**驰骋表单设计器**

**操**

**作**

**说**

**明**

**书**

**济南驰骋信息技术有限公司**

**2011年8月修订**

**目录**

[1总体说明 6](#_Toc321920884)

[1.1前言(必读) 6](#_Toc321920885)

[1.2一种存储格式(同一个表单) 6](#_Toc321920886)

[1.3两种表单设计器(同一个表单) 7](#_Toc321920887)

[1.3.1傻瓜表单设计器 7](#_Toc321920888)

[1.3.2自由表单设计器 7](#_Toc321920889)

[1.4 三种展现方式(同一个表单) 8](#_Toc321920890)

[1.4.1傻瓜表单 8](#_Toc321920891)

[1.4.2自由表单 9](#_Toc321920892)

[1.4.3手机表单 9](#_Toc321920893)

[1.5自由表单设计器预览 11](#_Toc321920894)

[1.6傻瓜表单设计器预览 11](#_Toc321920895)

[1.7表单样本 12](#_Toc321920896)

[企业基本情况 12](#_Toc321920897)

[财务负责人简历 13](#_Toc321920898)

[从表导入导出 13](#_Toc321920899)

[多表头从表 14](#_Toc321920900)

[从表的卡片表现形式 14](#_Toc321920901)

[1.8全局变量说明 15](#_Toc321920902)

[2调用API 15](#_Toc321920903)

[2.1与工作流程引擎API 15](#_Toc321920904)

[2.2与应用操作的API 16](#_Toc321920905)

[3自由表单的工具箱 16](#_Toc321920906)

[3.1装饰类的对象： 17](#_Toc321920907)

[3.1.1线 17](#_Toc321920908)

[3.1.2标签 17](#_Toc321920909)

[3.1.3装饰图片 17](#_Toc321920910)

[3.2数据类对象： 17](#_Toc321920911)

[文本框, 日期时间框, 选择框, bool类型控件. 18](#_Toc321920912)

[下拉框枚举类型 19](#_Toc321920913)

[下拉框(表/视图) 20](#_Toc321920914)

[单选按钮 20](#_Toc321920915)

[单个表单附件 21](#_Toc321920916)

[多表单附件 22](#_Toc321920917)

[图片附件 24](#_Toc321920918)

[从表(从表) 27](#_Toc321920919)

[一对多的关系表 29](#_Toc321920920)

[一对多多关系表 31](#_Toc321920921)

[隐藏字段 31](#_Toc321920922)

[3.3功能类 32](#_Toc321920923)

[3.3.1按钮 32](#_Toc321920924)

[3.3.2鼠标 36](#_Toc321920925)

[3.3.3选择 36](#_Toc321920926)

[3.3.4超连接 36](#_Toc321920927)

[4表单设计器功能 37](#_Toc321920928)

[4.1属性： 37](#_Toc321920929)

[4.2保存： 38](#_Toc321920930)

[4.3预览： 38](#_Toc321920931)

[4.4导出： 39](#_Toc321920932)

[4.5导入 39](#_Toc321920933)

[从ccflow网站上导入： 39](#_Toc321920934)

[从本机导入： 39](#_Toc321920935)

[从节点表单导入： 39](#_Toc321920936)

[从流程表单导入： 39](#_Toc321920937)

[4.6全选 40](#_Toc321920938)

[4.7复制 40](#_Toc321920939)

[4.8粘贴 40](#_Toc321920940)

[4.9对齐方式按钮 40](#_Toc321920941)

[4.10删除 40](#_Toc321920942)

[4.11撤消 40](#_Toc321920943)

[4.12恢复 41](#_Toc321920944)

[4.13加粗 41](#_Toc321920945)

[4.14斜体 41](#_Toc321920946)

[4.15删除线 41](#_Toc321920947)

[4.16增益 41](#_Toc321920948)

[4.17减益 41](#_Toc321920949)

[4.18调色版 41](#_Toc321920950)

[4.19表单事件 42](#_Toc321920951)

[事件类型: 42](#_Toc321920952)

[页面装载前 43](#_Toc321920953)

[页面装载后 43](#_Toc321920954)

[保存前 43](#_Toc321920955)

[保存后 43](#_Toc321920956)

[5字段扩展设置 43](#_Toc321920957)

[5.0 表单装载前数据填充 44](#_Toc321920958)

[5.1数据获取 45](#_Toc321920959)

[方式1：本表单中数据计算 46](#_Toc321920960)

[方式2：利用SQL自动填充. 46](#_Toc321920961)

[方式3：本表单中外键列。 47](#_Toc321920962)

[方式4：对一个从表的列求值。 47](#_Toc321920963)

[5.2级联下拉框 47](#_Toc321920964)

[主表中的级联 47](#_Toc321920965)

[从表中的级联 49](#_Toc321920966)

[常见问题 50](#_Toc321920967)

[5.3表单字段的输入js脚本验证 50](#_Toc321920968)

[设置步骤 50](#_Toc321920969)

[如何设置自定义验证函数? 51](#_Toc321920970)

[脚本验证的工作原理 51](#_Toc321920971)

[如何通过获取表单控件值？ 51](#_Toc321920972)

[如何在提交前做一个完整的表单输入验证? 52](#_Toc321920973)

[5.4自动完成 53](#_Toc321920974)

[5.4.1 演示环境说明 53](#_Toc321920975)

[5.4.2主表中的自动完成 53](#_Toc321920976)

[5.4.2主从表的自动完成 54](#_Toc321920977)

[5.4.3从表中的自动完成 55](#_Toc321920978)

[5.5 字段超连接 56](#_Toc321920979)

[在主表表单设计上的表现 57](#_Toc321920980)

[在从表的表现与设计 58](#_Toc321920981)

[5.6 Pop返回值 58](#_Toc321920982)

[事例： 58](#_Toc321920983)

[设置方式 59](#_Toc321920984)

[6从表的设计 60](#_Toc321920985)

[6.1从表的设计器 60](#_Toc321920986)

[6.2从表的属性 61](#_Toc321920987)

[6.3从表属性说明 61](#_Toc321920988)

[操作权限控制 62](#_Toc321920989)

[是否起用审核字段： 62](#_Toc321920990)

[是否启用多附件 62](#_Toc321920991)

[显示格式 64](#_Toc321920992)

[越位处理 65](#_Toc321920993)

[6.4 从表事件 66](#_Toc321920994)

[6.5为从表设计导入数据模版 66](#_Toc321920995)

[7一对多的关系设计 68](#_Toc321920996)

[7.1什么是一对多的数据关系? 68](#_Toc321920997)

[7.2一对多的关系设计 69](#_Toc321920998)

