|  |
| --- |
| 实施部署手册  V1.0 |

目 录

[1. 简介 2](#_Toc481092045)

[1.1. 编写目的 2](#_Toc481092046)

[2. 环境描述 2](#_Toc481092047)

[2.1. 操作系统及硬件、网络要求 2](#_Toc481092048)

[2.2. 软件列表 2](#_Toc481092049)

[2.3. 工程列表 3](#_Toc481092050)

[3. 基础应用部署 3](#_Toc481092051)

[3.1. Jdk和tomcat环境安装 3](#_Toc481092052)

[3.2. 数据库（MySQL）安装 3](#_Toc481092053)

[3.3. Memcached安装 3](#_Toc481092054)

[4. 应用部署 4](#_Toc481092055)

[4.1. 绩效考核应用部署 4](#_Toc481092056)

[4.1.1 数据结构初始化 4](#_Toc481092057)

[4.1.2 工程部署 4](#_Toc481092058)

[4.1.3 Tomcat配置 4](#_Toc481092059)

[4.1.4 创建资源存储路径 4](#_Toc481092060)

[4.1.5 工程必须配置 4](#_Toc481092061)

[4.1.6 启动tomcat服务 6](#_Toc481092062)

[4.2. 评教系统应用部署 6](#_Toc481092063)

[5. 优化配置*（可选）* 6](#_Toc481092064)

[5.1. 平台提供大量的可选配置，以优化系统。 6](#_Toc481092065)

[5.2. 动静分离 6](#_Toc481092066)

[5.3. jvm 优化。 7](#_Toc481092067)

[5.4. tomcat 优化 7](#_Toc481092068)

[5.5. 数据库优化。 8](#_Toc481092069)

[5.6. 系统优化。 8](#_Toc481092070)

[5.7. 集群优化 8](#_Toc481092071)

[6. 附录 9](#_Toc481092072)

[附录1: nginx 参考配置 9](#_Toc481092073)

# 简介

## 编写目的

本手册编写目的在于为学校教研系统提供部署帮助，以便运维、实施工程师能顺利完成系统的安装和部署，并确保系统能够正常运行。

文档由项目经理协调相关人员编写，应用于运维、实施工程师针对本文档相关内容进行生产环境部署测试。

# 环境描述

## 操作系统及硬件、网络要求

|  |  |  |  |
| --- | --- | --- | --- |
| **服务器** | **操作系统** | **硬件配置** | **网络需求** |
| 平台基础服务 | Cetos 6.6 | 8核8G 200G硬盘 | 可访问外网，对外映射80端口 |
| 数据库 | Cetos 6.6 | 4核8G 100G硬盘 | 可访问外网，纯内网环境 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## 软件列表

|  |  |
| --- | --- |
| **软件名称** | **版本** |
| Linux | Centos 6.6版本 |
| Tomcat | 7或更高 |
| Nginx（可选） | 1.5或更高 |
| JDK1.7 | 1.7或更高 |
| Mysql | 5.5 或更高 |

## 工程列表

|  |  |
| --- | --- |
| **工程包** | **工程说明** |
| tr-web-{版本号}.war | 学校教研前台 |
| tr-back-{版本号}.war | 学校教研后台 |

# 基础应用部署

## Jdk和tomcat环境安装

1. Jdk安装说明
2. Tomcat安装说明
3. 测试验证方式
4. 启动和关闭方式

## 数据库（MySQL）安装

1. MySQL安装说明
2. 用户创建
3. 导入

## Memcached安装

无

# 应用部署

安装基础软件环境后，解压应用程序包之后按以下步骤配置进行应用安装配置。

## 学校教研应用部署

* + 1. 数据结构初始化

本系统采用自动安装技术，在教研后台启动时将自动读取配置中的数据库配置，并执行数据库安装脚本。

* + 1. 工程部署

将两个工程war包放入tomcat webapps 目录下即可。如不想单独设置访问路径，可任意重命名war包名称。

如有特殊需求，可配置tomcat Context 自定义目录，或访问路径。

注意：如放置在tomcat webapps 目录下时，不可同时指定Context.

* + 1. Tomcat配置

server编码设置, 编辑tomcat安装目录/conf/server.xml 文件，在Connetor 元素上增加编码设置，如果<Connector URIEncoding="UTF-8"...

