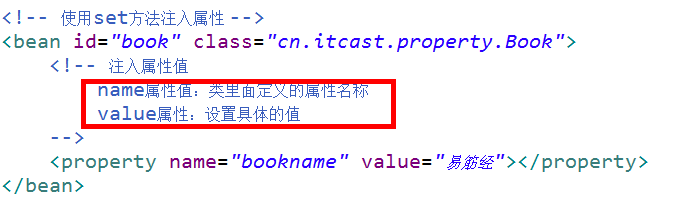
****

1-2-5-1图 属性注入三种方式图

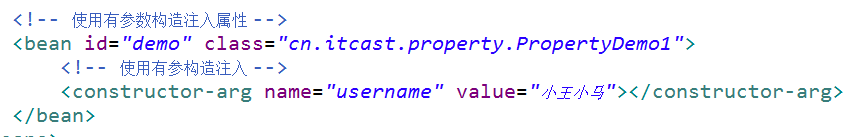
3.在spring框架里面，支持前两种方式

（1）set方法注入（重点）

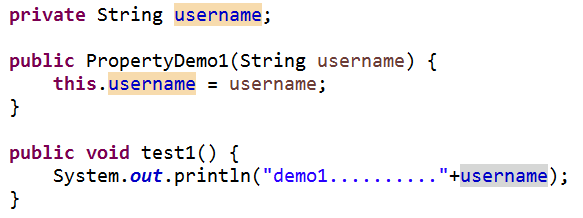
****

****

（2）有参数构造注入

****

1-2-5-2图 set方法注入

****

1-2-5-3图 有参数构造注入图

## 1.2.6 IOC和DI的区别

（1）IOC: 控制反转，把对象创建交给spring进行配置

（2）DI: 依赖注入，向类里面的属性中设置值

（3）关系：依赖注入不能单独存在，需要在ioc基础之上完成操作

## 1.2.7 AOP的概念及原理

### 1.2.7.1 AOP的概念

1 AOP：面向切面（方面）编程，扩展功能不修改源代码实现

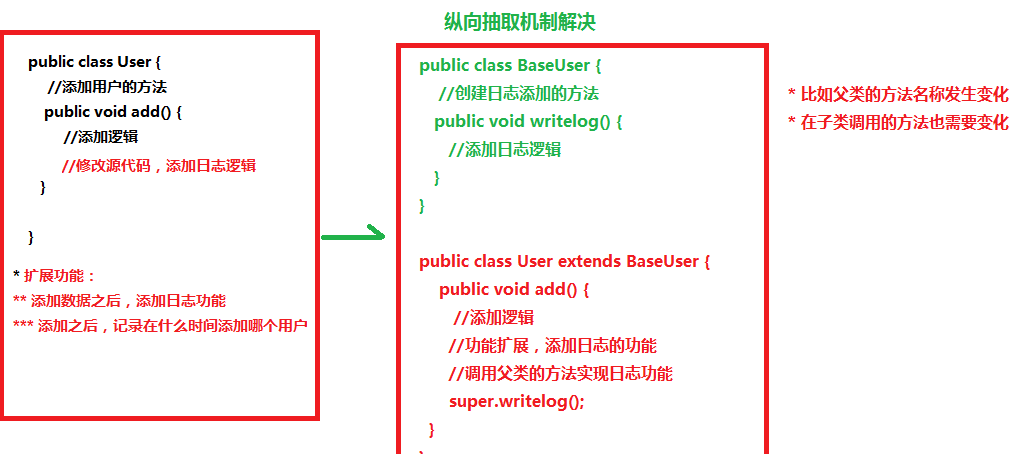
2 AOP采取横向抽取机制，取代了传统纵向继承体系重复性代码

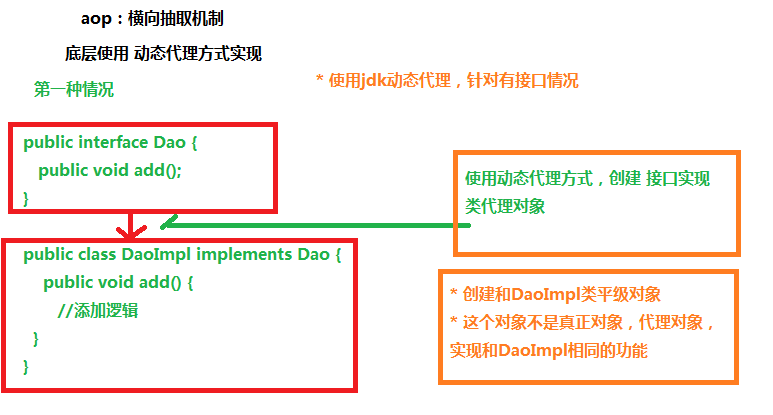
3 AOP底层使用动态代理实现

（1）第一种情况，有接口情况，使用动态代理创建接口实现类代理对象

（2）第二种情况，没有接口情况，使用动态代理创建类的子类代理对象

### 1.2.7.2 图示AOP的原理

****

****

1-2-7-1图 AOP原理图

### 1.2.7.3 AOPchangyong 操作术语

Joinpoint(连接点): 类里面可以被增强的方法，这些方法称为连接点

Pointcut(切入点):所谓切入点是指我们要对哪些Joinpoint进行拦截的定义.

Advice(通知/增强):所谓通知是指拦截到Joinpoint之后所要做的事情就是通知.通知分为前置通知,后置通知,异常通知,最终通知,环绕通知(切面要完成的功能)

Aspect(切面): 是切入点和通知（引介）的结合

Introduction(引介):引介是一种特殊的通知在不修改类代码的前提下, Introduction可以在运行期为类动态地添加一些方法或Field.