[7.3数据存储格式是什么 70](#_Toc321920999)

[8 一对多多的关系设计 71](#_Toc321921000)

[8.1应用背景 71](#_Toc321921001)

[8.2设计方式 71](#_Toc321921002)

[8.3数据存储格式 71](#_Toc321921003)

[9表单的单据打印 71](#_Toc321921004)

[9.1单据打印总体概述 71](#_Toc321921005)

[9.2单据模板的制作 71](#_Toc321921006)

[9.3单据输出格式约定 72](#_Toc321921007)

[9.4单据生成的工作原理 74](#_Toc321921008)

[9.5单据表单的分类 74](#_Toc321921009)

[9.5.1节点表单 74](#_Toc321921010)

[9.5.2 流程表单 75](#_Toc321921011)

[9.6表单设计常见的问题 75](#_Toc321921012)

[如何解决乱码的问题？ 75](#_Toc321921013)

[10表单事件 76](#_Toc321921014)

[10.1事件概述 76](#_Toc321921015)

[10.2装载前事件 77](#_Toc321921016)

[10.3装载后事件 78](#_Toc321921017)

[10.4保存前事件 78](#_Toc321921018)

[10.5保存后事件 78](#_Toc321921019)

[11技术文摘 78](#_Toc321921020)

[11.1常见问题类 78](#_Toc321921021)

[11.1.1在表单设计器中，如何获取自由表单下拉框的值？ 78](#_Toc321921022)

[11.1.2如何用自由表单设计器映射我的物理表? 80](#_Toc321921023)

# 1总体说明

## 1.1前言(必读)

驰骋自由表单设计器有一种存储格式、两种设计模式，三种展现方式，具有一定的通用的接口可以被外部调用，自己有可以独立运行起来的应用程序。

**一种存储格式：**采用映射的机制，以xml文件格式存储。

**两种设计模式：**傻瓜表单设计器，自由表单设计器。

**三种展现方式：**傻瓜表单、自由表单、手机表单。

表单设计器以映射思想为核心。所说的映射业务规则对字段，操作方法，界面展现，操作行为的描述。

表单设计者认为事物的业务规则与操作模式都是可以被抽象出来的，并且把其封装并实现。 但是并不是所有的业务规则都有价值去抽象，是否有价值去抽象并封装是应用的范围与实现的代价。

由以上两个理论为指导，把界面所有的元素的描述，业务规则的描述以对象的形式为描述。

Ccflow 自由表单设计器不一定满足所有的数据采集与数据展现格式，但是可以解决对数据库操作层面的绝大多数需求。

用户可根据自己的需要选择其中的一种表单的工作方式。

## 1.2一种存储格式(同一个表单)

Ccflow的表单存储格式与传统的表单存储格式不同，它是采用数据表的存储的，这是最大区别于文件格式的存储。

正是因为ccflow摆脱了对文件格式存储的表单的依赖，所以具有最大范围的扩展性通用性、灵活性。

|  |  |  |
| --- | --- | --- |
| **物理表名** | **中文名** | **说明** |
| Sys\_FrmLab | 标签 | 存储标签 |
| Sys\_FrmLine | 线 | 存储线，线只分横线与竖线。 |
| Sys\_FrmLink | 超连接 | 超连接 |
| Sys\_FrmImg | 图片 | 图片 |
| Sys\_FrmBtn | 按钮 | 按钮，按钮支持多样的事件方式 |
| Sys\_MapAttr | 字段 | 存放三种类型的字段  textbox,checkbox,dropdownlistbox. |
| Sys\_MapData | 表单主表 |  |
| Sys\_M2M | 一对多 |  |
| Sys\_M2MM | 一对多多 |  |
| Sys\_FrmAttachment | 附件 |  |
| …. | … | … 更多请参考程序代码. |

