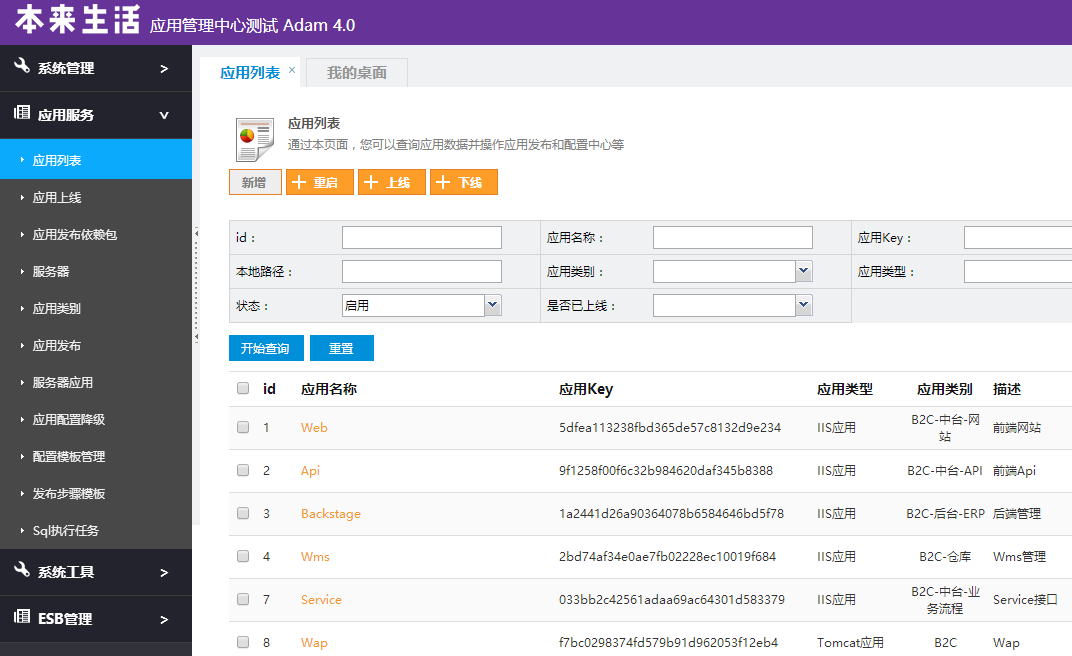
# 本来生活应用管理中心测试

<http://plf.k.benlailaile.com/default.aspx>



## 应用服务

### 应用列表

#### 应用重启

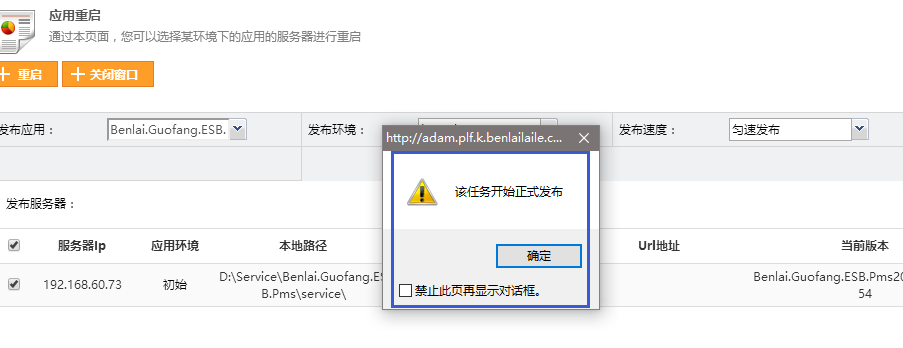
1)查看应用列表信息



2)应用重启