Target(目标对象):代理的目标对象(要增强的类)

Weaving(织入):是把增强应用到目标的过程，把advice 应用到 target的过程

Proxy（代理）:一个类被AOP织入增强后，就产生一个结果代理类

### 1.2.7.4 Spring的AOP操作

1.在spring里面进行aop操作，使用aspectj实现

（1）aspectj不是spring一部分，和spring一起使用进行aop操作

（2）Spring2.0以后新增了对AspectJ支持

2.使用aspectj实现aop有两种方式

（1）基于aspectj的xml配置

（2）基于aspectj的注解方式

## 1.3.1 Mybatis的概念

MyBatis 本是[apache](http://baike.baidu.com/view/28283.htm)的一个开源项目[iBatis](http://baike.baidu.com/view/628102.htm), 2010年这个项目由apache software foundation 迁移到了google code，并且改名为MyBatis，实质上Mybatis对ibatis进行一些改进。

MyBatis是一个优秀的持久层框架，它对jdbc的操作数据库的过程进行封装，使开发者只需要关注 SQL 本身，而不需要花费精力去处理例如注册驱动、创建connection、创建statement、手动设置参数、结果集检索等jdbc繁杂的过程代码。

Mybatis通过xml或注解的方式将要执行的各种statement（statement、preparedStatemnt、CallableStatement）配置起来，并通过java对象和statement中的sql进行映射生成最终执行的sql语句，最后由mybatis框架执行sql并将结果映射成java对象并返回。

### 1.3.1.1 Mybatis和传统JDBC比较

1. 优化获取和释放

我们一般在访问数据库时都是通过数据库连接池来操作数据库，数据库连接池有好几种，比如C3P0、DBCP，也可能采用容器本身的JNDI数据库连接池。我们可以通过DataSource进行隔离解耦，我们统一从DataSource里面获取数据库连接，DataSource具体由DBCP实现还是由容器的JNDI实现都可以，所以我们将DataSource的具体实现通过让用户配置来应对变化。

2.SQL统一管理，对数据库进行存取操作

我们使用JDBC对数据库进行操作时，SQL查询语句分布在各个Java类中，这样可读性差，不利于维护，当我们修改Java类中的SQL语句时要重新进行编译。Mybatis可以把SQL语句放在配置文件中统一进行管理，以后修改配置文件，也不需要重新就行编译部署。

3.生成动态SQL语句

我们在查询中可能需要根据一些属性进行组合查询，比如我们进行商品查询，我们可以根据商品名称进行查询，也可以根据发货地进行查询，或者两者组合查询。如果使用JDBC进行查询，这样就需要写多条SQL语句。Mybatis可以在配置文件中通过使用<if test=””></if>标签进行SQL语句的拼接，生成动态SQL语句。

4.能够对结果集进行映射

我们在使用JDBC进行查询时，返回一个结果集ResultSet,我们要从结果集中取出结果封装为需要的类型，在Mybatis中我们可以设置将结果直接映射为自己需要的类型，比如：JavaBean对象、一个Map、一个List等等。像上个例子中就是将结果映射为int类型。

### 1.3.1.2 Mybatis和Hibernate的对比

Mybatis和hibernate不同，它不完全是一个ORM框架，因为MyBatis需要程序员自己编写Sql语句，不过mybatis可以通过XML或注解方式灵活配置要运行的sql语句，并将java对象和sql语句映射生成最终执行的sql，最后将sql执行的结果再映射生成java对象。

Mybatis学习门槛低，简单易学，程序员直接编写原生态sql，可严格控制sql执行性能，灵活度高，非常适合对关系数据模型要求不高的软件开发，例如互联网软件、企业运营类软件等，因为这类软件需求变化频繁，一但需求变化要求成果输出迅速。但是灵活的前提是mybatis无法做到数据库无关性，如果需要实现支持多种数据库的软件则需要自定义多套sql映射文件，工作量大。

Hibernate对象/关系映射能力强，数据库无关性好，对于关系模型要求高的软件（例如需求固定的定制化软件）如果用hibernate开发可以节省很多代码，提高效率。但是Hibernate的学习门槛高，要精通门槛更高，而且怎么设计O/R映射，在性能和对象模型之间如何权衡，以及怎样用好Hibernate需要具有很强的经验和能力才行。

总之，按照用户的需求在有限的资源环境下只要能做出维护性、扩展性良好的软件架构都是好架构，所以框架只有适合才是最好。

## 1.3.2 图解Mybatis架构



1-3-2-1图 Mybatis架构图

## Mybatis操作步骤

1.mybatis配置SqlMapConfig.xml，此文件作为mybatis的全局配置文件，配置了mybatis的运行环境等信息。mapper.xml文件即sql映射文件，文件中配置了操作数据库的sql语句。此文件需要在SqlMapConfig.xml中加载。

2.通过mybatis环境等配置信息构造SqlSessionFactory即会话工厂

3.由会话工厂创建sqlSession即会话，操作数据库需要通过sqlSession进行。

4.mybatis底层自定义了Executor执行器接口操作数据库，Executor接口有两个实现，一个是基本执行器、一个是缓存执行器。

5.Mapped Statement也是mybatis一个底层封装对象，它包装了mybatis配置信息及sql映射信息等。mapper.xml文件中一个sql对应一个Mapped Statement对象，sql的id即是Mapped statement的id。

6.Mapped Statement对sql执行输入参数进行定义，包括HashMap、基本类型、pojo，Executor通过Mapped Statement在执行sql前将输入的java对象映射至sql中，输入参数映射就是jdbc编程中对preparedStatement设置参数。

7.Mapped Statement对sql执行输出结果进行定义，包括HashMap、基本类型、pojo，Executor通过Mapped Statement在执行sql后将输出结果映射至java对象中，输出结果映射过程相当于jdbc编程中对结果的解析处理过程。

## 1.3.4 MyBatis中的#{}和${}

#{}表示一个占位符号，通过#{}可以实现preparedStatement向占位符中设置值，自动进行java类型和jdbc类型转换，#{}可以有效防止sql注入。 #{}可以接收简单类型值或pojo属性值。 如果parameterType传输单个简单类型值，#{}括号中可以是value或其它名称。

