**fineReport简易使用**

[1. 技术简介 2](#_Toc328657511)

[1.1 基本结构简单 2](#_Toc328657512)

[1.2 远程设计 2](#_Toc328657513)

[1.3 多种数据源的支持和使用 2](#_Toc328657514)

[1.4 图表展现 3](#_Toc328657515)

[1.5 数据分析 3](#_Toc328657516)

[1.6 Web环境的即席报表 3](#_Toc328657517)

[1.7 在线构建数据透视表 3](#_Toc328657518)

[1.8 零客户端数据填报 3](#_Toc328657519)

[1.9 Flash打印 3](#_Toc328657520)

[1.10 多种输出方式 3](#_Toc328657521)

[1.11 参数获取与定时任务 4](#_Toc328657522)

[1.12 B/S资源管理方式 4](#_Toc328657523)

[1.13 数据实时更新 4](#_Toc328657524)

[2. 开发目录介绍 4](#_Toc328657525)

[3. Finereport 集成 5](#_Toc328657526)

[3.1 独立部署 5](#_Toc328657527)

[3.2 嵌入式部署 5](#_Toc328657528)

[3.2.1 复制目录 5](#_Toc328657529)

[3.2.2 整合web.xml文件 6](#_Toc328657530)

[3.3 检测是否部署成功 7](#_Toc328657531)

[4. 使用方法 8](#_Toc328657532)

[4.1 普通报表 8](#_Toc328657533)

[4.2 参数报表 11](#_Toc328657534)

[4.3 图表报表 12](#_Toc328657535)

[5. 外部改变数据源 14](#_Toc328657536)

# 技术简介

有了FineReport，做这些基于数据库的展示，查询和录入的页面，不再需要编写代码了；而是通过FineReport提供的模板设计器，拖拖拽拽就制作出实现数据展示(报表)，数据查询(参数)和数据录入(表单)功能的页面模板，然后将这些模板保存到FineReport服务器指定的目录下面。当最终用户通过浏览器远程访问某个页面模板，FineReport服务器会把用户指定的页面模板自动解析成零客户端的html页面给浏览器，这样用户就可以在浏览器里面查看或录入数据了。同时由于FineReport的易用性、傻瓜性，可以在项目交付后把FineReport这个工具软件提供给最终用户，让他们可以基于FineReport做简单的页面修改以达到自我维护和更新系统的目的。

## 基本结构简单

FineReport采用Excel形式的设计界面，全面兼容Excel的设计方式，Excel文件的无失真导入导出。

## 远程设计

FineReport支持对部署在服务器上的报表环境进行远程连接后直接进行报表的设计，避免报表从设计到部署过程中的一些意外情况发生。

## 多种数据源的支持和使用

支持多种数据源，且支持多个数据源的同时呈现。如数据库数据源、文本数据源、程序数据源等。定义好数据源之后通过简单的拖拉和设置即可实现在报表中引用。

## 图表展现

FineReport提供了设计风格类似Excel的图表设计工具，同时支持在线的图表样式切换。支持地图的数据分析，以更直观感性的展现数据。

## 数据分析

通过使用控件、参数面板并结合图表，可以实现强大的数据分析功能和对数据多样式的展现。

## Web环境的即席报表

无需了解后台的执行逻辑，只需要完成对所使用的数据源的定义即可实现在线的报表生成和导出打印等功能。

## 在线构建数据透视表

在浏览器端对大量数据快速汇总并建立交叉列表的交互式报表。帮助用户分析、组织数据，并能进行二次过滤筛选等操作以达到用户想要的效果。

## 零客户端数据填报

多数据源填报、多种提交方式以帮助客户进行数据的录入功能，同时提供零编码且丰富的编辑风格，提供多种控件来协助用户实现多种快速录入的编辑方法。FineReport支持浏览器端的自动计算和数据校验，减少客户的人工计算校验成本。

## Flash打印

安装Flash Player后即可进行浏览器端的打印，不像Applet和Pdf打印需要安装较大的插件。

## 多种输出方式

Excel，Word，Pdf，Txt，Html等多种文件形式的导出。

## 参数获取与定时任务

对于更新频繁的报表可以通过定时任务管理生成，减少忙时查询数据对服务器的压力，并减少频繁人工更新报表的工作量。

## B/S资源管理方式

数据连接、服务器数据集、缓存、打印服务、邮件等功能均能通过在线服务管理平台灵活的设置，并提供了报表的多项日志信息以供用户查询。

## 数据实时更新

报表的数据会跟随数据库的更新而更新。

# 开发目录介绍

报表设置的一系列基础文件，如何操作可看文档报表管理

/bpm/src/main/java/com/hotent/platform/controller/system/PositionController.java

/bpm/src/main/java/com/hotent/platform/controller/system/ReportParamFormController.java