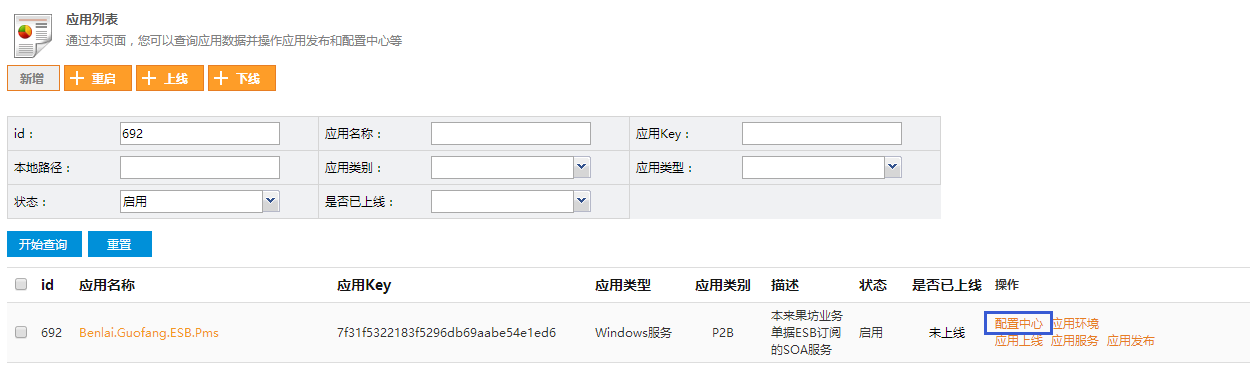


3)已重启



#### 配置中心配置

1)查看配置中心



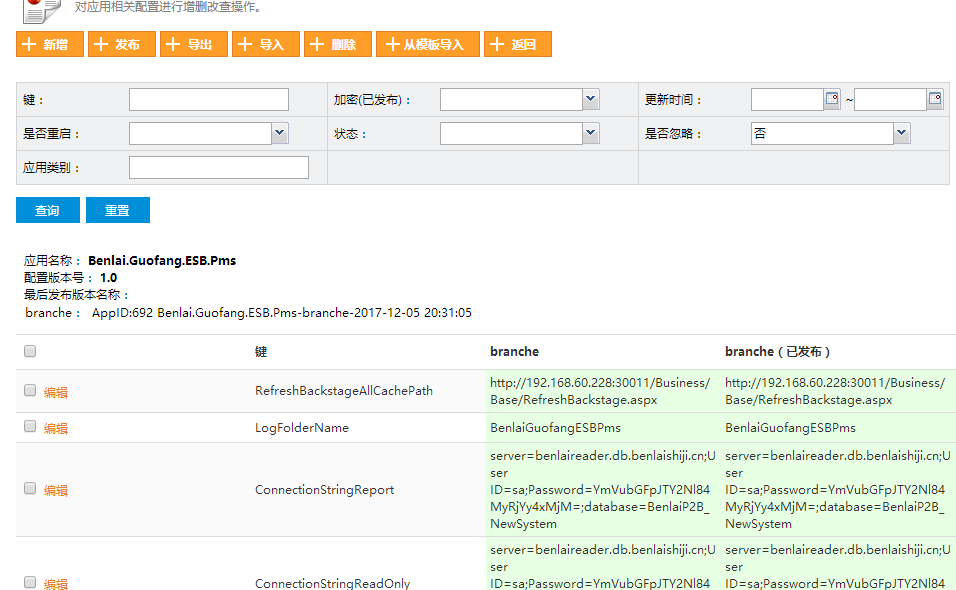
2)点击配置管理



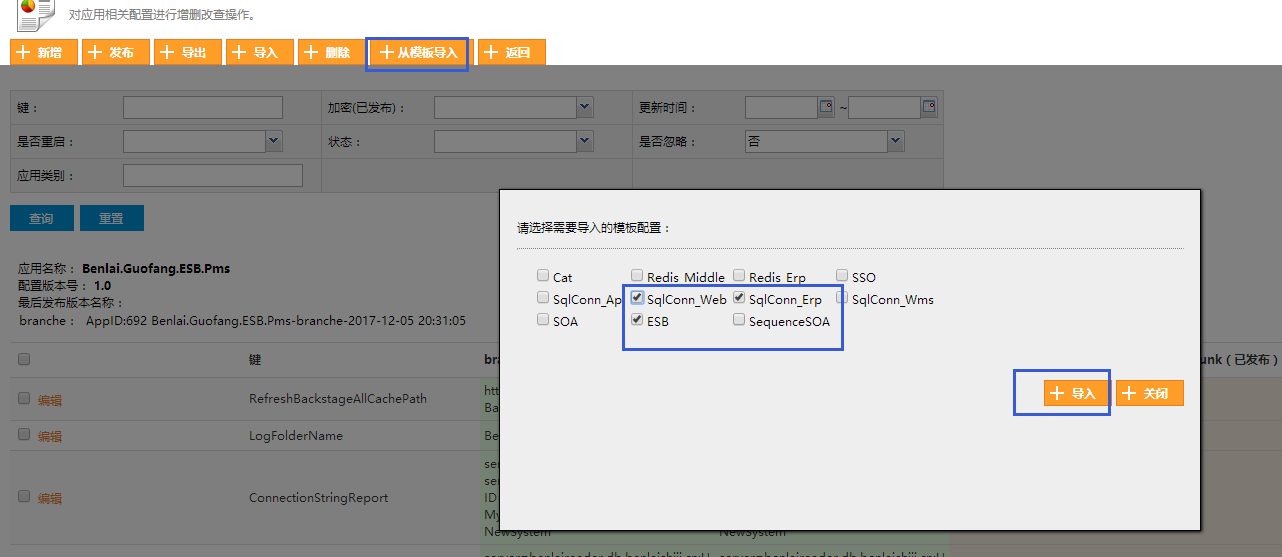
3)查看该应用所有的配置，该配置对应了ConfigManager.GetConfigValue所有的Key