${}表示拼接sql串，通过${}可以将parameterType 传入的内容拼接在sql中且不进行jdbc类型转换， ${}可以接收简单类型值或pojo属性值，如果parameterType传输单个简单类型值，${}括号中只能是value。

## 1.3.5 parameterType和resultType

parameterType：指定输入参数类型，mybatis通过ognl从输入对象中获取参数值拼接在sql中。

resultType：指定输出结果类型，mybatis将sql查询结果的一行记录数据映射为resultType指定类型的对象。

## 1.3.6 selectOne和selectList

1.selectOne查询一条记录，如果使用selectOne查询多条记录则抛出异常：

org.apache.ibatis.exceptions.TooManyResultsException: Expected one result (or null) to be returned by selectOne(), but found: 3 at org.apache.ibatis.session.defaults.DefaultSqlSession.selectOne(DefaultSqlSession.java:70)

2.selectList可以查询一条或多条记录。

## 1.3.7 Mybatis的Mapper动态代理方式

### 1.3.7.1原始Dao开发中存在以下问题：

(1)Dao方法体存在重复代码：通过SqlSessionFactory创建SqlSession，调用SqlSession的数据库操作方法

(2)调用sqlSession的数据库操作方法需要指定statement的id，这里存在硬编码，不得于开发维护。

### 1.3.7.2 Mapper实现原理

Mapper接口开发方法只需要程序员编写Mapper接口（相当于Dao接口），由Mybatis框架根据接口定义创建接口的动态代理对象，代理对象的方法体同上边Dao接口实现类方法。

### 1.3.7.3 Mapper接口开发需要遵循以下规范：

1.Mapper.xml文件中的namespace与mapper接口的类路径相同。

2.Mapper接口方法名和Mapper.xml中定义的每个statement的id相同

3.Mapper接口方法的输入参数类型和mapper.xml中定义的每个sql 的parameterType的类型相同

4.Mapper接口方法的输出参数类型和mapper.xml中定义的每个sql的resultType的类型相同

## 1.3.8 Mybatis的typeAliases（类型别名）

### 1.3.8.1 Mybatis支持的别名

|  |  |
| --- | --- |
| 别名 | 映射的类型 |
| \_byte | byte |
| \_long | long |
| \_short | short |
| \_int | int |
| \_integer | int |
| \_double | double |
| \_float | float |
| \_boolean | boolean |
| string | String |
| byte | Byte |
| long | Long |
| short | Short |
| int | Integer |
| integer | Integer |
| double | Double |
| float | Float |
| boolean | Boolean |
| date | Date |
| decimal | BigDecimal |
| bigdecimal | BigDecimal |

### 1.3.8.2 自定义别名

在SqlMapConfig.xml中配置：

<typeAliases>

<!-- 单个别名定义 -->

<typeAlias alias=*"user"* type=*"cn.wbz.mybatis.po.User"*/>

<!-- 批量别名定义，扫描整个包下的类，别名为类名（首字母大写或小写都可以） -->

<package name=*"cn.wbz.mybatis.po"*/>

<package name=*"其它包"*/>

</typeAliases>

# 第2章 经典运算供应商

## 2.1.Google的云计算

谷歌作为云计算的发起者，云计算在开发过程中有先天的概念和时间优势，但google的云计算1安全性，据报道google的文件在2009年3月7日大量的用户文件泄露事件。 美国的隐私保护组织在Google上吸引政府采取措施加强云计算产品的安全性。 云计算允许用户在世界的任何角落更新文档，并与他人共享。

如果您是Google文档用户，在不知情的情况下，您的许多文件会突然显示在其他人的帐户中，其他人可以轻松查看。

当时的安全漏洞是，谷歌可能会将用户的信息分享给以前分享过的用户，但现在已不再拥有该权利。这个安全漏洞从目前的技术角度来说可以使一个相对低级的漏洞，但对于一个大的，不透明的网络，这个问题可以说无处不在，要解决这个逻辑，技术漏洞是云计算安全最基本的问题，只有实现对那些可以拥有云计算安全应用程序的人保护。

## 2.1.2 两步认证机制

继信息泄露事件以后，各界对google的安全问题方面的质疑就一直没有停过，google推出了自己的两步认证机制，两步认证机制是很容易建立，管理和使用的。当管理员建立了 该认证机制以后，登录Google Apps账户就需要经过两步鉴定，一个是你知道的：一个密码;还有一个是你所有的：一个移动电话不需要任何特殊的密码或设备。 一个认证码就会发送到你的手机里，这个认证码是通过短信，语音电话发送或通过一个安装在你的Android，黑莓或iphone上的应用程序自动生成的。这使得得只有你自己才能访问你 的数据：就计是有人盗了你的密码，他们也访问不了你的账户。

从表面上看，这是一个好主意，但是，在福布斯发表的博客中表示，两步认证机制可能只会降低黑客攻击的效率，并且不能提供真正的安全防护。 Google还承认，两步验证机制无法阻止实时网上诱骗和木马攻击。 谷歌还说：我们不希望用户认为我们可以阻止所有的网络攻击。 他们提醒用户，除了Google的电话认证欺诈外，他们还应该使用安全的浏览器 - 他提到所有浏览器 - 并运行防病毒软件。

我个人想听听谷歌如何解决那些不喜欢使用手机，短信或智能手机应用程序用户问题的人。 或者应用程序的安全性，特别是如果手机被盗。

## 2.2 IBM

IBM的安全机制现在是几个主要的云计算提供商做得更好。所以须得在这个的角度深度分析。

## 2.2.1 IBM“蓝云”计算平台

“蓝云“解决方案是由IBM云计算中心开发的企业级云计算解决方案，集成了现有的企业基础设施，通过虚拟化技术和自动化技术构建自己的云计算中心，实现企业硬件资源和软件资源，统一管理， 统一分布，统一部署，统一监控和统一备份，打破专用资源的应用，从而帮助企业实现云计算。

