系统环境：Ubuntu 16.04 TLS x64

环境配置准备阶段：

1.#添加更新源

命令：sudo vim /etc/apt/sources.list

2.添加如下内容：

#阿里源：

deb http://mirrors.aliyun.com/ubuntu/ trusty main restricted universe multiverse

deb http://mirrors.aliyun.com/ubuntu/ trusty-security main restricted universe multiverse

deb http://mirrors.aliyun.com/ubuntu/ trusty-updates main restricted universe multiverse

deb http://mirrors.aliyun.com/ubuntu/ trusty-proposed main restricted universe multiverse

deb http://mirrors.aliyun.com/ubuntu/ trusty-backports main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ trusty main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ trusty-security main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ trusty-updates main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ trusty-proposed main restricted universe multiverse

deb-src http://mirrors.aliyun.com/ubuntu/ trusty-backports main restricted universe multiverse

3.输入:wq退出

4.# 更新源缓存

sudo apt-get update

5.#安装中文支持环境(install Chinese support)

命令：sudo apt-get install -y language-pack-zh-hant language-pack-zh-hans

export LANG "zh\_CN.UTF-8"

6.#安装去依赖管理包工具(install aptitude)

命令：sudo apt-get install -y aptitude

7.#安装基础工具(install basic tools)

命令：sudo apt-get install -y -f wget vim python

安装开发语言环境：

1.Python环境配置：

(1)#安装pip管理包工具(install pip)

命令：sudo wget https://bootstrap.pypa.io/get-pip.py;python get-pip.py;rm -f get-#pip.py

(2)#安装python库(lxml requests)

命令：pip install --upgrade lxml requests redis

(3)#安装爬虫环境

命令：pip install wheel Scrapy pymongo requests celery -i http://#pypi.douban.com/simple --trusted-host pypi.douban.com

2.安装jdk1.8：

(1)下载好jdk包

(2)命令：tar –xvf jdk1.8xxx. tar.gz

(3)命令：mv -r jdk1.8xx/ /opt/jdk1.8

(4)命令：vim /etc/profile

(5)添加如下内容：

export JAVA\_HOME=/opt/jdk1.8

export PATH=$PATH:$JAVA\_HOME/bin

安装集成管理环境：

1.安装apache-maven

(1)下载好maven包

(2)命令：tar –xvf apache-maven-xxxx.tar.gz

(3)命令：mv –r apache-maven-xxxx/ /opt/apache-maven

(4)命令：vim /etc/profile

(5)添加如下内容：

export MAVEN\_HOME=/opt/apache-maven

export PATH=$PATH:$MAVEN\_HOME/bin

export MAVEN\_OPTS="-Xmx2g -XX:MaxPermSize=512M -XX:ReservedCodeCacheSize=512m"

安装集成发布环境：

1.Docker配置安装：

(1)命令：sudo apt-get install docker.io

(2)命令：service docker.io status

(3)命令：service docker.io start

2.Docker-compose安装：

(1)命令：

curl -L https://get.daocloud.io/docker/compose/releases/download/1.13.0/docker-compose-`uname -s`-`uname -m` > /usr/local/bin/docker-compose

(2)命令：chmod +x /usr/local/bin/docker-compose

安装分布式数据库：

1.安装redis数据库

(1)命令：sudo apt-get update

(2)命令：sudo apt-get install redis-server

安装分布式计算分析环境：

1.安装hadoop和yarn

(1)下载hadoop

命令：wget <http://mirrors.hust.edu.cn/apache/hadoop/common/hadoop-2.7.3/hadoop-2.7.3.tar.gz>

(2)命令：tar –xvf hadoop-2.7.3.tar.gz

(3)命令：mv –r hadoop-2.7.3/ /opt/

(4)配置hadoop

命令：vim /etc/profile

添加如下内容：

export HADOOP\_HOME="/opt/hadoop-2.7.3"

export HADOOP\_COMMON\_LIB\_NATIVE\_DIR=$HADOOP\_HOME/lib/native

export HADOOP\_OPTS="-Djava.library.path=$HADOOP\_HOME/lib:$HADOOP\_COMMON\_LIB\_NATIVE\_DIR"

export HADOOP\_CONF\_DIR=$HADOOP\_HOME/etc/hadoop

export YARN\_CONF\_DIR=$HADOOP\_HOME/etc/hadoop

export PATH=$PATH: $HADOOP\_HOME/bin:$HADOOP\_HOME/sbin

(5)命令：vim /opt/hadoop-2.7.3/etc/hadoop/core-site.xml

在configuration节点下添加:

<property>

<name>fs.defaultFS</name>

<value>hdfs://master:9000</value>

</property>

<property>

<name>io.file.buffer.size</name>

<value>131072</value>

</property>

<property>

<name>hadoop.tmp.dir</name>

<value>/opt/hadoop-2.7.3/tmp</value>

</property>

(6)命令：vim /opt/hadoop-2.7.3/etc/hadoop/hdfs-site.xml

在configuration节点下添加:

<property>

<name>dfs.namenode.secondary.http-address</name>

<value>master:50090</value>

</property>

<property>

<name>dfs.replication</name>

<value>2</value>

</property>

<property>

<name>dfs.namenode.name.dir</name>

<value>file:/opt/hadoop-2.7.3/hdfs/name</value>

</property>

<property>

<name>dfs.datanode.data.dir</name>

<value>file:/opt/hadoop-2.7.3/hdfs/data</value>

</property>

(7)命令：cp /opt/hadoop-2.7.3/etc/hadoop/mapred-site.xml.template /opt/hadoop-2.7.3/etc/hadoop/mapred-site.xml

(8)命令：vim /opt/hadoop-2.7.3/etc/hadoop/mapred-site.xml

在configuration节点下添加：

<property>

<name>mapreduce.framework.name</name>

<value>yarn</value>

</property>

<property>

<name>mapreduce.jobhistory.address</name>

<value>master:10020</value>

</property>

<property>

<name>mapreduce.jobhistory.address</name>

<value>master:19888</value>

</property>

(9)命令：vim /opt/hadoop-2.7.3/etc/hadoop/yarn-site.xml

在configuration节点下添加：

<property>

<name>yarn.nodemanager.aux-services</name>

<value>mapreduce\_shuffle</value>

</property>

<property>

<name>yarn.resourcemanager.address</name>

<value>master:8032</value>

</property>

<property>

<name>yarn.resourcemanager.scheduler.address</name>

<value>master:8030</value>

</property>

<property>

<name>yarn.resourcemanager.resource-tracker.address</name>

<value>master:8031</value>

</property>

<property>

<name>yarn.resourcemanager.admin.address</name>

<value>master:8033</value>

</property>

<property>

<name>yarn.resourcemanager.webapp.address</name>

<value>master:8088</value>

</property>

(10)命令：vim /etc/hosts

在源文件的基础上加上:

127.0.0.1 master

(11)配置ssh互通(当前用户为root)

命令：ssh-keygen –t rsa –P ‘’#一路回车直到生成公钥

命令：cat /root/.ssh/id\_rsa.pub >> /root/.ssh/authorized\_keys #master主机

(12)命令：hadoop namenode –format #格式化节点

2.安装spark

(1)下载spark

命令：wget <http://d3kbcqa49mib13.cloudfront.net/spark-2.1.0-bin-hadoop2.7.tgz>

(2)命令：tar –xvf spark-2.1.0-bin-hadoop2.7.tgz

(3)命令：mv –r spark-2.1.0-bin-hadoop2.7/ /opt/

(4)命令：vim /etc/profile

添加如下内容：

export SPARK\_HOME=/opt/spark-2.1.0-bin-hadoop2.7/

export PATH="$SPARK\_HOME/bin:$PATH"

(5)命令:cp /opt/spark-2.1.0-bin-hadoop2.7/conf/spark-env.sh.template /opt/spark-2.1.0-bin-hadoop2.7/conf/spark-env.sh

(6)vim /opt/spark-2.1.0-bin-hadoop2.7/conf/spark-env.sh

在源文件基础上添加：

export JAVA\_HOME=/usr/java/jdk1.8.0\_112/

export SPARK\_MASTER\_IP=master

export SPARK\_WORKER\_MEMORY=1g

export HADOOP\_CONF\_DIR=/opt/hadoop-2.7.3/etc/hadoop

(7)命令：cp /opt/spark-2.1.0-bin-hadoop2.7/conf/slaves.template /opt/spark-2.1.0-bin-hadoop2.7/conf/slaves

(8)命令：vim /opt/spark-2.1.0-bin-hadoop2.7/conf/slaves

添加如下内容：

master

显示层：Spring框架搭建：(Eclipse为例):

1.新建一个Maven项目

选择以org.apache.maven.archetypes.maven-archetype-webapp.1.0为模版建立的Maven Webapp project.