如果应用程序中有用到ConfigManager.GetConfigValue，且没有在此配置的一定需要在此进行配置。

如果应用程序需要添加一个资源配置同样在此配置一下，发布即可



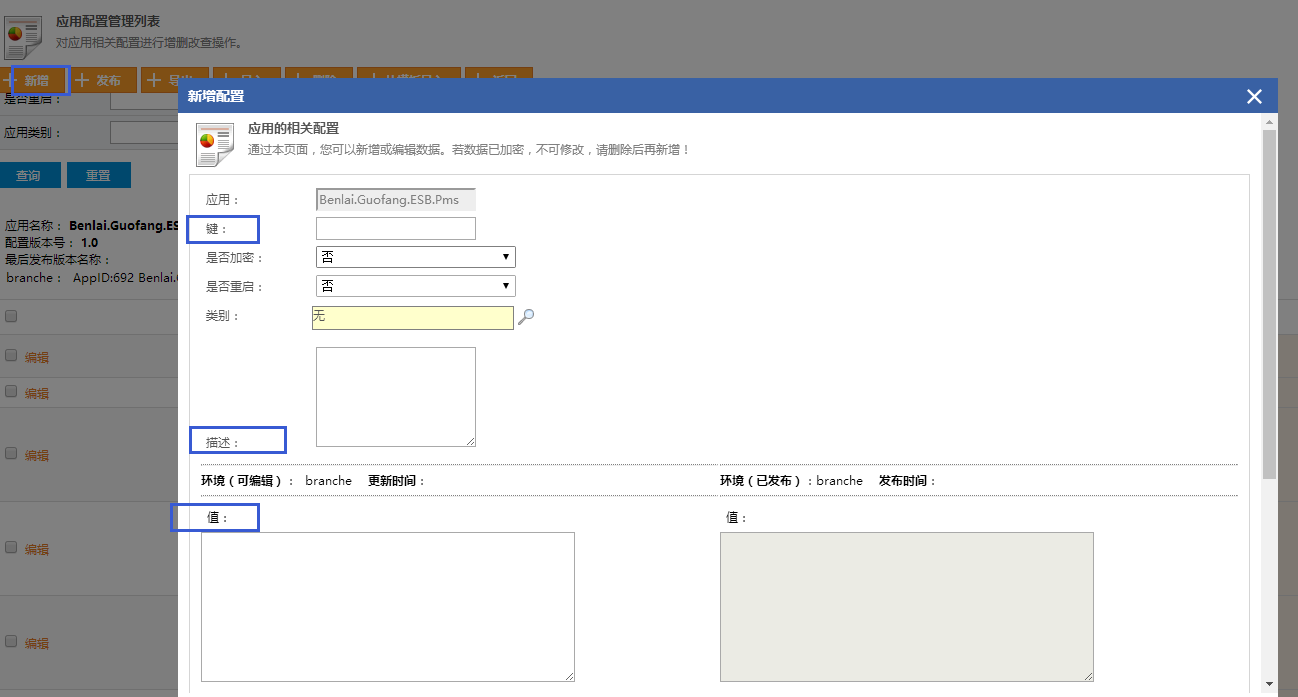
4)初次配置时选择模板导入



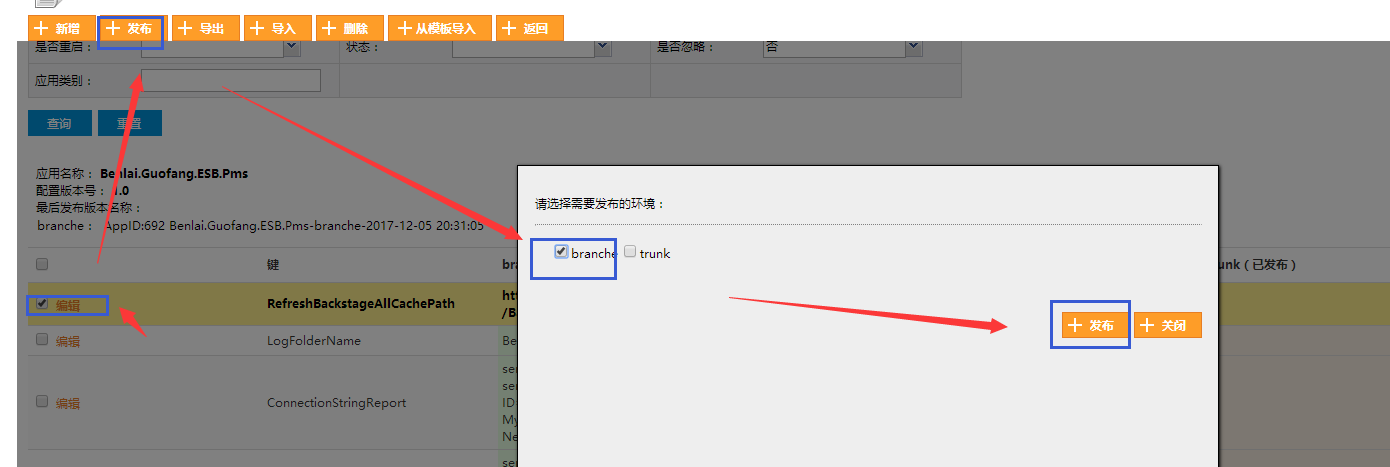
5)数据库连接设置不加密，1为不加密，加密方式为MD5



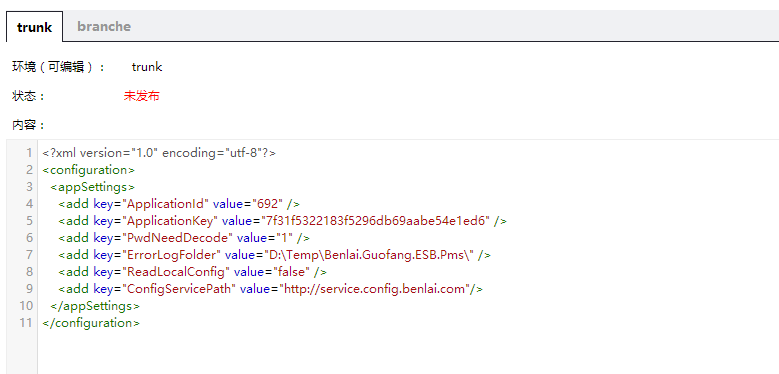
6)新增一个配置，设置键、值和描述