/bpm/src/main/java/com/hotent/platform/dao/system/ReportParamDao.java

/bpm/src/main/java/com/hotent/platform/model/system/ReportTemplate.java

/bpm/src/main/java/com/hotent/platform/service/system/ReportTemplateService.java

/bpm/src/main/webapp/WEB-INF/view/platform/system/reportTemplateManage.jsp

/bpm/src/main/webapp/WEB-INF/view/platform/system/reportTemplateList.jsp

/bpm/src/main/webapp/WEB-INF/view/platform/system/reportTemplateGet.jsp

/bpm/src/main/webapp/WEB-INF/view/platform/system/reportTemplateEdit.jsp

为了控制自定义报表控件而写的类。以后扩展

/bpm/src/main/java/com/hotent/core/report/ReportParamSearch.java

Finereport配置文件路径

/bpm/src/main/webapp/WEB-INF/resources

系统中报表文件上传后的存放路径

/bpm/src/main/webapp/WEB-INF/reportlets

# Finereport 集成

下载finereport，地址：http://www.finereport.com/

由于FineReport提供的报表服务器其实就是一个Servlet，所以在配置FineReport服务器之前，首先确保机器上已经安装了用来解析Servlet的Web服务器，这里以Tomcat服务器为例，介绍如何将报表工程部署到服务器上。

注意:FineReport报表服务器需要支持JDK1.4及以上版本的应用服务器版本。

## 独立部署

独立部署就是将FineReport应用集成到新的Web工程

把FineReport\_6.5安装目录下的WebReport文件拷贝到%Tomcat\_HOME%/webapps/下：



应用名WebReport可以根据需要改变，如可以改为“FineReport”。

## 嵌入式部署

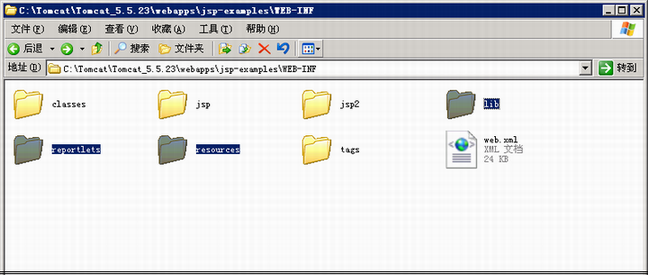
嵌入式部署将FineReport应用集成到已有的Web工程

在此使用%Tomcat\_HOME%\webapps\目录下的tomcat自带的jsp-examples工程：

### 复制目录

将%FineReport\_HOME%\WebReport\WEB-INF目录下面的classes,lib,reportlets,resources四个目录复制到%Tomcat\_HOME% \webapps\jsp-examples\WEB-INF下。

注意：%FineReport\_HOME%\WebReport\WEB-INF\classes文件夹下面可能会放置一些网络报表的class文件，在和您的工程集成时并无冲突。有关网络报表的介绍，可以在[二次开发](http://www.finereporthelp.com/api/" \t "_blank)文档中找到。Resources 中 fsconfig.xml 不用复制，会导致启动时有不友好提示。



### 整合web.xml文件

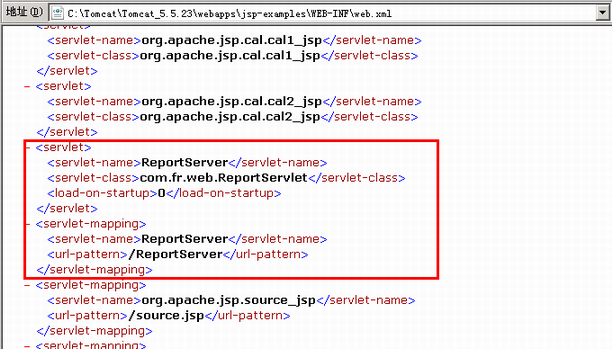
tomcat集成只需要在已有工程的web.xml中添加相应的servlet与servlet-mapping子元素。