2.配置文件pom.xml

更改如下：

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4\_0\_0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.edu.yc</groupId>

<artifactId>ResultShow</artifactId>

<packaging>war</packaging>

<version>0.0.1-SNAPSHOT</version>

<name>yc Maven Webapp</name>

<url>http://maven.apache.org</url>

<properties>

<!-- main version setting -->

<spring.version>4.1.9.RELEASE</spring.version>

<validator.version>5.2.4.Final</validator.version>

<mybatis.version>3.2.8</mybatis.version>

<mybatis-spring.version>1.2.3</mybatis-spring.version>

<druid.version>1.0.18</druid.version>

<ehcache.version>2.6.11</ehcache.version>

<ehcache-web.version>2.0.4</ehcache-web.version>

<shiro.version>1.2.3</shiro.version>

<sitemesh.version>2.4.2</sitemesh.version>

<activiti.version>5.21.0</activiti.version>

<!-- tools version setting -->

<slf4j.version>1.7.7</slf4j.version>

<commons-lang3.version>3.3.2</commons-lang3.version>

<commons-io.version>2.4</commons-io.version>

<commons-codec.version>1.9</commons-codec.version>

<commons-fileupload.version>1.3.1</commons-fileupload.version>

<commons-beanutils.version>1.9.1</commons-beanutils.version>

<jackson.version>2.2.3</jackson.version>

<fastjson.version>1.1.40</fastjson.version>

<xstream.version>1.4.7</xstream.version>

<guava.version>17.0</guava.version>

<dozer.version>5.5.1</dozer.version>

<poi.version>3.9</poi.version>

<freemarker.version>2.3.20</freemarker.version>

<!-- jdbc driver setting -->

<mysql.driver.version>5.1.30</mysql.driver.version>

<oracle.driver.version>10.2.0.4.0</oracle.driver.version>

<mssql.driver.version>1.3.1</mssql.driver.version>

<!-- environment setting -->

<jdk.version>1.6</jdk.version>

<tomcat.version>2.2</tomcat.version>

<jetty.version>7.6.14.v20131031</jetty.version>

<webserver.port>8181</webserver.port>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<downloadSources>true</downloadSources>

</properties>

<!-- 设定主仓库，按设定顺序进行查找。 -->

<repositories>

<repository>

<id>jeesite-repos</id>

<name>Jeesite Repository</name>

<url>http://maven.aliyun.com/nexus/content/groups/public</url>

</repository>

</repositories>

<!-- 设定插件仓库 -->

<pluginRepositories>

<pluginRepository>

<id>jeesite-repos</id>

<name>Jeesite Repository</name>

<url>http://maven.aliyun.com/nexus/content/groups/public</url>

</pluginRepository>

</pluginRepositories>

<dependencies>

<!-- SPRING begin -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-core</artifactId>

<version>${spring.version}</version>

<exclusions>

<exclusion>

<groupId>commons-logging</groupId>

<artifactId>commons-logging</artifactId>

</exclusion>

</exclusions>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-beans</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context-support</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-aop</artifactId>

<version>${spring.version}</version>

<exclusions>

<exclusion>

<groupId>commons-logging</groupId>

<artifactId>commons-logging</artifactId>

</exclusion>

</exclusions>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-tx</artifactId>

<version>${spring.version}</version>

</dependency>

<!-- spring orm -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-orm</artifactId>

<version>${spring.version}</version>

</dependency>

<!-- WEB begin -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-web</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-oxm</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>opensymphony</groupId>

<artifactId>sitemesh</artifactId>

<version>${sitemesh.version}</version>

</dependency>

<dependency>

<groupId>taglibs</groupId>

<artifactId>standard</artifactId>

<version>1.1.2</version>

<type>jar</type>

</dependency>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>jstl</artifactId>

<version>1.2</version>

<type>jar</type>

</dependency>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>servlet-api</artifactId>

<version>2.5</version>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>javax.servlet.jsp</groupId>