7)新增的配置需要发布之后才能有效应用,选择发布的环境进行发布

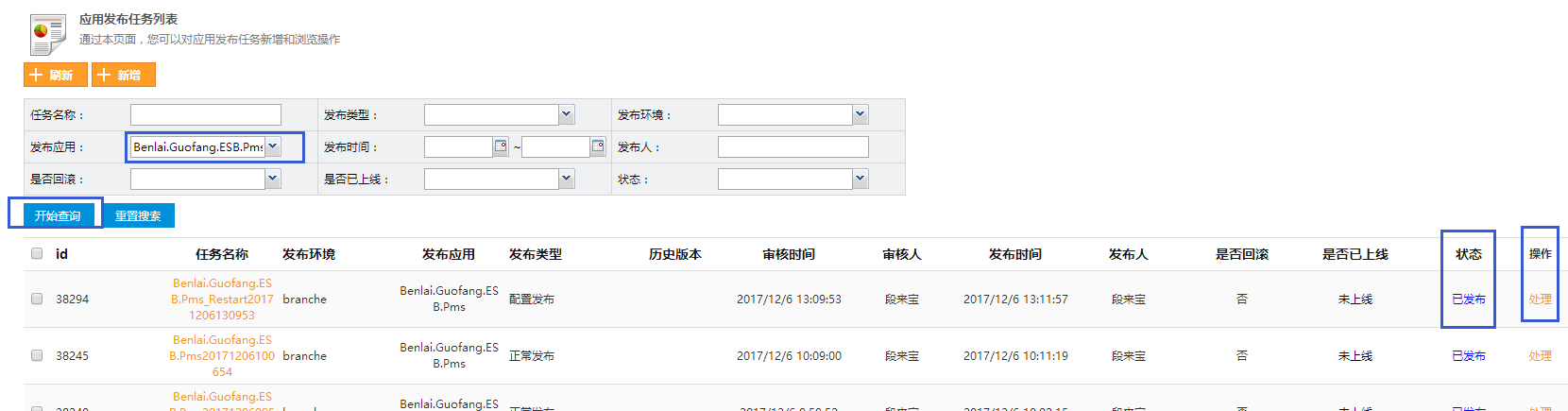


8）查看配置中心的文件信息,branchs和trunk配置均在此进行相关配置



### 应用发布

1)查看发布信息,选择发布应用进行开始查询，可以查看项目对应的发布状态和操作



2)发布应用，点击新增，选择发布应用和发布环境，保存更新，然后回到应用发布任务列表，找到刚刚新建的任务，点击操作项，选择提交任务，然后发布任务即可。

如果已经存在发布的任务，未发布成功，可以作废改发布后重新发布。