* + 1. 创建资源存储路径

在服务器硬盘上创建资源存储路径，资源存储路径所在硬盘需要有足够的空间。如 /opt/ftprooot/tr/

然后将 “预制教案模版，放资源根目录” 文件夹下的template 文件夹放置到以上目录下。 完成后效果如： /opt/ftprooot/tr/template

* + 1. pageoffice 注册

学校教研前台会使用到pageoffice 插件， 该插件需要有注册码，且注册码只能在一台服务器上使用。注册步骤如下：

1. 注册码需向pageoffice申请pageoffice 注册码。
2. 系统启动后台访问，[http://ip:port/{context}/p.jsp](http://ip:port/%7bcontext%7d/p.jsp), ip和port分别为系统部署服务器ip（或域名）和端口。context 为tr-web的虚拟目录。
3. 按照提示按照pageoffice 插件，如已安装，跳过此步骤。
4. 在弹出的注册框里输入注册码，确定即可。
   * 1. 配置中心配置
5. 配置文件。如下表：

|  |  |
| --- | --- |
| **项目** | **位置** |
| tr-web | WEB-INF\classes\config\init\application.properties |
| tr-back | WEB-INF\classes\config\init\application.properties |

**以下为各配置详情：**

**注意：有背景颜色的配置为环境相关配置需要特别注意，其他为优化或个性化配置。**

##### tr-web

一般只需修改数据库和文件存储相关设置。

前台配置项说明如下：

#数据库配置

jdbc.dbtype=mysql

jdbc.driver=com.mysql.jdbc.Driver

jdbc.url=jdbc:mysql://192.168.0.233:3306/jypt-online

jdbc.username=root

jdbc.password=ma10b

#资源存储根目录

local\_file\_directory=/opt/ftproot/jypt

#学年分割时间点

schoolyear\_split=07-11

#web 容器url编码

container\_url\_encode=utf-8

#数据库链接池配置,可根据预估用户数优化配置

druid.initialSize=2 //初始化连接数量

druid.minIdle=2 //最小空闲连接数

druid.maxActive=10 // 最大并发连接数

druid.maxWait=60000 //配置获取连接等待超时的时间

druid.timeBetweenEvictionRunsMillis=60000 //配置间隔多久才进行一次检测，检测需要关闭的空闲连接，单位是毫秒

druid.minEvictableIdleTimeMillis=300000 //配置一个连接在池中最小生存的时间，单位是毫秒

druid.validationQuery=SELECT 'x'

druid.testWhileIdle=true //申请连接的时候检测

druid.testOnBorrow=false //申请连接时执行validationQuery检测连接是否有效，配置为true会降低性能

druid.testOnReturn=false //归还连接时执行validationQuery检测连接是否有效，配置为true会降低性能

druid.poolPreparedStatements=false //打开PSCache，并且指定每个连接上PSCache的大小

druid.maxPoolPreparedStatementPerConnectionSize=20 //最大每个连接上PSCache的大小

druid.filters=stat,slf4j,wall,mergeStat //配置扩展插件

###### tr-back

后台配置项，特殊配置说明如下：

#前端的web 访问地址

front\_web\_url=http://jy.myuclass.com

#前端的web 根目录路径(后台)

front\_web\_root = /opt/jypt/front

#历史记录定时任务表达式，需与jyapp里的学年配置schoolyear\_split保持一致，以下代表7月11.

history\_split\_cron=0 0 0 11 7 ?

webAppRootKey = jypt.back.root

appid = 3249684c2d85490db076246c53fb12b2

appKey = 85e8efbec4a3472f9a06c7d6d2854afd

* + 1. 启动tomcat服务

启动tomcat，linux 平台执行 startup.sh ； windows 平台安装为tomcat 服务并启动。

# 优化配置*（可选）*

优化方案需要根据部署环境，及预估用户数等实际情况来选定。这里简单介绍部署时可进行的优化。

## 平台提供大量的可选配置，以优化系统。

配置详细说明参考《平台配置》文档。

## 动静分离

使用nginx 做反向代理，负责处理静态文件。

目前应用中静态文件统一放在static/ 目录中，在nginx 做相应的配置即可，参考配置见附录1。

## jvm 优化。

优化jvm 参数，设置合理的内存及垃圾回收机制。

参考配置：

-server -Xms2048m -Xmx2048m -XX:PermSize=256m -XX:MaxPermSize=512m -Xss512k -XX:+UseParNewGC-XX:+UseConcMarkSweepGC -XX:+CMSParallelRemarkEnabled -XX:SurvivorRatio=20 -XX:ParallelGCThreads=16 -XX:+DisableExplicitGC -XX:+AggressiveOpts

## tomcat 优化

优化tomcat 参数，linux 可以考虑使用apr。

参考配置： <Connector protocol="org.apache.coyote.http11.Http11AprProtocol" URIEncoding="UTF-8" port="8088" maxThreads="400" minSpareThreads="64" enableLookups="false" processorCache="256" acceptCount="1000" disableUploadTimeout="true" maxKeepAliveRequests="150" connectionTimeout="30000" redirectPort="8444" maxConnections="10240" />