可以利用ccform 的导出功能把这些规则化的数据导出成一个xml。

## 1.3两种表单设计器(同一个表单)

驰骋公司根据用户群体不同，用户对界面的要求不同特开发了两种表单设计器，三种展示模式。

自由表单设计器与傻瓜表单设计器，两者的存储格式都是一样的，就是展示与设计方式不同。

### 1.3.1傻瓜表单设计器

**优点：**设计方便，不需要排版，有分组显示功能界面简洁清晰，容易上手。

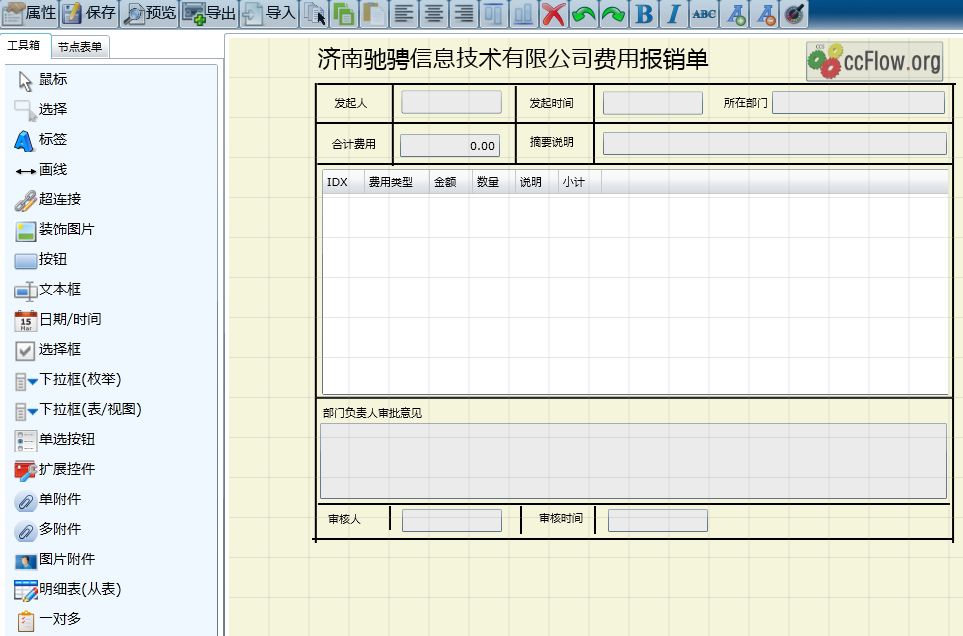
**缺点：**不形象直观，不符合用户的需求格式。



### 1.3.2自由表单设计器

**优点：**设计方便，形象直观，随心所欲。

**缺点：** 没有傻瓜表单简洁，需要排版，同样的功能没有傻瓜格式方便。



## 1.4 三种展现方式(同一个表单)

对于一个表单来说有三种展现方式，至于用户使用那种表单是有流程的节点属性来设置的。

### 1.4.1傻瓜表单

**优点：**界面清晰、简洁、紧凑。

**缺点：**格式固定。



### 1.4.2自由表单

**优点：**界面元素可以根据用户的需要任意排列。

**缺点：**无



### 1.4.3手机表单

**手机填写主表:**



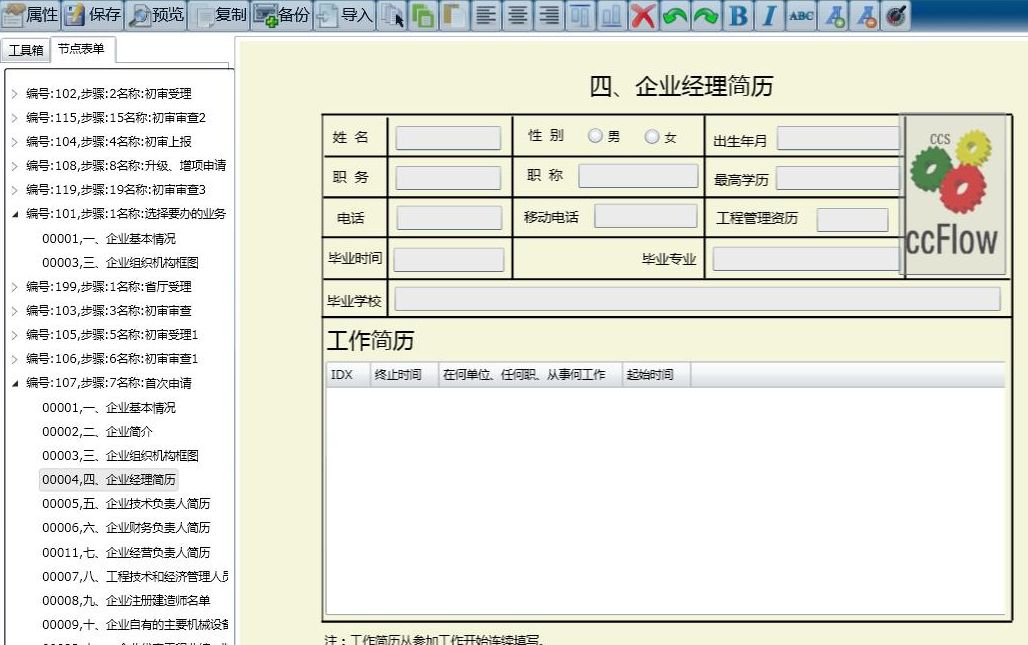
**手机明细表: 点上图中的费用明细就可以看到下面的明细表查看界面。**



手机表单是

手机表单特点

## 1.5自由表单设计器预览



## 1.6傻瓜表单设计器预览



## 1.7表单样本

### 企业基本情况

企业基本情况(打印)与打印模板.



### 财务负责人简历

具有附件图片



### 从表导入导出

从表的导入导出



### 多表头从表

企业代表工程业绩一览表(多级表头明细)



### 从表的卡片表现形式

从表的卡片方式展现



## 1.8表达式全局变量

在系统中有很多的地方需要用到表达式的地方，这些表达式就需要参数与变量，这些参数是以@+变量名。

我们把参数与变量分为全局变量，字段变量。

**操作人员登陆系统后就有如下全局变量。**

@WebUser.No 当前操作员编号

@WebUser.Name 当前操作员名称

@WebUser.FK\_Dept 当前操作员部门

@AppPath 当前的文件路径。比如:http://demo.ccflow.org/ccflow

变量用@加变量名体现。比如:@WebUser.No。以FF中，很多的表达式需要变量的支持。

比如一个sql表达公式。SELECT No,Name, FK\_Dept FROM PORT\_Emp WHERE FK\_DEPT=’@WebUser.FK\_Dept’

注意区分大小写。

表单字段变量：@+字段名 比如:

Select generBillNo(@OID) from dual OID 就是表单字段。

# 2调用API

## 2.1与工作流程引擎API

如何把Ccform 嵌入到工作流程引擎中去？

**(省略)**

## 2.2与应用操作的API

如何使用Ccform 在我的应用程序中?

调用连接 [http://localhost/XXXX/Ccform/Frm.aspx?FK\_MapData=YYYYYYY](http://localhost/XXXX/FreeFrm/Frm.aspx?FK_MapData=YYYYYYY)

对于表单的权限控制可见不可见来源来应用程序本身，对于表单内的按钮控制由Ccform 的按钮属性来确定的。

# 3自由表单的工具箱

工具栏中的对象分为装饰类的对象、数据类的对象、与功能类的对象。

|  |  |
| --- | --- |
|  | **鼠标：**类型vs设计器中的鼠标功能，当您点了其它的功能按钮后，您想取消选择，就点此功能。  **选择：**在画布画矩形在矩形内部的对象都被选择，比如对选择的线条的粗细进行+ - 。对颜色进行更换，对label 集合的字体颜色进行变化。  **标签：**文字类装饰元素，标签的文字大小、类型、颜色，都可以变化。  **画线：**线条的粗细、长度、颜色属性可以改变。  **超连接：**功能类的控件，它有两部分组成标签与连接，标签部分的功能与标签功能一致。  **装饰图片：**装饰类的控件，多用于把公司的icon图片放在上面。  **按钮：**为了实现一些功能  **文本框：**长度、高度、默认值都可以被编辑。  **日期框：**长度固定  **选择框：**Boolean类型的字段。  **下拉框（枚举）：**枚举类型下拉框。  **下拉框（表、视图）：**下拉框中是以表格为内容的。  **表单附件：**可以是任何格式的控件。  **图片附件：**比如简历的个人图片。  **从表：**与主表之单的表格  **一对多的关系：**比如与主表的关系，节点岗位，  **隐藏字段：**不需要在表单上显示出来的字段，多用于值的计算，或者节点方向条件的转换。 |

## 3.1装饰类的对象：

线、标签、装饰图片

### 3.1.1线

线是装饰类对象的基本元素。

对于线的操作

### 3.1.2标签

标签是用来文字说明装饰，标签可以改变颜色，字体大小，风格等等。

标签元素可以复制，选中后，ctrl+C Ctrl+V 就可以实现。

支持鼠标移动， aswd 键，方向键移动。

### 3.1.3装饰图片

图片是表单元素的一部分。

## 3.2数据类对象

文本框、日期框、选择框、下拉框、单选按钮、表单附件、图片附件、从表、一对多的关系表、隐藏字段。

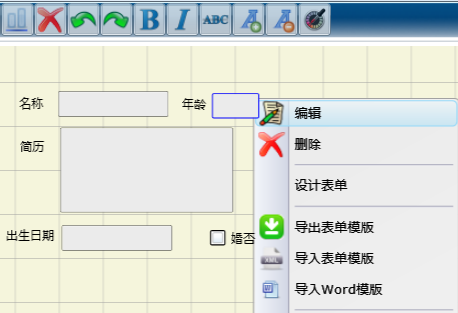
### 3.2.1文本框, 日期时间框, 选择框, bool类型控件.