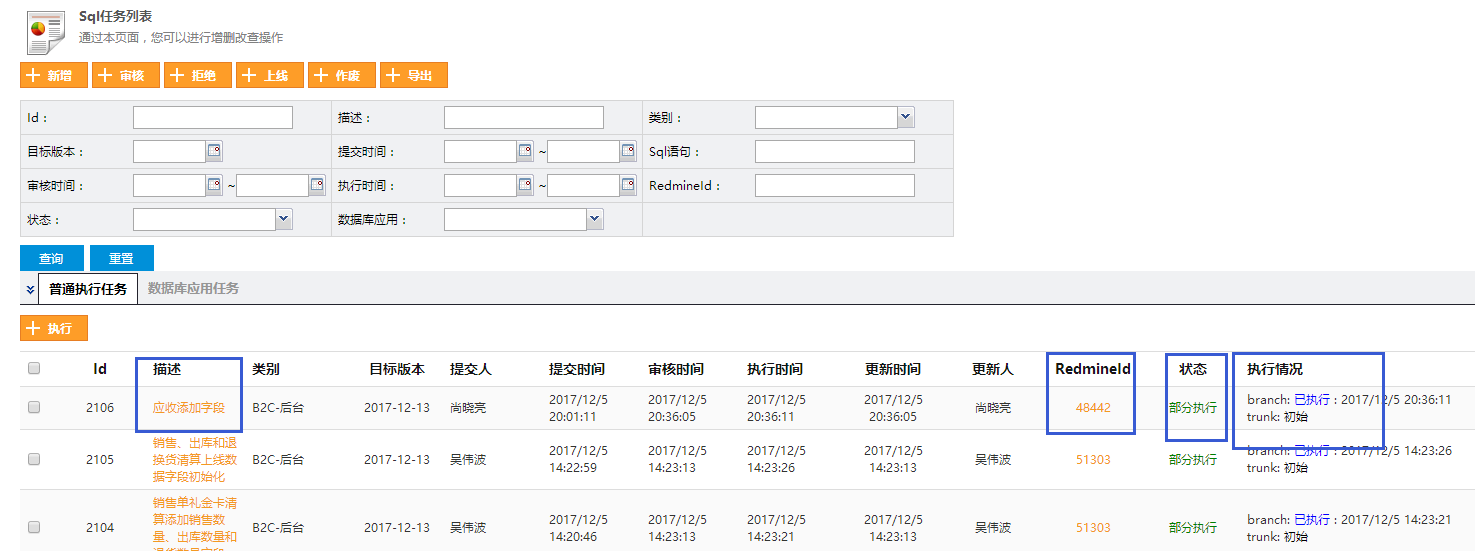
发布时会先启动和停止服务，然后在[http://192.168.60.31/](http://192.168.60.31/%20)  Jenkins里面会自动进行编译，编译完成之后正式发布。



### SQL执行任务

所有的测试环境执行sql脚本，在该菜单中进行提交，DBA审核后测试环境执行

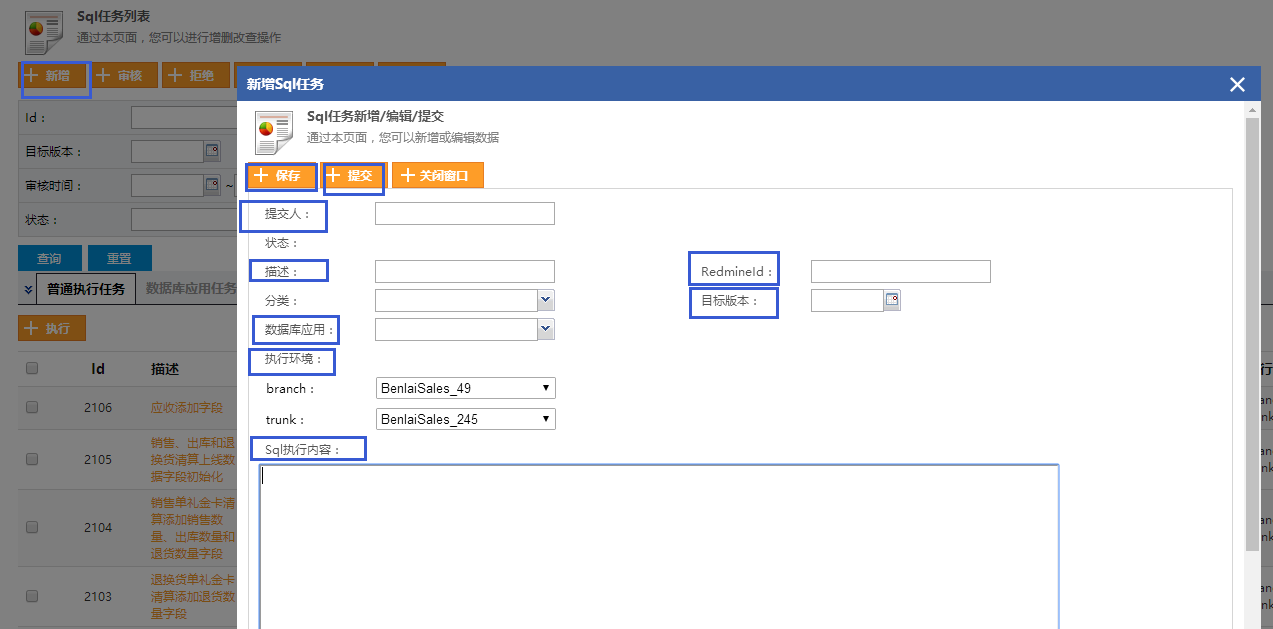
1)Sql执行任务列表，可以查看sql的描述、redemine编号，以及执行状态等等