将%FineReport\_HOME%/WebReport/WEB-INF下的web.xml中如下的部分复制到%tomcat\_home%/webapps/jsp-examples/WEB-INF下的web.xml中，在最后一个servlet之后插入：

1. <servlet>
2. <servlet-name>ReportServer</servlet-name>
3. <servlet-class>com.fr.web.ReportServlet</servlet-class>
4. <load-on-startup>0</load-on-startup>
5. </servlet>
6. <servlet-mapping>
7. <servlet-name>ReportServer</servlet-name>
8. <url-pattern>/ReportServer</url-pattern>
9. </servlet-mapping>



如上代码复制放到如下目录下面的配置文件里面结果如下：



注：Tomcat中web.xml中对于元素出现的顺序敏感，servlet必须出现在servlet-mapping标志之前，所以将所有servlet集合放置在servlet-mapping之前以便于管理与程序调用。

## 检测是否部署成功

可以通过浏览报表确定是否部署成功

重新启动Tomcat， 启动浏览器，在地址栏输入

http:/ip:服务器端口号/项目所在目录/ReportServer?op=fr\_platform，能成功进入管理平台，则表明FineReport应用部署Tomcat服务器成功：



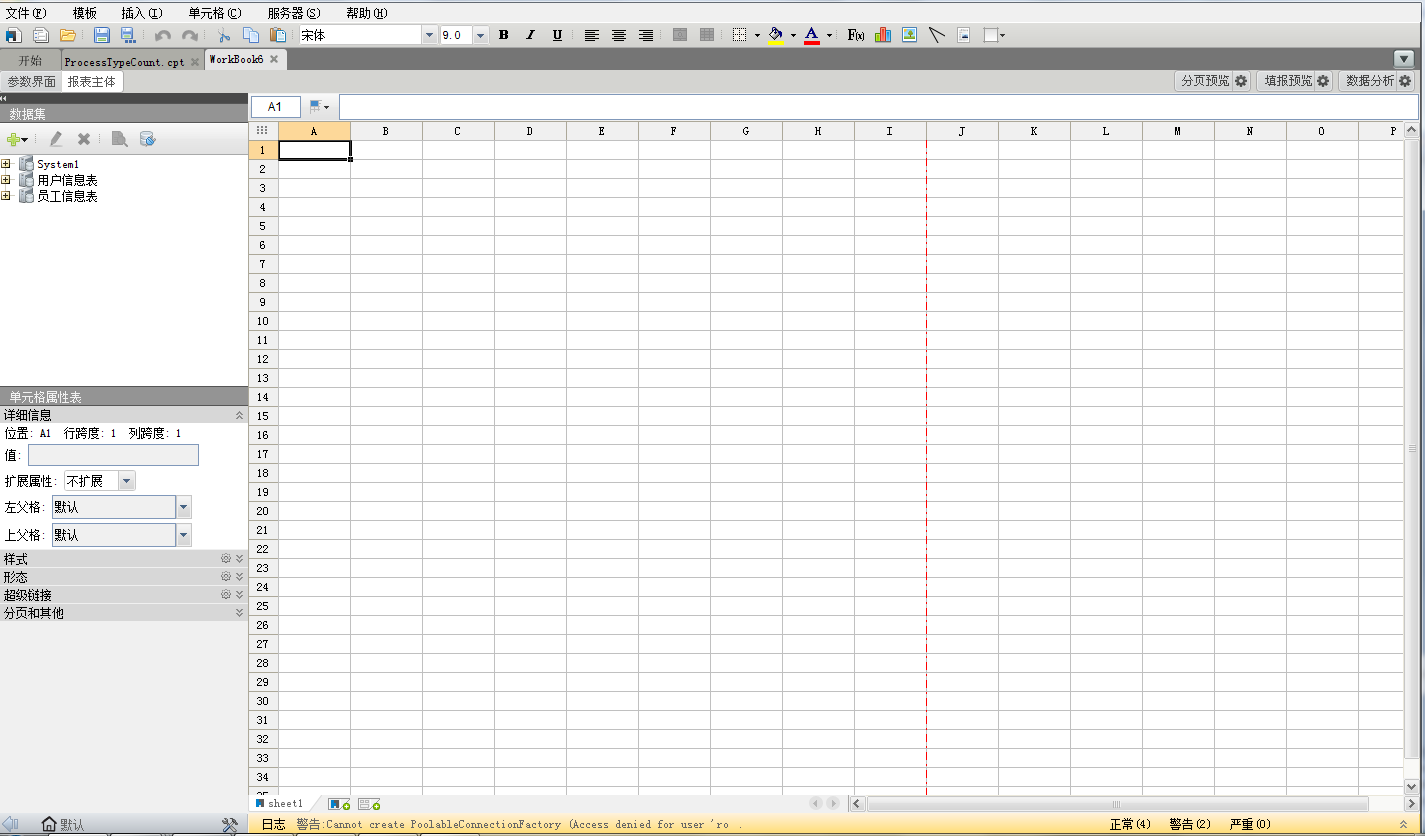
# 使用方法

## 普通报表

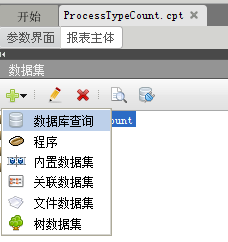
下面我会以bpm3系统中的常用流程分类作为例子详细说明