IBM的“蓝云”计算平台是一套软件，硬件平台，互联网将使用这种技术扩展到企业平台，使得数据中心类似于使用计算环境的互联网。 “Blue Cloud”广泛使用IBM的先进大规模计算技术，结合IBM自己的软件，硬件系统和服务技术，支持开放标准和开源软件。

“Blue Cloud”基于IBM Almaden研究中心的云基础设施，使用Xen和PowerVM虚拟化软件，Linux操作系统映像和Hadoop软件（Google文件系统和MapReduce开源实现）。IBM正式推出了x86- 基于服务器系统的“蓝云”产品。 下图显示了IBM“蓝云”的架构：

图2-2 蓝云基础框架

如图所示，“蓝云”计算平台由数据中心，IBM Tivoli部署工程师，IBM Tivoli监视，IBM WebSphere Application Server，IBM DB2数据库和一些开源信息处理软件和开源虚拟化软件组成 。 “蓝云”硬件平台环境和一般的x86服务器集群类似，使用的方式提高了刀片的计算密度。 “蓝云”软件平台功能主要体现在虚拟机和大型数据处理软件Apache Hadoop上的使用。

蓝云平台的一个重要特点是使用虚拟化技术。 虚拟化中的“蓝云”在两个层次上，一个是在硬件层面实现虚拟化，另一个是通过开源软件实现虚拟化。 硬件级虚拟化可以使用IBM p系列服务器获取硬件逻辑分区LPAR（逻辑分区）。 CPU资源的逻辑分区可以通过IBM Enterprise Workload Manager进行管理。 这样，实际使用过程中的资源分配策略可以用于将相应的资源分配给逻辑分区。 p系列系统的逻辑分区具有1/10 CPU的最小粒度。 Xen是软件级虚拟化，可以在基于另一个操作系统的Linux上运行。

## 2.2.2 蓝云 计算平台中的存储体系结构

蓝云计算平台中的存储架构：蓝云计算平台中的存储架构对于云计算也很重要，无论是操作系统，服务程序还是用户应用程序的数据存储系统。 Blue Cloud存储架构包含类似于Google File System的集群文件系统和基于块设备方法的存储区域网络SAN。

在为云计算平台设计存储架构时，您可以通过组合多个磁盘来获得大量磁盘容量。 相对于磁盘的容量，在云计算平台存储中，磁盘数据的读写速度是更重要的问题，因此需要对多个磁盘同时进行读写操作。 这种方法要求将数据分配给多个节点的多个磁盘。 要实现这一点，存储技术有两个选择，一个是使用类似于Google File System的集群文件系统，另一个是基于块设备的存储区域网络SAN系统。

在蓝云计算平台上，SAN系统和分布式文件系统（如Google File System）不是相互对立的系统，SAN提供了一个块设备接口，需要在此基础上构建一个文件系统，以便成为应用程序 的上层使用。 Google文件系统恰好是可以构建在SAN之上的分布式文件系统。 两者都可以提供可靠性，可扩展性，至于如何使用也需要在云计算平台上构建确定应用，这也反映了计算平台和应用之间的上下关系。

“蓝云”计算平台中的存储架构对于云计算也很重要，无论是操作系统，服务程序还是用户的应用程序数据都存储在存储系统中。 Blue Cloud存储架构包含类似于Google File System的集群文件系统和基于块设备方法的存储区域网络SAN。

在为云计算平台设计存储架构时，您可以通过组合多个磁盘来获得大量磁盘容量。 相对于磁盘的容量，在云计算平台存储中，磁盘数据的读写速度是更重要的问题，因此需要对多个磁盘同时进行读写操作。 这种方法要求将数据分配给多个节点的多个磁盘。 要实现这一点，存储技术有两个选择，一个是使用类似于Google File System的集群文件系统，另一个是基于块设备的存储区域网络SAN系统。

在蓝云计算平台上，SAN 系统和分布式文件系统(如Google File System)不是相互对立的系统，SAN提供了一个块设备接口，需要在此基础上构建一个文件系统。以便成为应用程序的上层使用。Google文件系统恰好是可以构建SAN之上的分布式文件系统。两者都可以提供可靠性，可扩展性，至于如何使用也需要云计算平台上构建确定应用，这也反映了计算平台和应用之间的上下关系。

## 2.2.3 IBM打出“组合拳”

进入IBM公司的安全机制的引进，几家主要的云计算提供商做得更好。 所以要分析。

IBM于2009年5月宣布推出新的服务，旨在提高硬件和软件的云计算环境的安全性，保证数据的机密性，并防止对应用程序的攻击。

IBM X-Force安全研究团队的最新研究表明，全球犯罪组织正以惊人的速度开发新的攻击技术。 同时，公司越来越多地部署更多的协作式商业模式，并使用新的IT基础设施，包括云计算，虚拟化和Web 2.0，这些都为安全团队带来了新的复杂性。 IBM新的安全产品组合旨在满足其客户的需求，使公司能够利用新技术模型解决新出现的危险，并通过简化的安全解决方案身份验证和数据安全管理来帮助关键的安全和合规项目。

“新的计算模式从根本上要求公司重新思考如何对合规，风险管理，数据和应用程序保护做出响应。 IBM互联网安全系统（ISS）总经理Brian Truskowski说，“业界采取的方法是'首先，考虑安全性，这造成了高成本，复杂的情况，今天，IBM推动嵌入安全的创新 通过平台的基础设施，将安全从昂贵的障碍转变为商业。

