## 设计思想

对标准的HTML的元素进行整合，在页面上进行分区。用java对象的方式去操作这些页面组件。让业务开发人员彻底的摆脱调整页面格式的枯燥，同时，让整个系统风格看起来更加的一致。尽量不要用size，length等来设置页面元素的格式，尽量用系统标准的class，如果系统标准的class满足不了需求，自定义class，然后，addClass

### 界面逻辑：

一个block可以有最多一个commandbar，commandbar的内容多了怎么办？系统标准提供的command有：增删改查，便签，操作，界面配置，分页操作;

除了标准的command之外，用户还可以定义自己的command，这些命令默认会增加到操作中去，不在commandbar上显示，操作的内容在多行模式下显示选中的行共同支持的命令，取交集。单记录模式下显示该行所有的客户添加的命令。当用户添加的命令中包含icon信息，才会放到commandbar上？（还是增加一个专门的属性来判断呢）增加这个客户方法要么在commandbar上显示，要么在操作里显示，二选一。还要支持动态显示与否的判断条件。

command支持分组，支持客户化方法，包括server端java方法和客户端js代码。

tab依赖不依赖于block呢？先不依赖吧。

## 遇到的技术问题

1. 如何保证输出的键值顺序是put到map的顺序呢？采用LinkedHashMap保证顺序: field, commandbaritem, tab。
2. java格式化数据包是哪个？java格式化数据java.text.\*

### 设计模式：

总管（门面，外观）：屏蔽子系统的复杂关系，对外仅仅提供简单的接口

观察者模式：

在创建对象的时候，就将新创建的对象放到订阅者手中。

**public** ASPBlock newASPBlock(String name, String viewName){

ASPBlock block = **new** ASPBlock(**this**, name, viewName);

blocksMap.put(name, block);

**return** block;

}

这要比创建完成之后，再把对象放进来要好的多。节约代码量。避免遗漏。

### 如何在maven管理的项目里添加j2ee依赖关系。

<!-- j2ee -->

<dependency>

<groupId>javax</groupId>

<artifactId>javaee-api</artifactId>

<version>6.0</version>

<scope>runtime</scope>

</dependency>

## 日期控件整合：

采用datetimepicker日期控件

增加两个文件：

jquery.datetimepicker.css

jquery.datetimepicker.js

## 树形插件整合

采用ZTree插件。后台能不能设计成不依赖于前端框架的实现呢？

新增文件

jquery.zTreeStyle.css

jquery.ztree.all-3.5.js

图片

line\_conn.gif

loading.gif

zTreeStandard.gif

zTreeStandard.png

实现起来超级简单。待深入学习。

## Knockout.js

前台的数据绑定框架。研究一下，准备使用。

## 新增/编辑模式：

输入框与后面的值列表图片对不齐：解决方法：定义一个class

*.shortcut-img*{

vertical-align: *top*;

}

**系统采用什么方式来支持国际化**

包名前缀，后面的字符串加语言代码，默认是英文的。是主键；"ASP: view name can not be null."

**采用什么方式来支持授权？**

**事务如何管理？**

**日志**

日志采用slf4j来管理。

http://www.ilifedrops.com/?p=278

## 系统约定：

1. 字符串替换一律采用{1}，{2}这种形式

## Js面向对象开发：

值列表用到了js中的class定义。

参考文件：/bingo/src/main/webapp/js/bingo.lov.js

语法：花括号键值对

**思考**：什么时用javascript的class，什么时候用对象？当一个页面就只有一个实例的时候可以考虑定义对象。