首先我们先打开finereport 的设计器，目录在FineReport\_7.0\bin

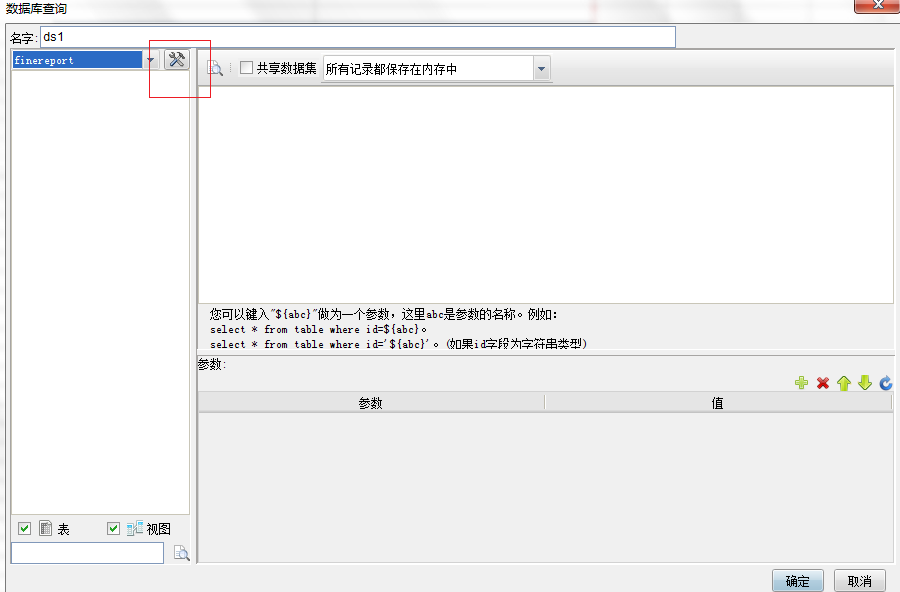
这是一个空白文档

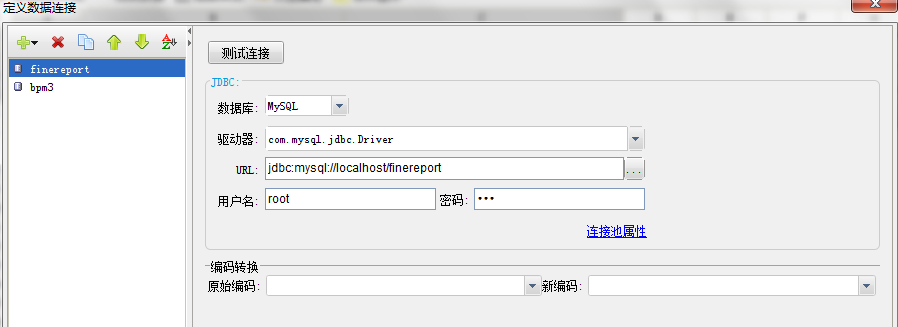


添加数据库查询

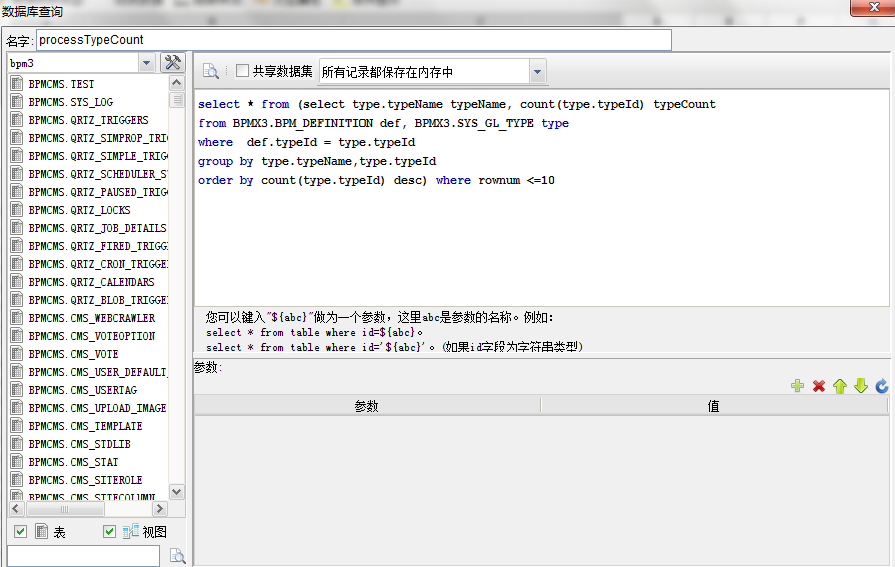


点击添加数据源

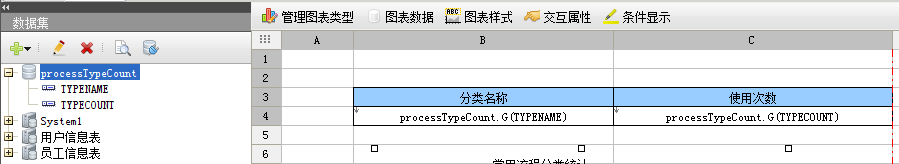




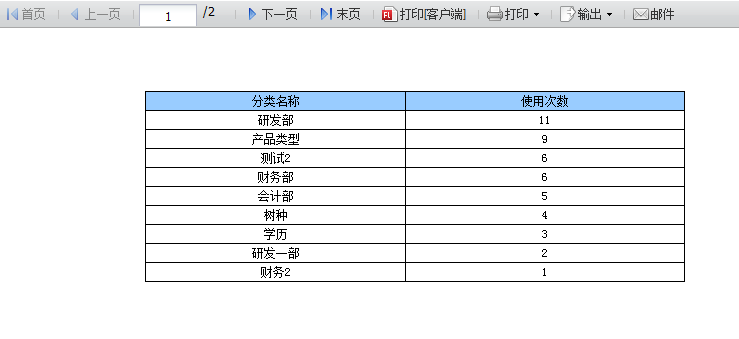
选择数据源并编写查询语句，也可以为参数设值



回到主界面左边树出现刚才添加的查询，展开把字段拖放到相应单元格，预览可以看到效果

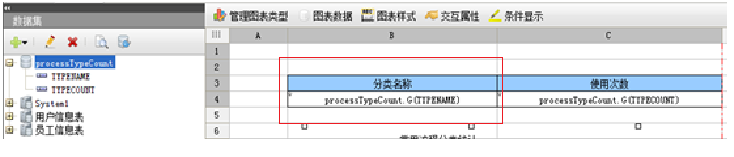


预览