IBM Rational AppScan恶意软件扫描（针对IBM Rational AppScan的恶意软件扫描） - 此软件将Rational AppScan强大的扫描和测试软件与ISS X-Force恶意软件研究和检测引擎相结合。 通过简单的操作，用户可以自动主动扫描和测试站点，以查明是否存在嵌入式恶意软件，并对内容进行分析，以确定恶意内容和位置的存在。 如果发现恶意软件，这些问题将被报告并从应用程序和网站中删除，并且用户将受到保护。

Proventia Web应用程序防火墙 - 此新模块嵌入在IBM ISS Proventia产品组合中，以帮助客户通过单一解决方案防御更多网络漏洞。 强制安全团队研究结果，旨在在网络内提供保护，防止网络应用，主要漏洞和攻击。

IBM Tivoli身份和访问保证 - 通过集中式身份，访问，身份验证和审阅服务，帮助监控，管理和降低身份和访问的风险。 它还提供关闭的特权用户监视，控制和修复。 此服务将通过IBM ISS托管身份服务提供，以帮助进一步降低保护成本。

IBM Tivoli Data和Application Security - 通过加密数据来帮助您解决与丢失的备份磁带和磁盘相关的隐私和合规性风险。 此服务将增强访问控制，并能够跟踪用户对策略的合规性。

IBM Tivoli Security Management for z / OS（IBM Tivoli Security Management for z / OS）有助于改进资源访问控制设施（RACF）的操作管理和合规性状态。 这项新服务加强了大型机的集中管理，监控事件和事件，并减少了安全漏洞。

基于IBM提供的云计算系统和和多种安全措施，近年来IBM公司的云计算业并没有出现像谷歌这样的大规信息披露事件在业内也受到广泛赞誉，这是IBM公司云计算业务近年来的快速发展的一个重要因素。

## 2.2.4 IBM安全机制的分析

但个人认为，IBM的安全措施比较繁琐，虽然IBM拥有丰富的产品线和充足的人力资源，但为了突然有一些不同的云计算解决方案做得不错，这并不容易做到。理论机制比较健全，但在真正长期连续安全运行的过程中不可避免地会有一些遗漏，疏忽，安全问题是不可避免的。再次，IBM的“拳击组合”措施缺乏技术和产品的原创性：这主要是与当前的私有云技术领导者思科和VMware相比，虽然在Cisco UCS上的IBM ex5架构做了一定的反击，但是它是难以震撼思科，VMware和EMC这种组合的黄金在私有云上的技术优势和集成优势。一般来说，IBM的云计算解决方案基本上是一些以前的更新解决方案，而不是革命。

## 2.3 Amazon的弹性计算云

## 2.3.1 Amazon的云机制

亚马逊是互联网上最大的在线零售商，必须购买很多服务器才能应对交易的高峰期。 在大多数时候，大多数服务器空闲，导致大量浪费，为了合理使用空闲服务器，亚马逊已经建立了自己的云计算平台弹性计算云EC2（弹性计算云），并且是第一个 基础设施作为公司的销售服务。

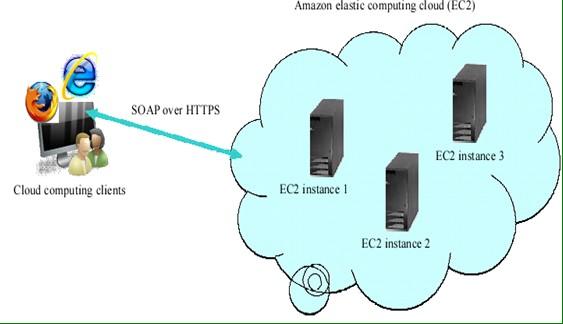
亚马逊自己的灵活计算云构建在公司内大规模集群计算的平台上，用户可以通过灵活的云计算网络接口操纵云计算平台的各种实例。 用户使用实例的支付方式的方式由用户的使用状态决定，即用户只需要为用户使用的计算平台实例付费，在操作完成后结束计费。 这里描述的示例是由用户控制的完整的虚拟机运行实例。 这样，用户不必建立自己的云计算平台，节省了设备和维护成本。

图2-3 EC2系统

该图显示了EC2系统使用模式。从图中可以看出，弹性计算云用户使用客户端通过SOAP 、over HTTPS协议与弹性计算云的实例进行交互。这样，灵活的计算云平台为用户或开发人员提供了一个虚拟集群环境，具有用户的灵活性，同时还降低了云计算平台所有者（Amazon）的管理负担。弹性计算云的每个实例代表正在运行的虚拟机，用户具有对其虚拟机的完全访问权限，包括此虚拟机操作系统的管理员权限。虚拟机收费是基于虚拟机计算成本的能力。

## 2.3.2 Amazon的安全分析

总而言之，亚马逊通过提供灵活的计算云来降低维护负担，满足小规模软件开发人员对集群系统的需求。 收费方法比较简单明了：用户使用多少资源，只需支付这部分资源即可。

为了灵活计算云的进一步发展，亚马逊计划如何帮助用户在云计算平台的基础上开发网络应用程序。 除了网络零售业务，云计算也是亚马逊公司的核心价值。 亚马逊将来会在基于云的灵活计算平台上添加更多的网络服务组件模块，为用户搭建云计算应用程序提供便利。

对于安全机制，亚马逊没有任何创新的设计。 换句话说，亚马逊近年来的快速增长势头成为云计算提供商的领导者，而且在2011年最新的最佳云计算提供商排名第一，因此他们取得这样的成绩是他们成熟的业务运营的主要原因 意味着亚马逊的云计算服务机制更适合企业对云计算的要求。

毕竟，亚马逊是网络书业务的开始，相对于IBM，微软，谷歌等技术中心云计算提供商的技术实力，在整体运营方面有一定的差距，有不同的方向。

# 第3章 云计算在校园网中的应用与研究

目前，高校缺乏统一的网络教学信息系统，不同的系统，学校与教师，教师和学生之间的信息传递，资源信息共享不能有效保证，在教学信息整理，分类，查询， 统计和学生课堂教学，家庭作业调查，评价和评价，系统功能不完善或尚未完全建立。 出现云计算，高校提供系统运行模式的信息，高校通过访问云服务提供商建立教学信息系统云软件库，可以获得所需的系统应用程序。