2)Sql执行任务添加

在列表点击新增，打开详情页面，输入必须的字段提交人、描述、数据库应用、执行环境、redmine编号、目标版本、以及可执行的sql执行内容,

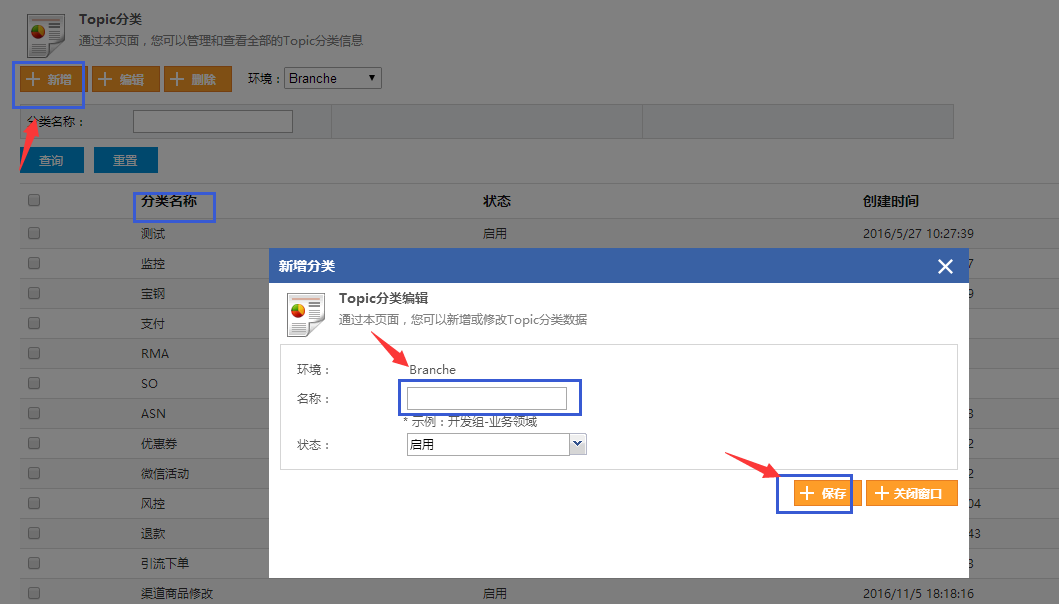
然后保存、提交即可



## ESB管理

### Topic分类

Topic分类，指定了Topic的管理类别,在列表页可以新增



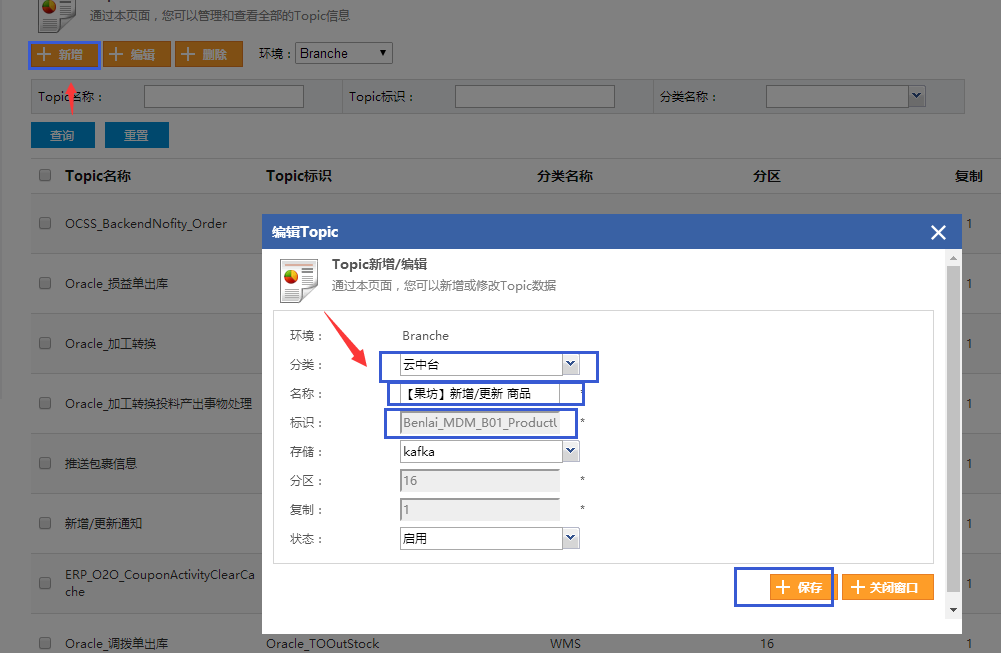
### Topic管理

Topic管理，所有的订阅或者发布的Topic都需要在此进行添加。我们的订阅的Topic和发布的Topic是一个Topic



