# 系统架构

1. xx-serve 里面是不会含有配置文件的，配置文件统一放入xx-web
2. /mybatis 放mybatis相关配置如缓存，与spring集成的部分在spring配置中

/resource 放系统需要配置的地址等参数数据，日志文件除外

/spring 放与spring集成需要的配置

1. 访问后返回的结果都是 json对象，但是前端还是需要以下方法转换

//var result = JSON.parse(result);//Json对象

var result = $.parseJSON(result);//jq对象

# xml语句注意

1. xml中有关"创建时间"的默认值，修改xml和修改数据库(加上默认值)
2. 如果有like语句，修改相应xml的语句。

如 name like CONCAT('%',#{name},'%')

1. 如果涉及时间的比较（范围）修改修改相应xml的语句。

sbdate <![CDATA[ >= ]]> #{starttime}

或者sbdate <![CDATA[ <= ]]> #{endtime}

1. 有字段需要翻译时，采用left join
2. 主键生成方式：不是高并发采用int，高并发采用uuid
3. 对于表中需要转换的字段，使用另一个字段来存，避免修改带来的麻烦

如sys\_user 表中的deptid 列表显示时 翻译为 deptname

1. 注意mysql数据库的关键字不要出现在表中或者当做表名

如desc, database，order

1. 关于批量操作的性能问题

insert 最好是每10-15条一组，使用foreach批量操作

delete 最好采用foreach批量操作，单条循环执行会存在连接性能问题

update 同delete

1. 。。。

# Java部分

1. xxDate.toLocaleString() 这个少用，不兼容系统，建议采用日期转换器

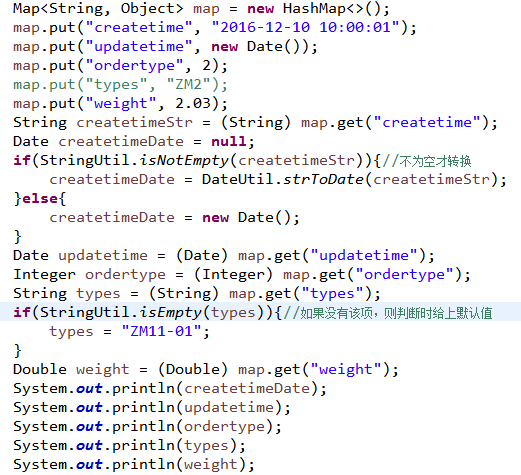
工具类DateUtil.*strToDate*((String) map.get("createtime"));

1. 如果列表展示的时间格式有问题，可以考虑在model的相应字段加上注解

@JSONField(format="yyyy-MM-dd HH:mm:ss")

对应的包 import com.alibaba.fastjson.annotation.JSONField;

1. 获取Map<String, Object>对象的值直接使用强转



1. 。。。
2. @Transactional 手动触发回滚需要是RuntimeException或其子类

**throw** **new** RuntimeException("不能删除");

（Transactional的异常控制，默认是Check Exception 不回滚，unCheck Exception回滚）

1. 。。。

# Java拓展部分

1. 。。。
2. 。。。
3. 。。。

# easyui部分

1. 使用js方式时,需要渲染的组件放在$(function(){});里面，避免出现渲染有bug
2. 。。。

# 插件拓展部分

1. 。。。
2. 。。。
3. 。。。