无需单独投资建立一整套内部软件程序，成本相对较低。使用云计算平台，将分布在各地建立各种虚拟资源，事项资源共享的应用水平，是高校能适应教学情况的变化，重点加强教学、推进教学管理，提高课堂教学效率。

## 3.1资源共享

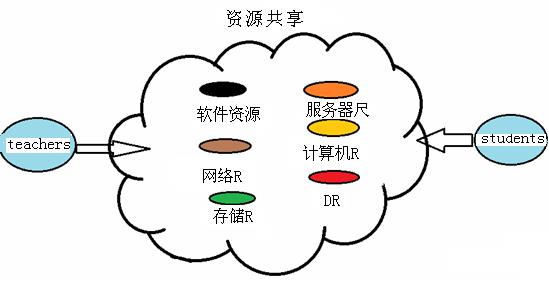
根据云资源共享的范围，云共享分为公共云，私有云和混合云。

（1）公共云：云资源提供给外部团体和组织使用，一般公众通过网络可以动态，灵活，自助地访问云中的公共资源。 对于一般大众来说，使用公有云资源，不需要购买任何硬件和软件，不需要考虑数据安全问题，使其专注于自己的个人业务，具有很高的性价比。

（2）私有云：企业自己构建，云资源只为自己内部协作和共享。 它仅限于单个组织或组，外部组织无法访问这些资源。 私有云的主要目的不是销售他们的能力，而是允许本地用户在其管理范围内使用其私有和灵活的基础架构，以便他们可以控制和运行部署在基础架构上的应用程序。 另外，企业还可以将私有云托管到云提供商进行构建，使企业对云资源进行更高的控制，并将基础设施的安装，配置和操作等任务移交给云计算提供商的实施 。 提供商，如Sun和IBM，托管私有云服务提供商。

（3）混合云：私有和公共云服务的组合，多个组或组织通过可靠的网络在公有云和私有云之间共享资源。 使用虚拟化技术公司可以在数据中心内部构建自己的私有云，公司也可以选择使用公有云，组合两者，形成混合云。 混合云是在促进企业需求方面形成和产生的。 通过使用混合云，企业可以在应用和成本之间实现平衡，这有助于减少由于迁移到云而产生的一些问题。 对长远发展更有利。 因此，混合云被认为是大多数企业使用云计算的趋势。

电子技术和信息技术的快速发展，今天，软件和硬件不断更新，对于各大高校，无论是日常办公还是教学和研究方面的硬件和软件设施投资都是巨大的费用，这些设备经常 一个巨大的投资在使用它需要多长时间才能更新或升级。 教学和研究，往往需要安装和使用大量软件，正版软件版税，后期软件更新和教学软件，涉及硬件安装并提出新的要求

由于云计算对终端计算机本身的使用不是太高的要求，而且设备的更换任务都被抛到了服务提供商 - 建立了数据中心或相关服务提供商。 云计算技术的本质是实现资源的全部共享，核心思想是利用大量的网络资源来链接统一的管理和调度，构成资源库并为用户提供按需服务。

云资源可以无限扩展，分析，建立以下资源共享图：

图3-1 云计算资源共享图

通过建立云资源共享，利用混合云共享，使园区网络硬件和软件资源共享。

## 3.2 搭建“云”交互学习平台

云计算允许学生快速轻松地构建个人学习环境，以支持和促进个人学习环境和非正式学习的发展。 以学生为中心的个人学习环境将是在线学习的发展趋势，非正式学习将占学生学习活动的比例越来越大。 互联网学习不再局限于学校机构的范围，而是更广泛的，学生可以创建和管理自己的空间，学习活动更多的是由学生自己而不是学校机构来控制。 学生控制和管理个人学习，选择所需的学习资源和服务。

不同的学生将选择不同的学习内容和工具来创建他们的个人学习环境。 云计算技术允许学生在任何时间，任何地点从任何终端访问信息技术和其他服务，并提高电子学习的灵活性和灵活性。 云计算解决方案不仅允许学生参与研究，还允许他们更好地管理自己的项目和工作负载。

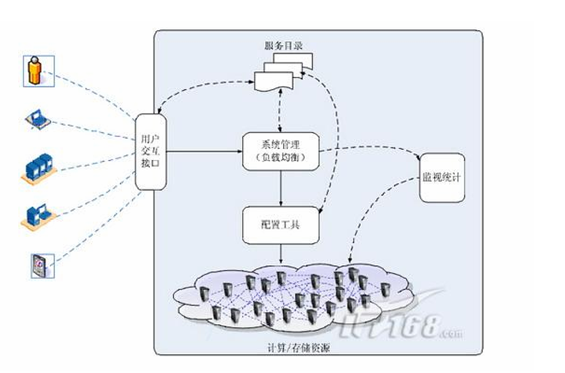
在构建云交互式学习平台之前，要进行详细规划，绘制业务数据流，了解手动部署过程，组织必要的结构，是实现标准化，以及当前自动化过程集成实现更高层次，全球自动化管理 。 最后构建云交换平台地图，通过云计算园区网，学生可以通过园区网互通，等等提高学生的电脑学习知识。

图3-2 云交互学习平台图

## 3.3建立基于云计算的安全校园网

电子技术和信息技术的快速发展，今天，软件和硬件不断更新，对于各大高校，无论是日常办公还是教学和研究方面的硬件和软件设施投资都是一笔巨大的开支，这些设备经常一个巨大的投资在使用它需要多长时间才能更新或升级。教学和研究，往往需要安装和使用大量软件，正版软件版税，后期软件更新和教学软件，涉及硬件安装并提出新的要求。由于云计算对终端计算机本身的使用不是太高的要求，而且设备的更换任务都被抛到了服务提供商 - 建立了数据中心或相关服务提供商。教育组织数据中心，网络中心相关任务将能够使用云计算服务完成，通过云计算提供的IT基础设施，可以节省成本，不再需要投资昂贵的硬件设备，频繁维护和升级的负担。大大降低硬件和软件资源的管理和维护成本，节省人力和资源。