<artifactId>jsp-api</artifactId>

<version>2.1</version>

<scope>provided</scope>

</dependency>

<!-- AOP begin -->

<dependency>

<groupId>org.aspectj</groupId>

<artifactId>aspectjrt</artifactId>

<version>1.7.4</version>

</dependency>

<dependency>

<groupId>org.aspectj</groupId>

<artifactId>aspectjweaver</artifactId>

<version>1.7.4</version>

</dependency>

<dependency>

<groupId>cglib</groupId>

<artifactId>cglib</artifactId>

<version>3.1</version>

</dependency>

<!-- AOP end -->

<!-- LOGGING begin -->

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-api</artifactId>

<version>${slf4j.version}</version>

</dependency>

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-log4j12</artifactId>

<version>${slf4j.version}</version>

</dependency>

<!-- common-logging 实际调用slf4j -->

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>jcl-over-slf4j</artifactId>

<version>${slf4j.version}</version>

</dependency>

<!-- java.util.logging 实际调用slf4j -->

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>jul-to-slf4j</artifactId>

<version>${slf4j.version}</version>

</dependency>

<!-- LOGGING end -->

<!-- redis.clients/jedis -->

<dependency>

<groupId>redis.clients</groupId>

<artifactId>jedis</artifactId>

<version>2.7.3</version>

</dependency>

<!-- org.apache.spark/spark-core\_2.10 -->

<dependency>

<groupId>org.apache.spark</groupId>

<artifactId>spark-core\_2.10</artifactId>

<version>1.6.1</version>

</dependency>

<!-- GENERAL UTILS begin -->

<dependency>

<groupId>org.apache.commons</groupId>

<artifactId>commons-lang3</artifactId>

<version>${commons-lang3.version}</version>

</dependency>

<dependency>

<groupId>commons-io</groupId>

<artifactId>commons-io</artifactId>

<version>${commons-io.version}</version>

</dependency>

<dependency>

<groupId>commons-codec</groupId>

<artifactId>commons-codec</artifactId>

<version>${commons-codec.version}</version>

</dependency>

<dependency>

<groupId>commons-fileupload</groupId>

<artifactId>commons-fileupload</artifactId>

<version>${commons-fileupload.version}</version>

</dependency>

<dependency>

<groupId>commons-beanutils</groupId>

<artifactId>commons-beanutils</artifactId>

<version>${commons-beanutils.version}</version>

<exclusions>

<exclusion>

<groupId>commons-logging</groupId>

<artifactId>commons-logging</artifactId>

</exclusion>

</exclusions>

</dependency>

<!-- xstream xml -->

<dependency>

<groupId>com.thoughtworks.xstream</groupId>

<artifactId>xstream</artifactId>

<version>${xstream.version}</version>

</dependency>

<!-- jackson json -->

<dependency>

<groupId>com.fasterxml.jackson.core</groupId>

<artifactId>jackson-core</artifactId>

<version>${jackson.version}</version>

</dependency>

<dependency>

<groupId>com.fasterxml.jackson.core</groupId>

<artifactId>jackson-databind</artifactId>

<version>${jackson.version}</version>

</dependency>

<dependency>

<groupId>com.fasterxml.jackson.core</groupId>

<artifactId>jackson-annotations</artifactId>

<version>${jackson.version}</version>

</dependency>

<dependency>

<groupId>com.fasterxml.jackson.module</groupId>

<artifactId>jackson-module-jaxb-annotations</artifactId>

<version>${jackson.version}</version>

</dependency>

<dependency>

<groupId>com.alibaba</groupId>

<artifactId>fastjson</artifactId>

<version>${fastjson.version}</version>

</dependency>

<!-- TEST begin -->

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.11</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-test</artifactId>

<version>${spring.version}</version>

</dependency>

<!-- TEST end -->

</dependencies>

<build>

<finalName>yc</finalName>

<plugins>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.5.1</version>

<configuration>

<source>${jdk.version}</source>

<target>${jdk.version}</target>

<showWarnings>true</showWarnings>

</configuration>

</plugin>

<!-- 打包jar文件时，配置manifest文件，加入lib包的jar依赖 -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-jar-plugin</artifactId>

<version>2.4</version>

<configuration>

<encoding>${project.build.sourceEncoding}</encoding>

</configuration>

</plugin>

<!-- war 打包插件, 设定war包名称不带版本号 -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-war-plugin</artifactId>

<version>2.4</version>

<configuration>

<webappDirectory>${project.build.directory}/${project.artifactId}</webappDirectory><!--