新增一个Topic，需要选择Topic分类，然后输入Topic名称即Topic用来做什么，输入标识，即Topic标识的Key值，保存即可

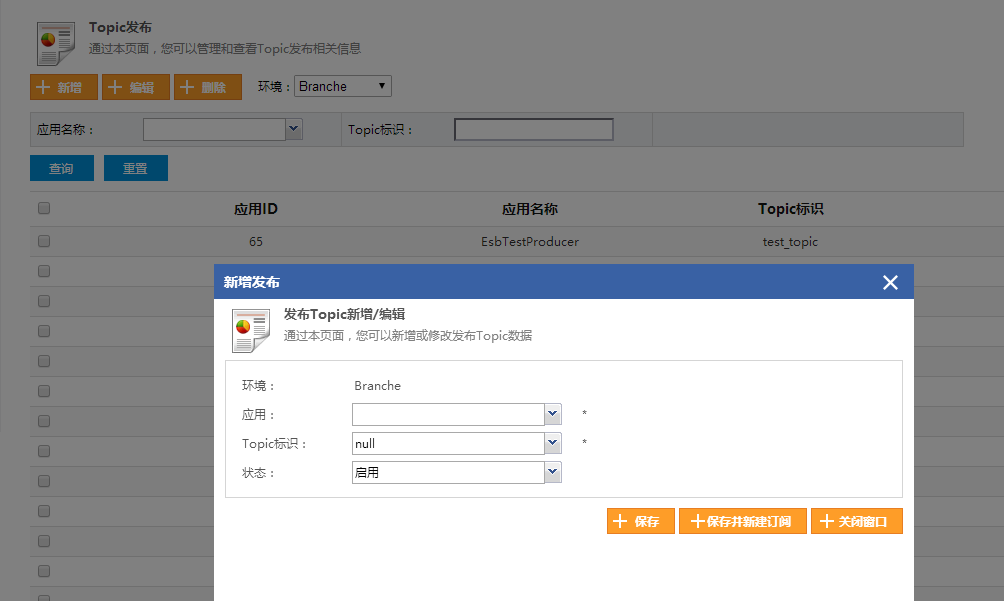
应用程序中的静态资源则需要通过该Topic标识的Key值来实现订阅和发布



### Topic发布

发布，在Topic管理里面新建的Topic，把它和发布的应用程序关联起来实现不断的发布。

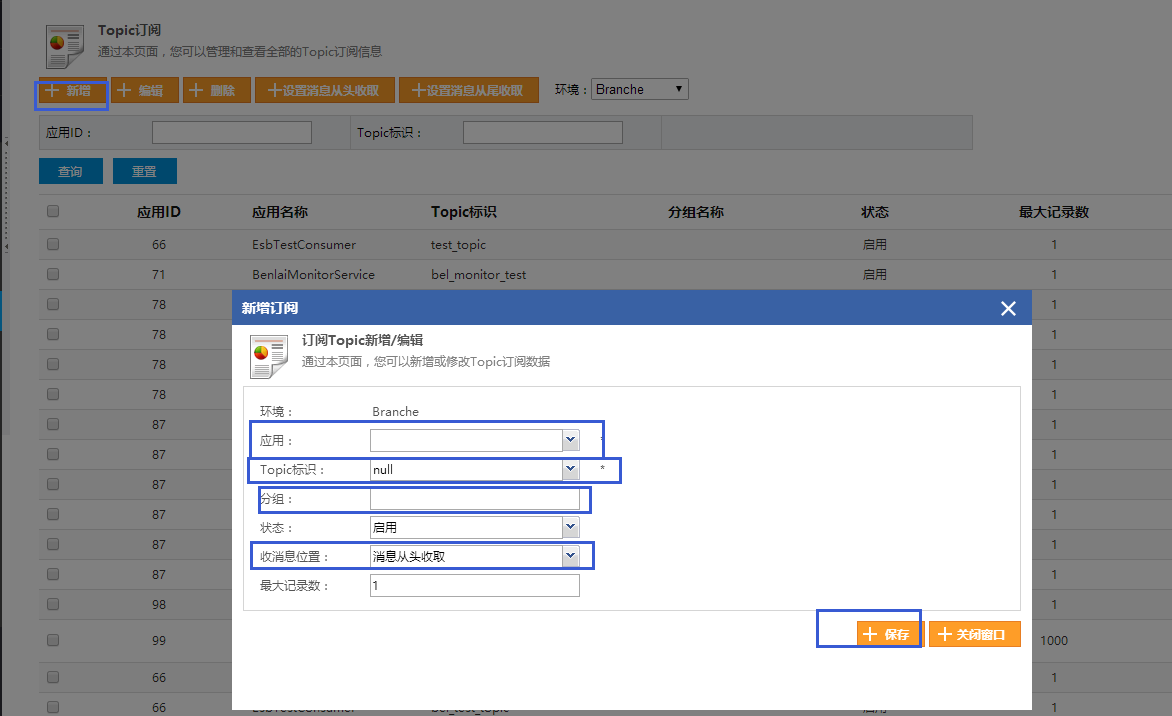
这里需要获得发布的应用程序以及在Topic管理里面新建的Topic，订阅的Topic和发布的Topic需要保持一致方能实现订阅发布。