**说明：**

1, 日期与日期时间框不能改变大小，其它的类型的文本框可以改变大小。

2, 要改变文本框的位置可以按下它的边框，然后移动，或者选中它，使用方向键移动。

3, 改变文本框的宽度可以使用 shift + 方向键。 改变文本框的高度与宽度。  


属性编辑:

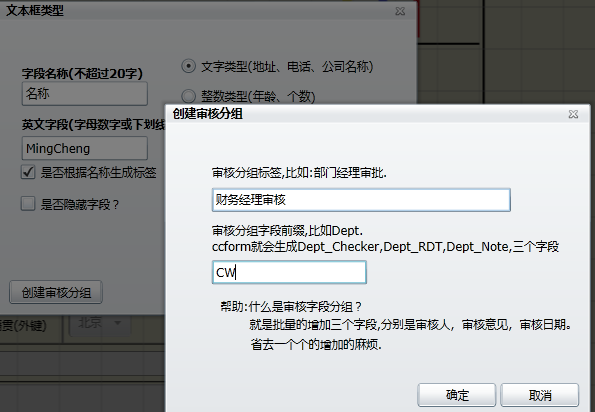


如上图

#### 3.2.1.1使用自由表单设计器创建审核分组

在表单设计中经常遇到领导审核、或者审批，这种审批与审核通常有三个字段组成分别是审核意见、审核人、审核时间。

审核意见是一个大块文本采集框、审核人是默认当前操作人员、审核时间就是当前时间。如果在自由表单设计器中一个个的字段添加也是可以的，ccform为了提高设计者的效率与傻瓜表单设计器一样推出增加审核分组功能。如下图：



填写说明见上图:

如果审核分组的字段前缀是CW，那系统就会创建三个字段分别是CW\_Checker审核人 , CW\_RDT审核日期,CW\_Note审核意见。同时也建立了一个字段分组，条理化规则的显示在傻瓜表单设计器中。

#### 3.2.1.2 创建隐藏字段

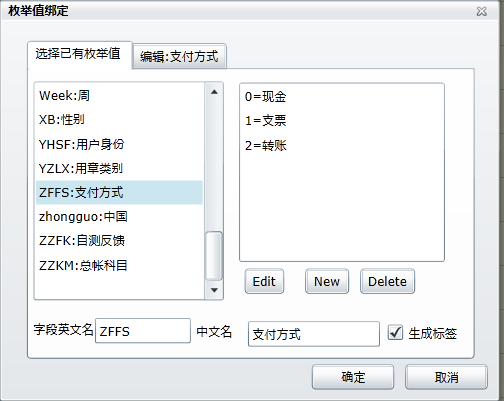
**隐藏字段的用途？**

隐藏字段不体现在表单里，隐藏字段一般用于计算临时计算，在节点表单中做为方向条件。

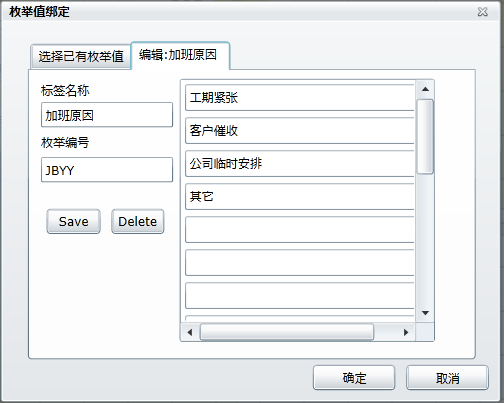
**如何创建与编辑隐藏字段？**

在创建正常字段时您是是否是隐藏字段勾上，如果要编辑隐藏字段在工具栏里有一个隐藏字段图标。

### 3.2.2下拉框枚举类型



新建枚举类型



### 3.2.3物理表或视图

**关键字：ccform 外键表 外键字段**

对于一个表单来说外键是经常遇到的，ccform为我们提供系统的基础表(Port\_Emp,Port\_Dept, CN\_PQ ,CN\_SF,CN\_City,)，当然也可以把自己定义外键表，或者说字典表。

在您初始化系统后，您可以打开SELECT \*FROM　Sys\_SFTable你将会看到一些演示的数据。’

外键实体分为物理表类型与系统类库. 用类库可以很好的控制查询权限,比如人员查询权限，在表单中只列出该操作员的部门人员。部门类可以控制只查询本人的部门与本人的下级部门。如果用物理表就没有这个概念了。

外键表存储在Sys\_SFTable 系统表中，您可以通过界面维护它，也可以手工的维护它。如果您手工的维护(注意：SFTableType此字段已经不用了)。

**对于Sys\_SFTable 列描述如下：**

No: 外键,

Name: 表名称,

FK\_Val 在创建外键字段时默认的外键字段名称

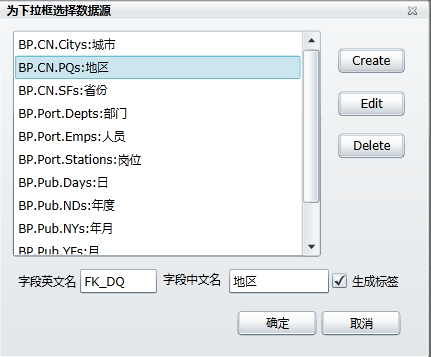
TableDesc: 描述。

IsEdit 是否可以编辑。

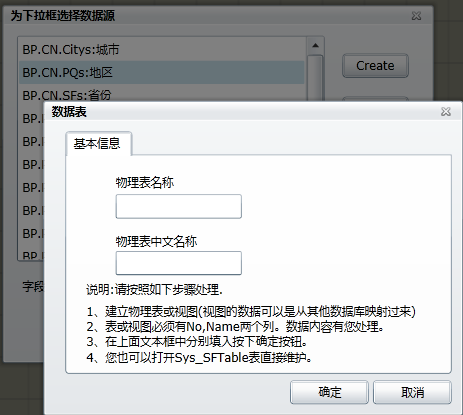
**如何把ERP，OA，CRM应用程序中的物理表对应到ccform中去？**

首先把应用系统的物理表通过视图映射到ccflow 可访问的数据库中来，这个表或者视图必须有No，Name,两个列。其次把这个数据维护到Sys\_SFTable中去。