<webXml>${project.basedir}/target/jspweb.xml</webXml> -->

<warName>${project.artifactId}</warName>

</configuration>

</plugin>

<!-- jetty插件 -->

<plugin>

<groupId>org.mortbay.jetty</groupId>

<artifactId>jetty-maven-plugin</artifactId>

<version>${jetty.version}</version>

<configuration>

<connectors>

<connector implementation="org.eclipse.jetty.server.nio.SelectChannelConnector">

<port>${webserver.port}</port>

</connector>

</connectors>

<webAppConfig>

<contextPath>/${project.artifactId}</contextPath>

</webAppConfig>

<systemProperties>

<systemProperty>

<name>org.mortbay.util.URI.charset</name>

<value>${project.build.sourceEncoding}</value>

</systemProperty>

</systemProperties>

</configuration>

</plugin>

</plugins>

</build>

</project>

3.在src/main下面新建一个resources文件夹，点击右键build-path->Use as Source Folder

4.在resources文件夹下建一个applicationContext.xml

内容如下：

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:context="http://www.springframework.org/schema/context"

xmlns:jdbc="http://www.springframework.org/schema/jdbc" xmlns:jee="http://www.springframework.org/schema/jee"

xmlns:util="http://www.springframework.org/schema/util"

xmlns:task="http://www.springframework.org/schema/task"

xsi:schemaLocation="

http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-4.1.xsd

http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.1.xsd

http://www.springframework.org/schema/jdbc http://www.springframework.org/schema/jdbc/spring-jdbc-4.1.xsd

http://www.springframework.org/schema/jee http://www.springframework.org/schema/jee/spring-jee-4.1.xsd

http://www.springframework.org/schema/util http://www.springframework.org/schema/util/spring-util-4.1.xsd

http://www.springframework.org/schema/task http://www.springframework.org/schema/task/spring-task-4.1.xsd"

default-lazy-init="true">

<!-- 加载配置属性文件 autowired 去掉-->

<context:property-placeholder ignore-unresolvable="true" location="classpath:yc.properties" />

<context:component-scan base-package="com.edu.yc">

<context:exclude-filter type="annotation" expression="org.springframework.stereotype.Controller"/>

</context:component-scan>

<!-- <bean id="EmployeeDao" ...>

<property name="sqlSessionFactory" ref="sqlSessionFactory"></property>

</bean> -->

</beans>

5. 在resources文件夹下建一个applicationContext-mvc.xml

内容如下:

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:context="http://www.springframework.org/schema/context"

xmlns:mvc="http://www.springframework.org/schema/mvc"

xsi:schemaLocation="http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans-4.1.xsd

http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.1.xsd

http://www.springframework.org/schema/mvc http://www.springframework.org/schema/mvc/spring-mvc-4.1.xsd">

<!-- 加载配置属性文件 autowired 去掉 -->

<context:property-placeholder

ignore-unresolvable="true" location="classpath:yc.properties" />

<context:component-scan base-package="com.edu.yc">

<context:include-filter type="annotation" expression="org.springframework.stereotype.Controller" />

</context:component-scan>

<!-- 默认的注解映射的支持，org.springframework.web.servlet.mvc.method.annotation.RequestMappingHandlerMapping -->

<mvc:annotation-driven content-negotiation-manager="contentNegotiationManager">

<mvc:message-converters register-defaults="true">

<!-- 将StringHttpMessageConverter的默认编码设为UTF-8 -->

<bean class="org.springframework.http.converter.StringHttpMessageConverter">

<constructor-arg value="UTF-8" />

</bean>

<!-- 将Jackson2HttpMessageConverter的默认格式化输出为false -->

<bean class="org.springframework.http.converter.json.MappingJackson2HttpMessageConverter">

<property name="supportedMediaTypes">

<list><value>application/json;charset=UTF-8</value></list>

</property>

<property name="prettyPrint" value="false"/>

</bean>

<!-- 使用XML格式输出数据 -->

<bean class="org.springframework.http.converter.xml.MarshallingHttpMessageConverter">

<constructor-arg>

<bean class="org.springframework.oxm.xstream.XStreamMarshaller">

<property name="streamDriver">

<bean class="com.thoughtworks.xstream.io.xml.StaxDriver"/>

</property>

</bean>

</constructor-arg>

<property name="supportedMediaTypes" value="application/xml"></property>

</bean>

</mvc:message-converters>

</mvc:annotation-driven>

<!-- REST中根据URL后缀自动判定Content-Type及相应的View -->

<bean id="contentNegotiationManager" class="org.springframework.web.accept.ContentNegotiationManagerFactoryBean">