### Topic订阅

订阅，在Topic管理里面新建的Topic，把它和订阅的应用程序关联起来实现不断的订阅。

这里需要获得订阅的应用程序以及在Topic管理里面新建的Topic，订阅的Topic和发布的Topic需要保持一致方能实现订阅发布。



### Topic错误信息

我们在订阅之后可能会出现相关的问题，比如说发布应用已经正常的发布信息给订阅应用，但是订阅应用没有获得任何消息。这时候可能出现的2种情况是：

1. Topic不正确，会出现订阅无法连接的问题
2. Execute方法的模型不正确，会出现序列化的问题
3. Topic进行了分组，而建立订阅的连接时没有设置分组，导致了订阅失败的问题

对于第1种情况，我们检查发布和订阅的Topic是否一致，然后再检查订阅或者发布的应用连接的Topic是否正确

var client = new BenlaiQueueConsumerClient(base.AppId, base.AppKey, this.TopicName);// TopicName是订阅的Topic AppId是订阅的应用程序AppId, AppKey是订阅的应用程序AppKey

client.Use(typeof(MeasurementUnitProcess));// MeasurementUnitProcess是订阅信息

client.Start();

clientList.Add(client);

对于第2中情况，我们检查订阅应用的服务器日志，查看里面的Error信息，解决好即可

对于第3种情况，我们检查订阅Topic如果有分组，那么在订阅应用里面需要加上分组如下：

var client = new BenlaiQueueConsumerClient(base.AppId, base.AppKey, this.TopicName);// TopicName是订阅的Topic AppId是订阅的应用程序AppId, AppKey是订阅的应用程序AppKey

client.Use(typeof(MeasurementUnitProcess));// MeasurementUnitProcess是订阅信息

ConsumerOptions consumer = new ConsumerOptions

{

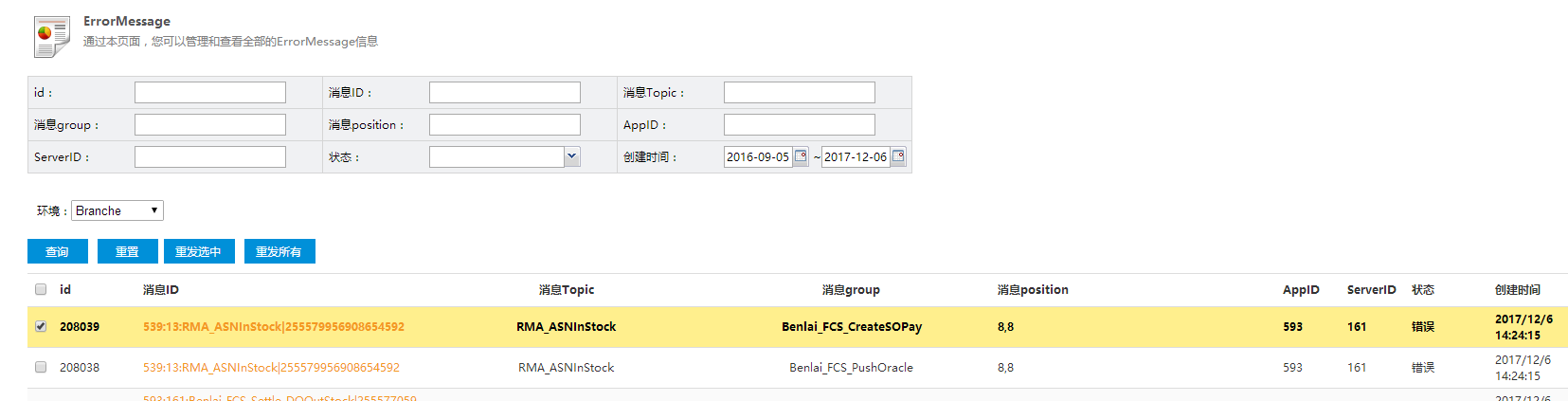
GroupId = ESBTopicGroup.DOOutStock.CreateARAP.ToString(),

};

client.Start(consumer);

clientList.Add(client);

当这些问题解决好之后，需要编译订阅或者发布应用，然后选择消息点击重发选中即可.



## 订阅流程

* 建立订阅的ESB应用，该应用当部署为windows服务,而非SOA
* 应用启动后通过异步任务加载订阅连接,设置配置的应用程序AppId和AppKey,以及Topic
* 并且设置订阅信息(即发布后的对象)
* 处理订阅信息
* 更新并通知ERP(可选)
* 在管理中心创建应用获得AppId和AppKey
* 设置配置中心配置，发布配置
* 应用发布
* 在Topic管理创建Topic，然后查看Topic发布是否已经为该Topic进行了发布，如果没有发布，则创建一个Topic发布。
* 然后再Topic订阅菜单，新建一个Topic订阅
* 应用重启

如发现文档描述与实际不符，麻烦更新一下上传。谢谢