#### 3.2.3.1 自定义表列表



#### 3.2.3.2 创建与编辑



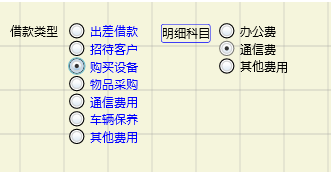
### 3.2.4单选按钮

单选按钮是枚举值的一种表现形式。

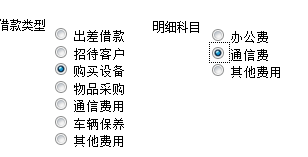
选择枚举



**设计样式**



**浏览样式**

****

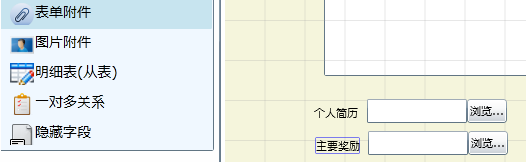
### 3.2.5单个表单附件

文件是以上传一个文件保存到服务器上，如下图所：



附件的格式需要以, 分开比如:\*.doc,\*.docx,\*.txt

**拖放后：**



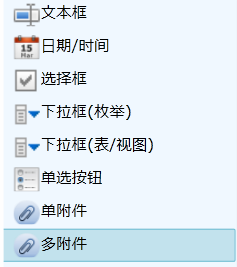
**预览表现：**



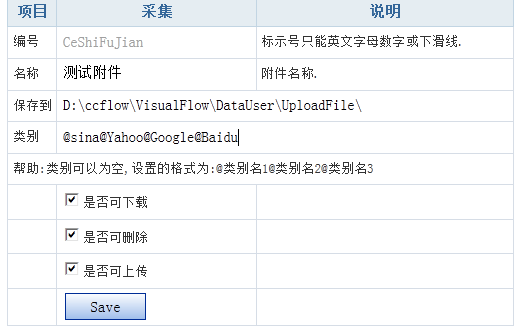
### 3.2.6多表单附件

多表单附件是用户上传的附件格式不能确定，数量不能确定情况下使用。

设计步骤：在工具栏点多附件按钮，如下图：



**多表单附件属性：**



附件名称：就是为控件起一个名称，附件编号需要全局唯一。保存到是文件要保存的位置。

是否可下载，是否可删除，是否可上传是用来控制附件的访问权限的。

**特别说明：**

类别可以为空，如果不为空时系统就会自动为它附件按约定的格式分类。

**两种附件的展现模式：**



### 3.2.7手写板



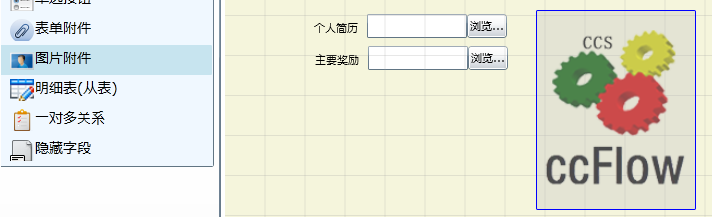
### 3.2.8图片附件

图片附件与文件附件不同的地方就是图片附件需要预览。

比如：我们做一个简历表单，需要把人员的照片放在指定的位置。

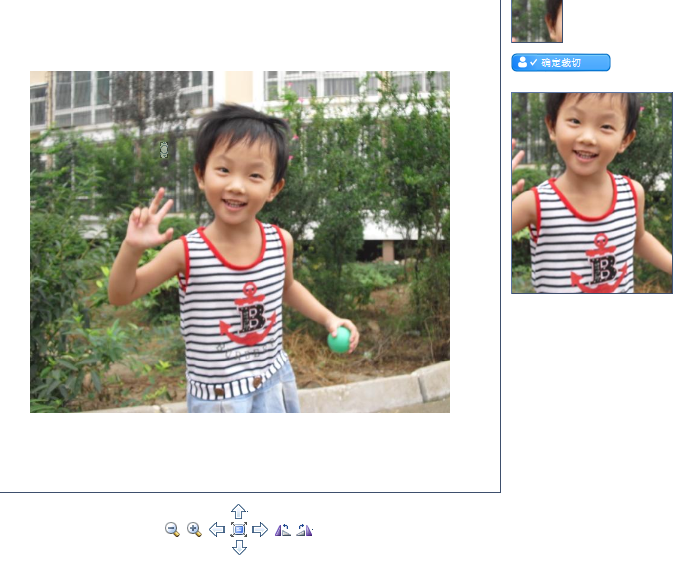


**设计方式：**

1. 在工具栏中找到图片附件。  
   
2. 用shift+ 方向键改变图片的大小。  
   预览效果如下图。  
   

编辑与上传图片：

点编辑连接，系统弹出如下对话框。



可以把剪切的图片点确定按钮，返回。



### 3.2.9从表(明细表)

从表也叫明细表。

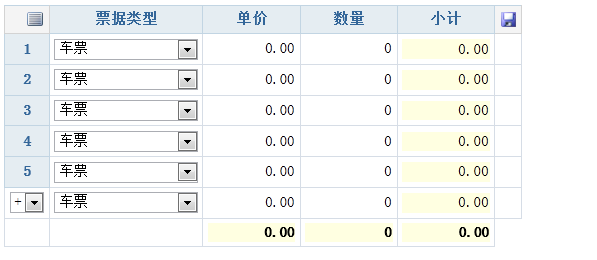
#### 设计界面



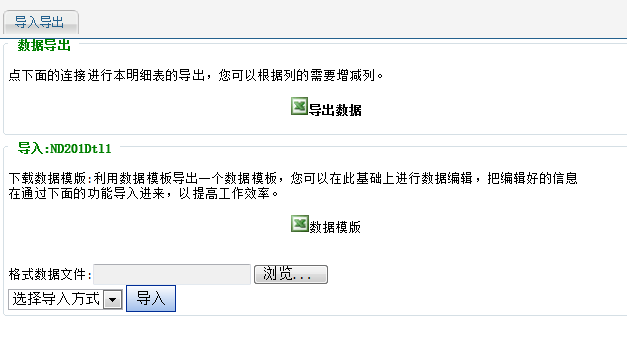
#### 从表属性界面



#### 预览与操作界面



点选项按钮



### 3.2.10一对多的关系表

一对多的关系是对多项选择来确定的。

比如：单位人员旅游名单。这些名单可以让用户去选择它。

#### 运行界面



#### 属性编辑



显示在分组，对所有的自由表单设计器来说无意义。

### 一对多多关系表

一对多有三个维度

第一维度是 对象列表.

