**GWT中调用RPC方法**

**（1）在shared的intf包中定义接口和异步接口**

服务接口中可用单例方式获取异步接口hk.guosen.sha.gwt.shared.intf. HandleAllocationSvc

*@RemoteServiceRelativePath("rpc/handleAllocationSvc")*

*public interface HandleAllocationSvc extends RemoteService {*

*public static class HandleAllocationSvcUtil {*

*private static HandleAllocationSvcAsync instance;*

*public static HandleAllocationSvcAsync getInstance(){*

*if (instance == null) {*

*instance = GWT.create(HandleAllocationSvc.class);*

*}*

*return instance;*

*}*

*}*

*/\*\**

*\* 生成Allocation*

*\*/*

*public Map<String,Object> createAllocation();*

*}*

异步接口 hk.guosen.sha.gwt.shared.intf. HandleAllocationSvcAsync

*public interface HandleAllocationSvcAsync {*

*void createAllocation(AsyncCallback<Map<String, Object>> callback);*

*}*

**（2）实现类**

*HandleAllocationSvcImpl.* *HandleAllocationSvcImpl*

*public class HandleAllocationSvcImpl implements HandleAllocationSvc {*

**（3）在Spring中配置实现接口，id需与接口定义的id一致**

*<bean id="handleAllocationSvc"*

*class="hk.guosen.sha.gwt.server.HandleAllocationSvcImpl">*

*</bean>*

**（4）在页面上调用RPC接口**

获得异步接口的引用

*final HandleAllocationSvcAsync svc = HandleAllocationSvcUtil.getInstance();*

调用接口

*buttonDataGather.addClickHandler(new ClickHandler() {*

*public void onClick(ClickEvent event) {*

*svc.createAllocation(new AsyncCallback<Map<String, Object>>() {*

*public void onFailure(Throwable paramThrowable) {*

*MessageBox.alert("消息提示", "生成文件有误！", null);*

*}*

*public void onSuccess(Map<String, Object> paramT) {*

*if(paramT!=null){*

*MessageBox.alert("消息提示", (String)paramT.get("message"), null);*

*}else{*

*MessageBox.alert("消息提示", "生成文件有误！", null);*

*}*

*}*

*});*

*}*

*});*