校园网络由于计算机和存储设备的交叉使用，导致广泛的计算机病毒在校园网络中传播。 安装防病毒软件，除了使用费用，必须定期备份重要的数据和文件，防止病毒是通过网络打滑破坏。 云计算为园区网络提供最可靠，最安全的数据存储中心。 我们不必担心数据丢失，病毒入侵等问题。 云计算严格的权利管理策略可以帮助您与指定的人共享数据。 同时，数据的集中存储更容易实现安全监控：通过将信息存储在一个或多个数据中心中， 对于校园内部的学生，老师及工作人员，通过云安全来访问校园网络软硬件资源，而外部网络不能访问。

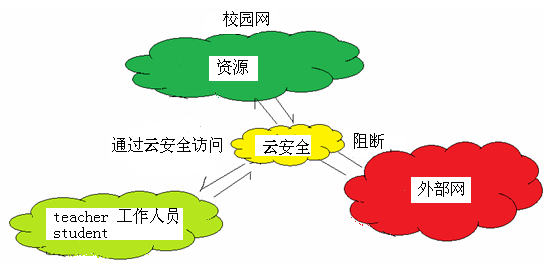


图3-3 安全校园网图

# 第四章 云计算搭建

## 4.1 Ubuntu系统的安装

Ubuntu（乌班图）是一个以桌面应用为主的Linux操作系统，其名称来自非洲南部祖鲁语或豪萨语的“Ubuntu”一词，意思是“人性”、“我的存在是因为大家的存在”，是非洲传统的一种价值观，类似华人社会的“仁爱”思想。Ubuntu基于Debian发行版和GNOME桌面环境，而从11.04版起，Ubuntu发行版放弃了Gnome桌面环境，改为Unity，与Debian的不同在于它每6个月会发布一个新版本。Ubuntu的目标在于为一般用户提供一个最新的、同时又相当稳定的主要由自由软件构建而成的操作系统。

LTS 是 Ubuntu 的长期支持版，因此 Ubuntu 14.04 支持周期长达 3-5 年。因此 Ubuntu 14.04 是追求稳定的用户和企业的最佳选择。所以本次课程设计选择ubuntu14.04LTS版本完全能够应付云平台搭建与相关实验的任务。

图4-1 Ubuntu14.04LTS系统

## 4.2 Hadoop系统部署

修改机器名：

打开/etc/hostname文件，将/etc/hostname文件中的Ubuntu改为你想取的机器名。这里我取“s15“。重启系统后才会生效。

安装ssh服务：

在terminal窗口中输入：Sudoa apt-get install open ssh-server

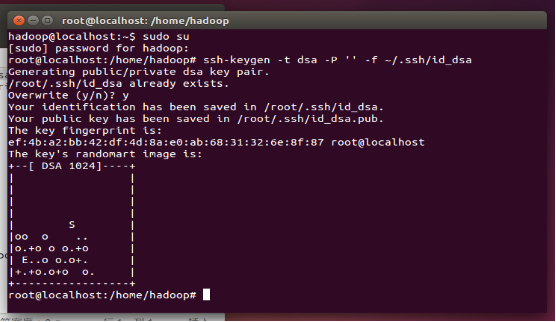
建立ssh无密码登录本机

在terminal窗口中输入：

ssh-keygen -t dsa -P '' -f ~/.ssh/id\_dsa

 cat ~/.ssh/id\_dsa.pub >> ~/.ssh/authorized\_keys

得到如下图说明操作正确：

图4-2 建立ssh无密码登录本机

登录local host：

在terminal窗口中输入：bin/start-all.sh

安装Hadoop :

下载Hadoop 安装包并解压，打开Hadoop/conf/Hadoop.sh文件，配置conf/Hadoop.sh：找到#export JAVA\_HOME=...一行，去掉#，然后加上本机JDK的路径。