他有两种表达方式。

1， sql 必须有两个列 No,Name.

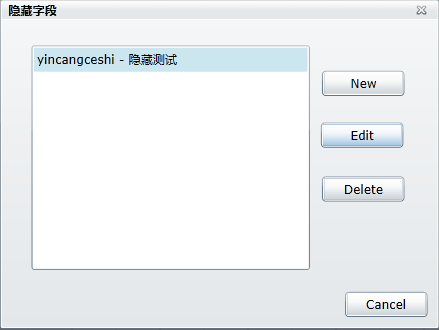
2, @+本表单中的多对多编号.

第二维度是: 如同 m2m 中的第一维度. where 表达公式中需要一个 @Key 就是来自于第一维度的 实体 PK.

第三维度是: 如同 m2m 中的第二纬度.

### 隐藏字段

用于存储临时计算值，在节点表单中，经常被用到方向条件。在您创建文本框时您如果选择了隐藏字段，系统就会把此字段不体现在表单设计器中。



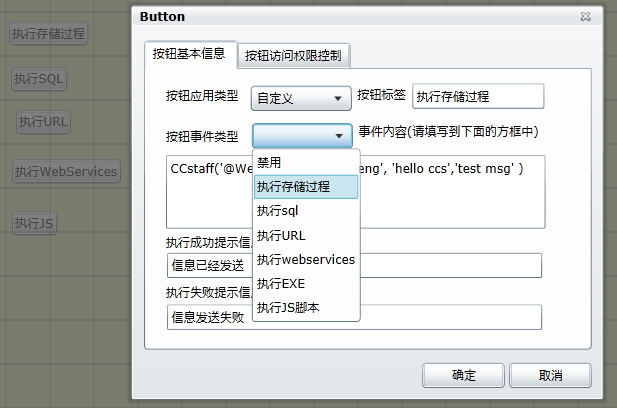
在创建正常字段时，您只要把是否是隐藏字段勾上就可以了。

## 3.3功能类

按钮，鼠标、选择，超连接。

### 3.3.1按钮

表单上面的按钮对象，可以处理事件内容。



特别说明：ccflow所有的事件内容的处理方式都是一致的，目前所涉及的事件有，表单事件、按钮事件、流程事件。

#### 总体说明

每个按钮有自己的按钮类型，使用自定义按钮类型。

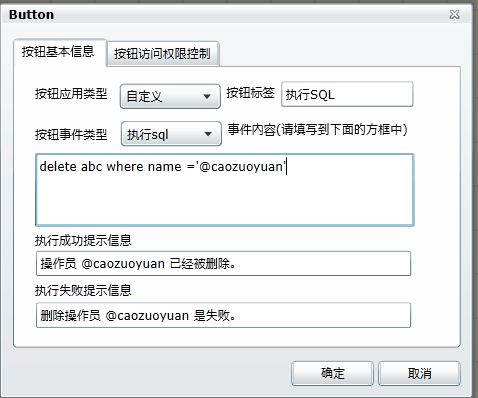
|  |  |
| --- | --- |
| **属性** | **说明** |
| 按钮事件类型 | 事件类型决定按钮执行的内容。 |
| 事件内容 | 就是要执行的内容，内容里面可以有变量，变量来自与表单。变量的表达方式仍然用@+变量名来决定，比如：@jiner , @shenpiren,  系统变量有@WebUser.No 当前操作员编号，@WebUser.Name 当前操作员名称 @WebUser.FK\_Dept 当前操作员部门。 |
| 执行成功提示信息 | 当内容执行成功时提示的信心。 |
| 执行失败提示信息 | 当内容执行失败时，抛出异常信息时提示的信息。 |

#### 执行存储过程案例



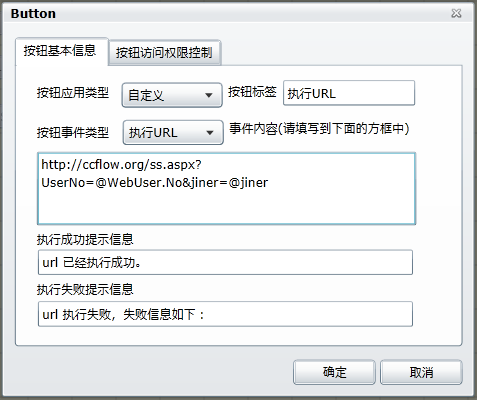
说明：@WebUser.No 表示当前的操作人员系统变量。

#### 执行SQL 事例



以上事例说明了删除一条数据时提示的信息。@+字段名是用来取当前节点字段。

#### 执行URL



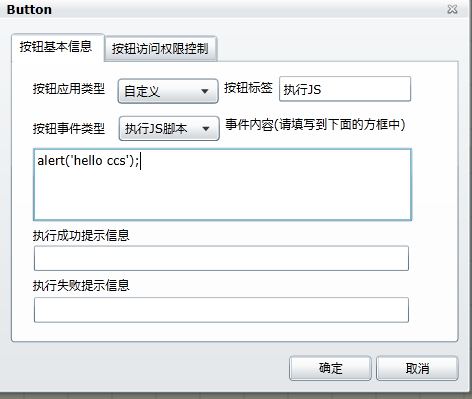
**特别说明:**

执行url 是静默执行，需要对方的页面直接返回 string 串就可以了。如果超时连接，或者，返回串的前三个字母是Err ，ccflow就会执行失败，就抛出失败信息给操作者。

在执行url时系统就会把两个默认的参数增加上。分别是UserNo, SID。

比如：您设定的是 http://serverName/a.aspx?type=@MyType 实际执行的是：http://serverName/a.aspx?type=@MyType&UserNo=abc &SID=a2345bc