Linux Apr 安装：

1. 按Openssl 安装

tar zxvf openssl-1.0.2h.tar.gz

cd openssl-1.0.2h

./config 【64位系统，增加参数-fPIC enable-shared】

make depend

make install

1. 安装apr

tar zxvf apr-1.5.1.tar.gz

cd apr-1.5.1

./configure

make

make install

1. Tomcat-native 安装

cd apache-tomcat-7.xx/bin/

tar zxvf tomcat-native.tar.gz

cd tomcat-native-1.1.24-src/jni/native/

./configure --prefix=/usr/locat/tom-native --with-apr=/usr/local/apr --with-ssl=/usr/local/ssl

make

make install

1. 配置

Edit $CATALINA\_HOME/bin/setenv.sh (creating the file if necessary) and add the path to the tc-native libraries to LD\_LIBRARY\_PATH. Something like:

LD\_LIBRARY\_PATH=$LD\_LIBRARY\_PATH:$CATALINA\_HOME/lib

export LD\_LIBRARY\_PATH

## 数据库优化。

完善索引、优化数据库链接、日志等相关参数。

## 系统优化。

优化操作系统网络环境，文件读写等。

## 集群优化

tomcat 集群，设计到的集群配置有：

1. Session 共享。建议使用tomcat-redis-session 插件，将session 存储到redis 中，redis 插件需要定制版本，以支持sevlet 3.0 。
2. 缓存集群。 项目使用ehcahe 做为缓存，需要在缓存配置文件中增加集群配置。
3. 存储集群。 使用阿里云存储方案，修改默认存储设置

# 附录

## 附录1: nginx 参考配置

#user nobody;

worker\_processes 32; #请根据你的服务器实际核数调整，一般是实际核数\*2

#error\_log logs/error.log;

#error\_log logs/error.log notice;

#error\_log logs/error.log info;

#pid logs/nginx.pid;

events {

#worker\_connections 1024;

worker\_connections 10000;

multi\_accept on;

use epoll;

#worker\_connections 10000;

}

http {

include mime.types;

default\_type application/octet-stream;

log\_format main '$remote\_addr - $remote\_user [$time\_local] "$request" '

'$status $body\_bytes\_sent "$http\_referer" '

'"$http\_user\_agent" $http\_x\_forwarded\_for';

access\_log logs/access.log main;

access\_log off;

sendfile on;

tcp\_nopush on;

tcp\_nodelay on;

#send\_timeout 3m;

#send\_timeout 9m;

send\_timeout 10;

client\_header\_timeout 20;

client\_body\_timeout 20;

reset\_timedout\_connection on;

open\_file\_cache max=100000 inactive=20s;

open\_file\_cache\_valid 30s;

open\_file\_cache\_min\_uses 2;

open\_file\_cache\_errors on;

source\_charset utf-8;

client\_max\_body\_size 50m;

keepalive\_timeout 32s;

#keepalive\_timeout 220;

upstream tomcat {

server 127.0.0.1:8080 max\_fails=100 fail\_timeout=3s; #tomcat

keepalive 12;

}

#gzip on;

server {

listen 80;

server\_name jy.myuclass.com;

#charset koi8-r;

access\_log logs/access.log main;

#location = / {

# proxy\_pass http://127.0.0.1:8088;

#}

location ^~ /static/{

root /opt/jypt/;

}

location ^~ /back/static/{

alias /opt/jypt/back/static/;

}

location / {

#root /opt/jypt/update/;

#rewrite ^(.\*)\.png$ /bg.png break;

#rewrite ^(.\*)$ /system.html break;

proxy\_pass http://tomcat;

#Proxy Settings

proxy\_redirect off;

proxy\_set\_header Host jy.myuclass.com; #$host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_next\_upstream error timeout invalid\_header http\_500 http\_503;

proxy\_max\_temp\_file\_size 0;

proxy\_connect\_timeout 75s;

proxy\_send\_timeout 300s;

proxy\_read\_timeout 200s;

proxy\_buffer\_size 256k;

proxy\_buffers 4 256k;

proxy\_busy\_buffers\_size 256k;

proxy\_temp\_file\_write\_size 256k;

}

#error\_page 404 /opt/ftproot/update/system.html;

# redirect server error pages to the static page /50x.html

#

error\_page 500 502 503 504 /50x.html;

location = /50x.html {

root html;

}

# proxy the PHP scripts to Apache listening on 127.0.0.1:80

#

#location ~ \.php$ {

# proxy\_pass http://127.0.0.1;

#}

# pass the PHP scripts to FastCGI server listening on 127.0.0.1:9000

#

#location ~ \.php$ {

# root html;

# fastcgi\_pass 127.0.0.1:9000;

# fastcgi\_index index.php;

# fastcgi\_param SCRIPT\_FILENAME /scripts$fastcgi\_script\_name;

# include fastcgi\_params;

#}

# deny access to .htaccess files, if Apache's document root

# concurs with nginx's one

#

#location ~ /\.ht {

# deny all;

#}

}