<property name="mediaTypes" >

<map>

<entry key="xml" value="application/xml"/>

<entry key="json" value="application/json"/>

</map>

</property>

<property name="ignoreAcceptHeader" value="true"/>

<property name="favorPathExtension" value="true"/>

</bean>

<!-- 对静态资源文件的访问， 将无法mapping到Controller的path交给default servlet handler处理 -->

<mvc:default-servlet-handler />

<!-- 静态资源映射 -->

<mvc:resources mapping="/static/\*\*" location="/static/" cache-period="31536000"/>

<!-- 定义无Controller的path<->view直接映射 -->

<mvc:view-controller path="/" view-name="redirect:/index"/>

<!-- 定义视图文件解析 -->

<bean class="org.springframework.web.servlet.view.InternalResourceViewResolver">

<property name="prefix" value="${web.prefix}"/>

<property name="suffix" value="${web.suffix}"/>

</bean>

<!-- 上传文件拦截，设置最大上传文件大小 10M=10\*1024\*1024(B)=10485760 bytes -->

<bean id="multipartResolver" class="org.springframework.web.multipart.commons.CommonsMultipartResolver">

<property name="maxUploadSize" value="${web.maxUploadSize}" />

</bean>

</beans>

6.在resources下建一个log4j.properties

内容如下：

# Output pattern : date [thread] priority category - message FATAL 0 ERROR 3 WARN 4 INFO 6 DEBUG 7

log4j.rootLogger=WARN, Console, RollingFile

#Console

log4j.appender.Console=org.apache.log4j.ConsoleAppender

log4j.appender.Console.layout=org.apache.log4j.PatternLayout

log4j.appender.Console.layout.ConversionPattern=%d %-5p [%c{5}] - %m%n

#RollingFile

log4j.appender.RollingFile=org.apache.log4j.DailyRollingFileAppender

log4j.appender.RollingFile.File=/logs/gmp/gmp.log

log4j.appender.RollingFile.layout=org.apache.log4j.PatternLayout

log4j.appender.RollingFile.layout.ConversionPattern=%d [%t] %-5p [%c] - %m%n

#log4j.logger.java.sql=DEBUG

log4j.logger.java.sql=DEBUG

log4j.logger.com.best.crm=DEBUG

log4j.logger.org.springframework=INFO

7.在resources下建一个yc.properties

内容如下：

web.prefix=/WEB-INF/views/

web.suffix=.jsp

web.maxUploadSize=10485760

8.将webapp目录下的webapp\WEB-INF\web.xml

改为：

<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://java.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd" id="WebApp\_ID" version="2.5">

<display-name>ddd</display-name>

<welcome-file-list>

<welcome-file>index.html</welcome-file>

<welcome-file>index.htm</welcome-file>

<welcome-file>index.jsp</welcome-file>

<welcome-file>default.html</welcome-file>

<welcome-file>default.htm</welcome-file>

<welcome-file>default.jsp</welcome-file>

</welcome-file-list>

<context-param>

<param-name>contextConfigLocation</param-name>

<param-value>classpath\*:/applicationContext\*.xml</param-value>

</context-param>

<listener>

<listener-class>org.springframework.web.context.ContextLoaderListener</listener-class>

</listener>

<listener>

<listener-class>org.springframework.web.context.request.RequestContextListener</listener-class>

</listener>

<listener>

<listener-class>org.springframework.web.context.ContextCleanupListener</listener-class>

</listener>

<filter>

<filter-name>encodingFilter</filter-name>

<filter-class>org.springframework.web.filter.CharacterEncodingFilter</filter-class>

<init-param>

<param-name>encoding</param-name>

<param-value>UTF-8</param-value>

</init-param>

<init-param>

<param-name>forceEncoding</param-name>

<param-value>true</param-value>

</init-param>

</filter>

<filter-mapping>

<filter-name>encodingFilter</filter-name>

<url-pattern>/\*</url-pattern>

</filter-mapping>

<servlet>

<servlet-name>crm-servlet</servlet-name>

<servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

<init-param>

<param-name>contextConfigLocation</param-name>

<param-value>classpath\*:/applicationContext-mvc.xml</param-value>

</init-param>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>crm-servlet</servlet-name>

<url-pattern>/</url-pattern>

</servlet-mapping>

</web-app>