#### 执行JavaScript



执行js 时，成功信息与失败提示信息就不需要填写了。

### 3.3.2鼠标

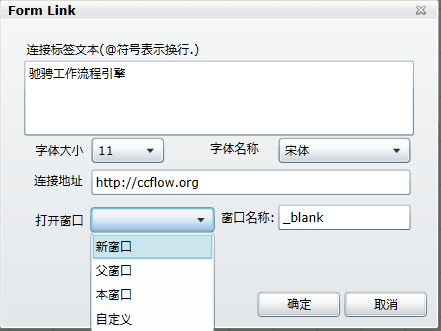
点它就是取消所有的工具选择。

### 3.3.3选择

全选特定的区域，比如要删除特定区域的对象，就可以使用它来选择。

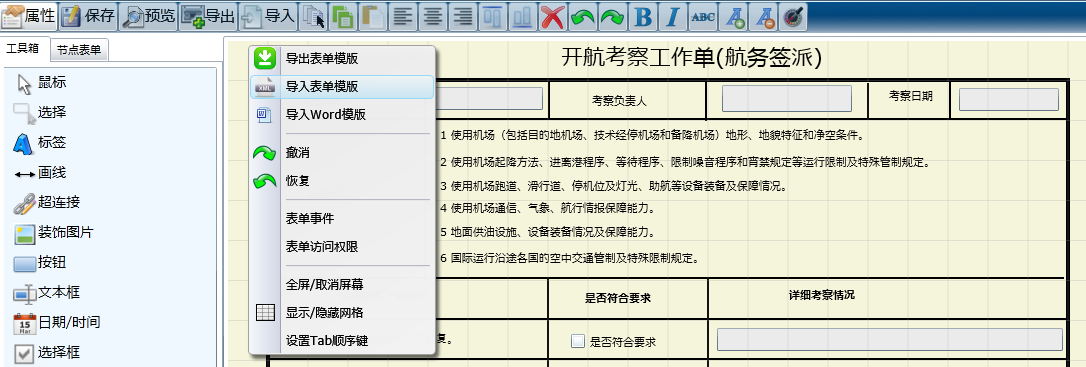
### 3.3.4超连接

显示超级连接.