打开conf/core-site.XML文件，加入如下代码：

<configuration>

<property>

  <name>fs.default.name</name>

  <value>hdfs://localhost:9000</value>

 </property>

</configuration>

打开conf/map red-site.XML文件，编辑如下：

<configuration>

     <property>

      <name>mapred.job.tracker</name>

      <value>localhost:9001</value>

     </property>

    </configuration>

打开conf/masters文件和conf/slaves文件，添加secondary的主机名，作为单机版环境，这里只需填写local host就Ok了。

到这里Hadoop系统就部署完毕了。调用bin/start-all.sh 命令即可以启动Hadoop，用JSP命令查看系统状态，出现如下信息说明系统部署成功：

图4-3 用jsp命令查看状态

## 4.3分布式云计算实验

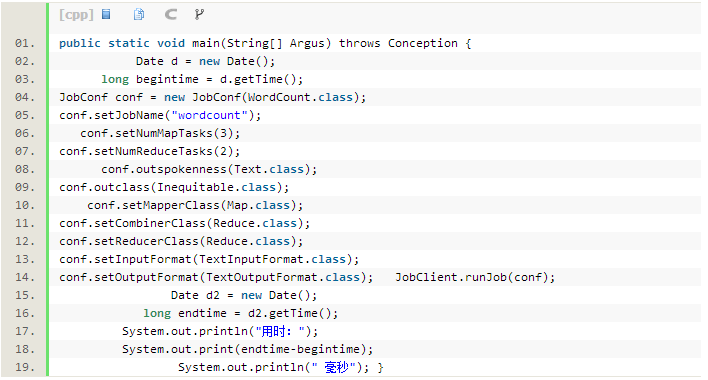
用于分布式计算的程序核心设计如下：

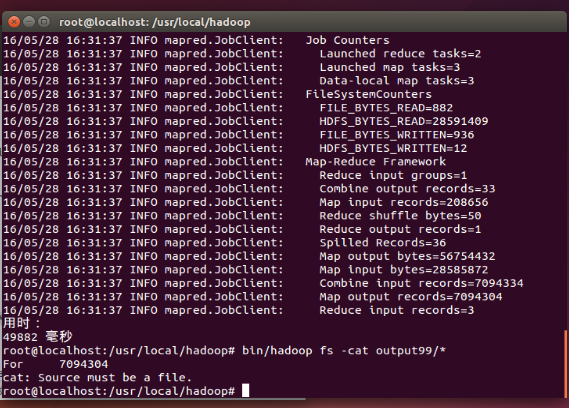
图4-4 程序设计

其中 Date d1，d2用于捕获程序运行开始和结束的时间其时间差即为程序运算所耗费的时间；conf.setNumMapTasks(3)设置分布式运算时所以的map数量为3个，conf.setNumReduceTasks(2);设置运算中所以的reduce数量为2个，其他的设置为配置Map Reduce的其他必要参数，在这里对本实验无影响就不解析了。

想要利用云平台来为我们做计算服务还要把写好的Map Reduce程序进行编译，打包成库文件，然后将模拟大数据的TXT文档提交到HDFS空间，执行以下命令就能启动云计算了：

bin/Hadoop as -put input/ input 把文件提交到HDFS空间

bin/hadoop jar WordCount.jar WordCount input output 启动云计算

图4-5 云计算实验结果

总 结

电子技术和信息技术的快速发展，今天，软件和硬件不断更新，对于各大高校，无论是日常办公还是教学和研究方面的硬件和软件设施投资都是巨大的费用，这些设备经常有一个巨大的投资在使用它需要多长时间才能更新或升级。 教学和研究，往往需要安装和使用大量软件，正版软件版税，后期软件更新和教学软件，涉及硬件安装并提出新的要求。

由于云计算对终端计算机本身的使用不是太高的要求，而且设备的更换任务都被抛到了服务提供商 - 建立了数据中心或相关服务提供商。 教育组织数据中心，网络中心相关任务将能够利用云计算服务完成，通过云计算提供的IT基础设施，可以节省成本，不再需要投资昂贵的硬件设备，频繁维护和升级的负担。 大大降低硬件和软件资源的管理和维护成本，节省人力和资源。

# 参考文献

1. 杨永川.信息安全[M].中国人民公安大学出版社.2007年1月.
2. 游向峰.打造安全的网络环境之”云计算” [J].电脑编程技巧与维护2009，(16):12-23.
3. 薛质.信息安全技术基础和安全策略[M].北京：清华大学出版社.2007.
4. 门汝静.近期网络安全的特点与热点[J].现代电信科技2009，（1）:14-17.
5. 张帅.安全云计算你准备好了吗[M],2008.
6. [胡小菁,范并思.云计算给图书馆管理带来挑战[J].大学图书馆学报, 2009(4):7-12.
7. 周 舒,张岚岚. 云计算改善数字图书馆用户体验初探[J].图书馆学研究, 2009(4):28-30
8. 王长全，艾雰.云计算时代的数字图书馆信息安全思考[J].图书馆建设, 2010(1):50-52
9. 魏志鹏，李慧佳，祖央.云计算影响下的图书馆信息服务研究[J].图书馆,2010(2):87-88,93
10. 杨明芳，袁曦临.云计算环境下的数字图书馆[ J].图书馆建设,2009(9):7-9,12
11. 王文清，陈凌. CALIS 数字图书馆云服务平台模型[J].大学图书馆学报，2009(4):13-18
12. 严真·云计算环境下图书馆的变革[ J].图书馆工作与研究, 2010(2): 37-39
13. 陈全,邓倩妮·云计算及其关键技术[ J].计算机应用, 2009, 29(9): 2562-2564
14. 吴华.“云计算”环境下图书馆信息资源建设探讨[ J].科技信息,2010（25）
15. 644. 张凌超.基于“云计算”的数字图书馆建设模式初探[J].图书馆学研究,2010(11):39-42.
16. 薛毅飞.云计算在数字图书馆中的应用[ J].图书与档案，2009(29):372一373.
17. 孙卫.图书馆在云时代的思考[ J].数字图书馆论坛，2009(6):35一41
18. 胡新平.云图书馆构想[ J].情报理论与探索，2010(6):29一30.
19. 杨明芳，袁曦临.云计算环境下的数字图书馆[ J].图书馆建设.2009(9):7一9.
20. 王长宁.云计算环境下数字图书馆信息资源建设的理性思考[J].中国科技信息，2010(17):163一164.
21. 王红.“云图书馆”平台的架构与实现[ J ].情报理论与实践，2010(10):108一109.
22. 张健.云计算概念和影响力解析[ J ].电信网技术，20O9(1):15一18.
23. 卢军.云计算离企业应用有多远? [ J].信息系统工程，2008(7):31一33.

# 致 谢

感谢电子科技大学成都学院三年来对我的辛苦培育，让我在大学这三年来学到很东西，特别感谢云计算科学与技术系为我提供了良好的学习环境、感谢领导、老师们三年来对我无微不至的关怀和指导，让我得以在这三年中学到很多有用的知识。在此，我还要感谢在班里同学和朋友，感谢你们在我遇到困难的时候帮助我，给我支持和鼓励，感谢你们。

特别感谢我的指导老师姚红，平时给予我悉心指导，在我遇到很多困难情况下她总会给我鼓励与指引，使我能够克服重重困难，将毕业设计做完成，在此谨向姚老师致以诚挚的谢意和崇高的敬意。谢谢!