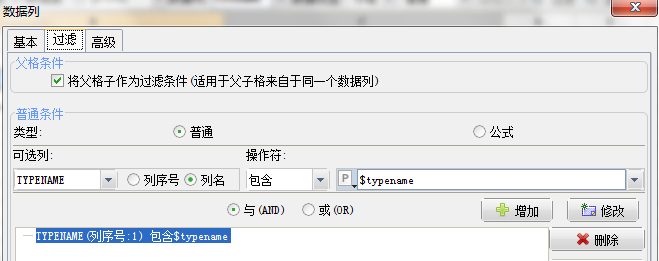


## 参数报表

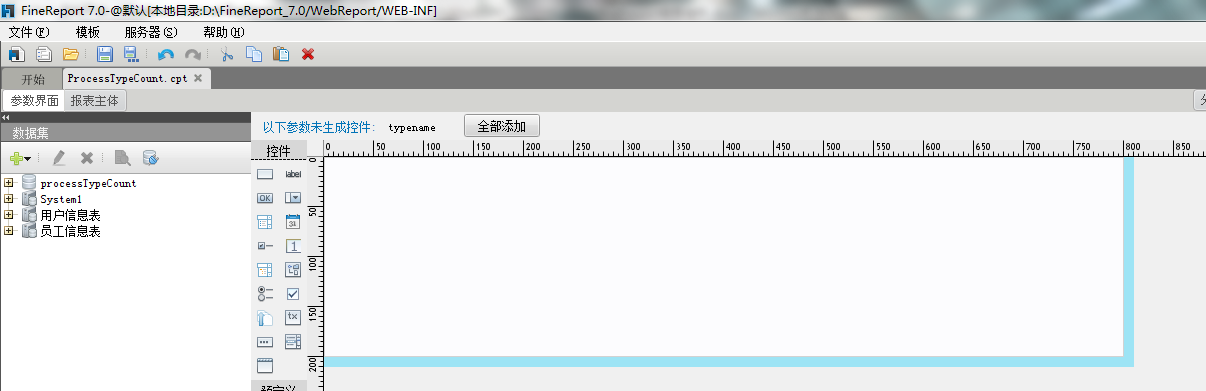
例如现在要根据类型名称查询



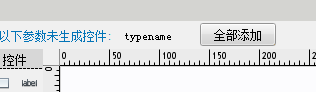
右键打开字段单元格--- 编辑，这是个模糊查询设定

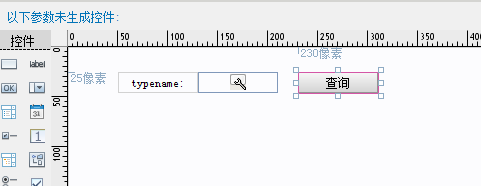


点击左上角界面参数

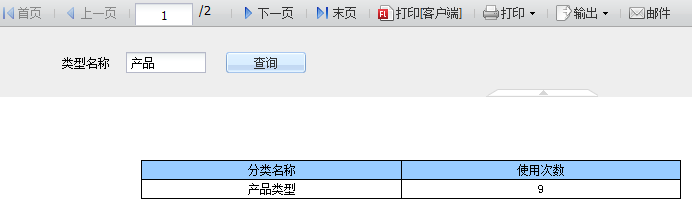


上方有刚才加的参数，点击生成控件



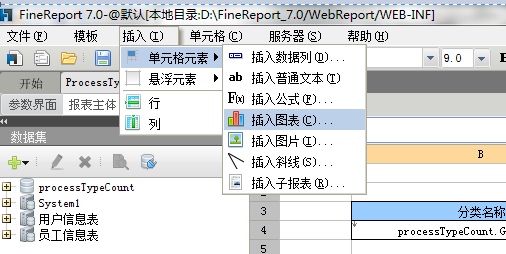


预览

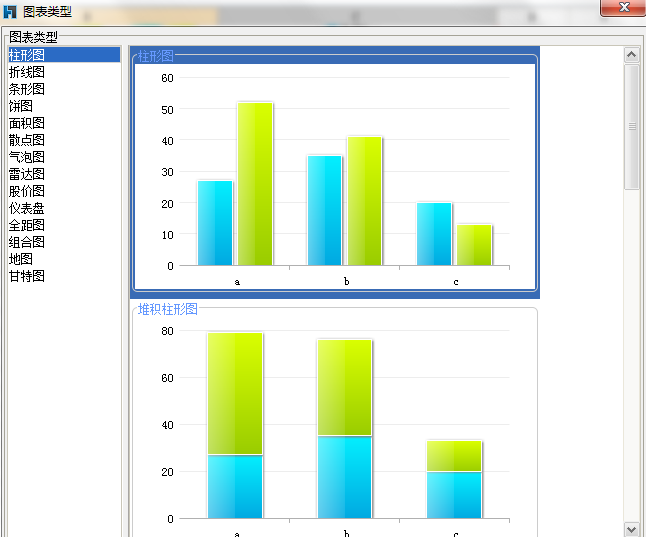


## 图表报表

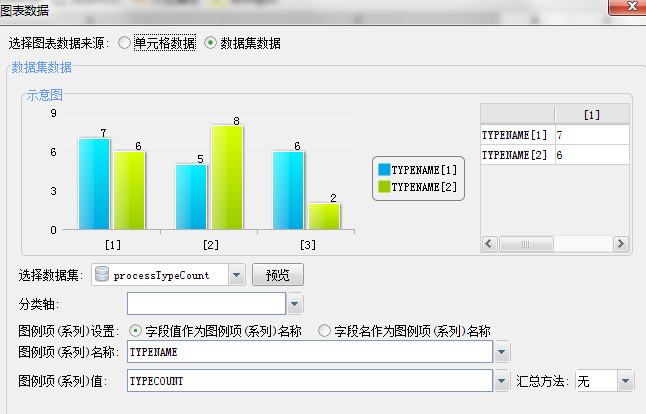
插入图表

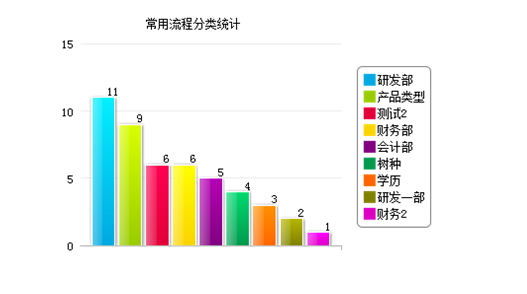


选取图表类型



以柱状图为例，设置柱状图，字段显示

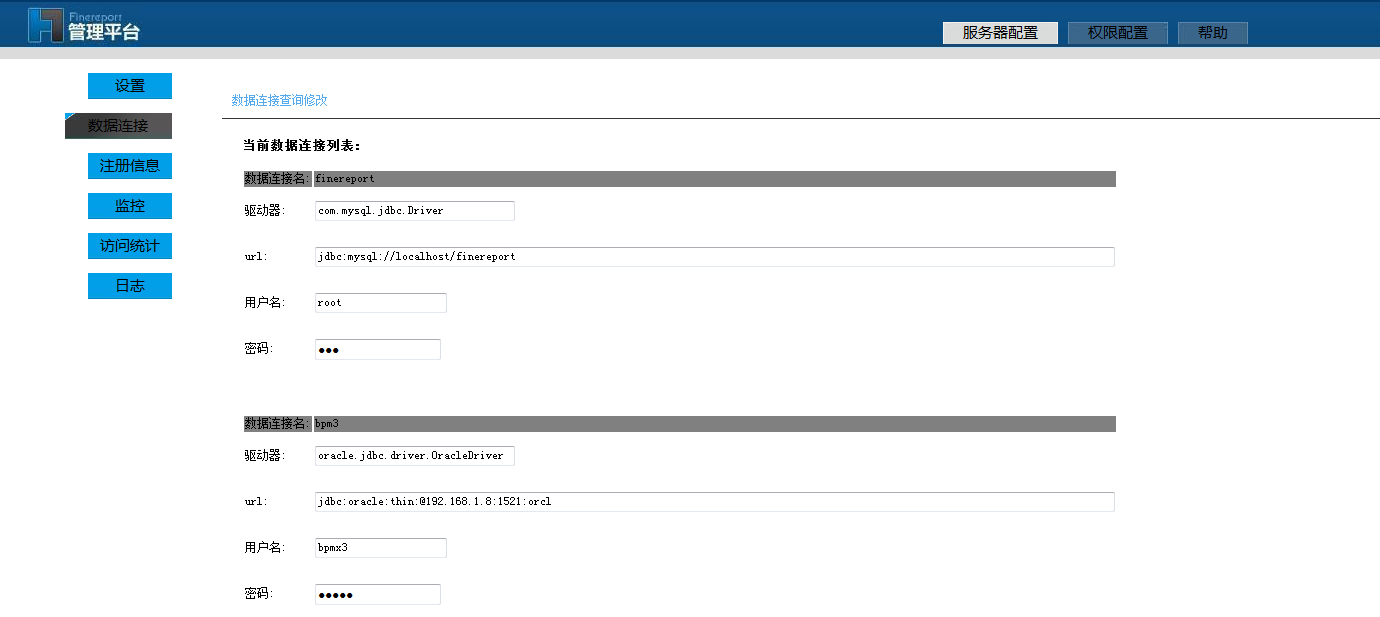




# 外部改变数据源

切换数据库如何改变数据源的配置，它提供了外部改变

[**http://localhost:8080/bpm/ReportServer?op=fr\_platform**](http://localhost:8080/bpm/ReportServer?op=fr_platform)



它的配置文件在

/bpm/src/main/webapp/WEB-INF/resources/datasource.xml

已将改变报表数据源，移植至 报表管理 -- 报表数据源

在资源管理中配置菜单 /platform/system/fineReportDs.ht。即可操作