提示:@符号表示换行。

# 4表单设计器功能



## 4.1属性：

用来控制表单的基本属性

****

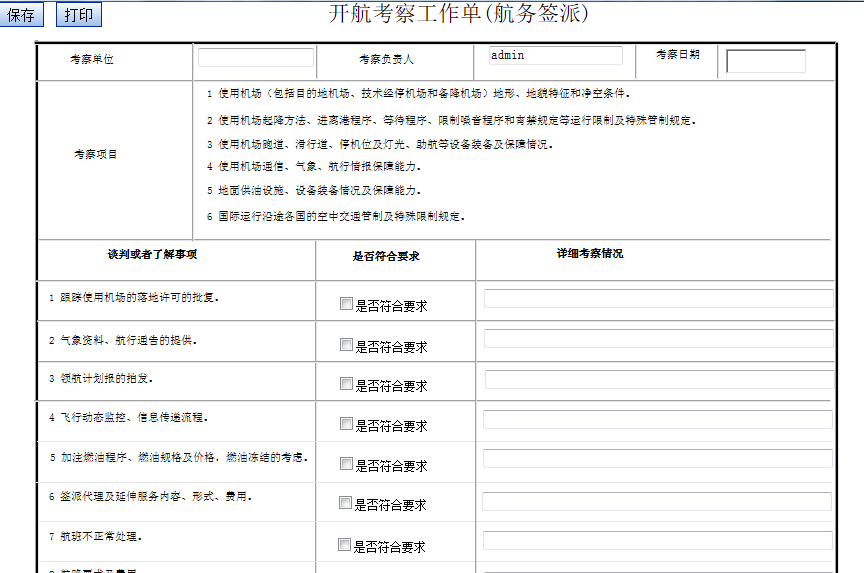
如果画布不够宽度与高度可以通过设置此属性来满足。

## 4.2保存：

保存当前的设计，在设计过程中多执行保存并预览按钮。目前的保存具有保存与预览的功能。

## 4.3预览：

浏览当前的设计。



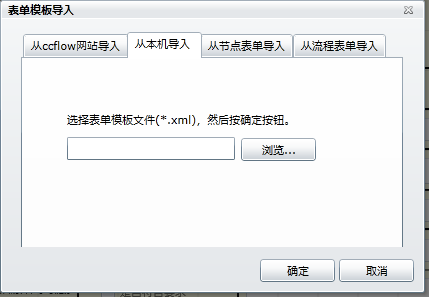
## 4.4导出：

导出ccform模板。

导出ccform模板并共享。

## 4.5导入

把当前的设计导出xml模板。



### 从ccflow网站上导入：

在ccflow官方网站上有一些网友共享的表单模板，您可能 通过他们方便的导入进来。

### 从本机导入：

导入存储在本机上的表单模板。

### 从节点表单导入：

从其它节点上导入表单进行二次编辑。

### 从流程表单导入：

从流程表单上导入表单进行二次编辑。

## 4.6全选

选择全部的元素，一般用于全部删除，全部平移对象，目前不太成熟。

## 4.7复制

复制选择的元素，到内存里。

## 4.8粘贴

只有如下对象支持粘贴：线，标签，文本框。

## 4.9对齐方式按钮

**左对齐**

在选择多个同类的控件时，比如：标签、线、文本框，执行此按钮会执行左对齐。

**居中**

在选择多个同类的控件时，比如：标签、线、文本框，执行此按钮会执行居中。

**右对齐**

在选择多个同类的控件时，比如：标签、线、文本框，执行此按钮会执行左对齐。

**顶部对齐**

在选择多个同类的控件时，比如：标签、线、文本框，执行此按钮会执行左对齐。

**底部对齐**

在选择多个同类的控件时，比如：标签、线、文本框，执行此按钮会执行底部对齐。

## 4.10删除

删除选择元素，如果选择的元素有多条，就要删除选择的元素集合。

## 4.11撤消

撤消上一步的操作。

## 4.12恢复

恢复上次的操作。

## 4.13加粗

对标签字段加粗，可对标签控件批量执行。

## 4.14斜体

对字体设置成斜体，可对标签控件批量执行。

## 4.15删除线

对字体设置成删除线，可对标签控件批量执行。

## 4.16增益

如果选择的是线条：点一次就增加一个像素的宽度。

如果选择的是标签：点一次就增加一个字体的大小。

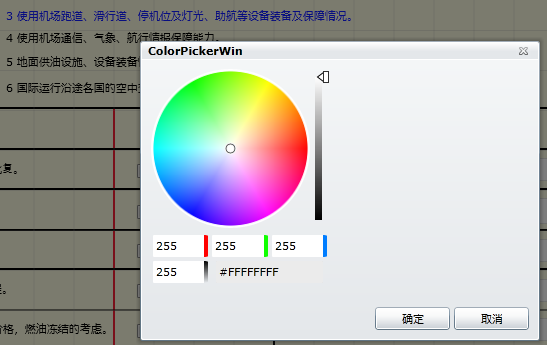
## 4.17减益

如果选择的是线条：点一次就减小一个像素的宽度。

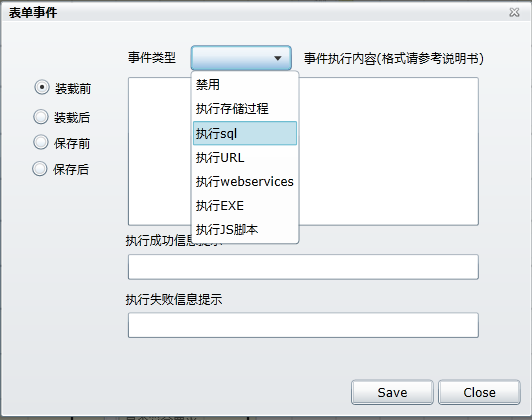
如果选择的是标签：点一次就减小一个字体的大小。

## 4.18调色版

用来改变线条，标签的颜色。



## 4.19表单事件



每个事件的填写内容都是一致的：

### 事件类型:

事件类型就是执行的方式，您可以根据自己的需要来选择执行方式。

**目前支持的事件内容：**

执行存储过程，执行sql, 执行url, 执行JS脚本。

在执行内容中，执行成功信息提示，执行失败信息提示都支持@字段名类型的参数。

**执行内容约定：**

请参考本章节:Button 事件内容约定。

**对异常的处理：**

当对执行的内容失败时，系统就会把执行失败的信息提示出来。

比如：在执行存储过程时，如果执行期间出现异常，或者在存储过程抛出异常，系统就会把异常提示信息与抛出异常信息提示给用户。

### 页面装载前

可以填写一个为表单初试化的一个脚本，比如预先把一些值放入文本主从表里。

### 页面装载后

同上

### 保存前

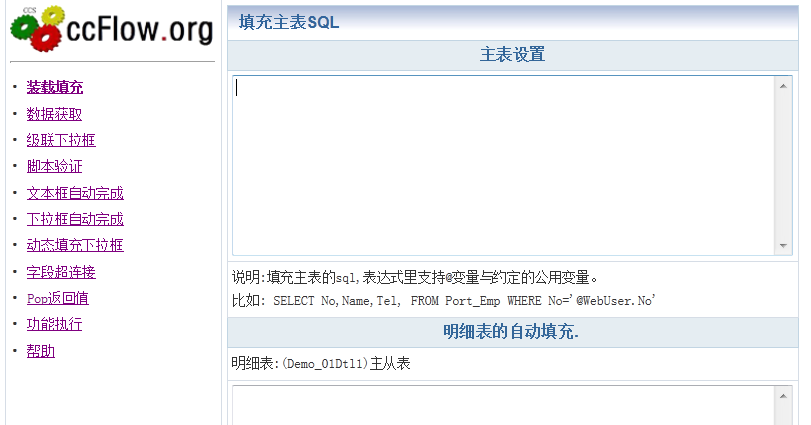
做一些业务逻辑的检查，不符合条件不让其保存。

### 保存后

执行相关的业务操作，比如把数据copy到其它的数据表中。

# 5字段扩展设置

字段的扩展设置是高级应用，如下图：



## 5.0 表单装载前数据填充

**需求背景：**

我们知道一个表单的数据包含四种模式，主表数据、从表数据（可多个）、一对多的数据（可多个）、还有一些附件与图片数据（可多个）。在一个表单被加载前用户往往要为这个表单初始化一些数据。比如发起一个请假流程，用户进入请假流程表单时，想把该用户主表数据比如电话、邮件、部门、部门负责人信息（主表数据），请假的历史数据（从表）都要初始化出来。

在比如：已知一个订单号在发起流程前要把该用户的客户基本信息（主表）与订购清单（从表）预先填充表单并显示出来。

这样的示例会很多，当然可以利用表单事件来完成，但是书写那个存储过程太麻烦，为了解决这个问题，现在抽象出来这种应用模式，以方便各位使用。

**设置方式：**

表单设计器-》扩展设置=》职





特别说明：

从表也是从表，应该为只读，才符合正常的逻辑，如果非只读也不能保存它。

## 5.1数据获取

什么是数据自动获取？

当一个字段的值是从其它表中取出来，或者通过后台的计算出来，或者通过js脚本计算出来，我们称为字段的数据自动获取。自由表单的获取方式分为，保存后获取与js计算获取两种方式。

自动获取分为如下几种方式：



### 方式1：本表单中数据计算

1, 表单中的字段计算分为主表字段之间的计算与从表之间的计算，它们的设置格式与约定规则都是相同的。如果设置的是表单的主表字段，那么是表单上主表控件值之间的计算了。如果设置的是从表的字段，那么就是表单列这间的计算。

2, 表达公式支持中文也支持字段名称，比如：你可以书写成 @单价\*@金额 也可以书写成@danjia\*@jiner

3, 计算是以javascript 方式完成的。所以当数值变化后就可以看到计算结果。

### 方式2：利用SQL自动填充.

比如: Select Addr From 商品表 WHERE No=@FK\_Pro FK\_Pro是本表中的任意字段名。

**应用案例：**在请假流程表单中获取两个表单日期（开始日期，结束日期）的相隔的天数。

**处理步骤：**

1. 首先在建立一个函数 GetSpanDays(dtForm,dtTo) 。 dtFrom，日期从，dtTo 日期到。
2. 设置一个SQL，放在文本框里。  
    For sqlserver: SELECT GetSpanDays(@DTForm,@DTTo)   
    For Orace : SELECT GetSpanDays(@DTForm,@DTTo) FROM DUAL
3. 把此字段设置为只读（只有只读的字段才可以被自动计算）。

**工作原理：**

CCForm在保存后，首先检查一个只读字段是否有扩展属性，如果有就把设置的信息SQL中的变量替换下来，执行这个SQL,把返回值赋给当前这个只读字段。

### 方式3：本表单中外键列。

比如:表单中有商品编号列,需要填充商品地址、供应商电话。

### 方式4：对一个从表的列求值。

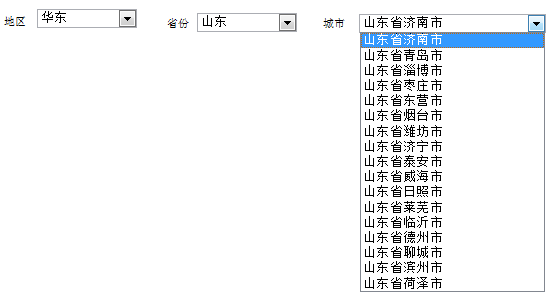
对于有从表的有效，在案例视频中有一个报销总金额=从表的小计中求合。

**总结：**对于3，4两种模式都可以变相的用第二种模式实现它。

## 5.2级联下拉框

### 主表中的级联

在表单中级联下拉框经常用到，下面是一个典型的事例。

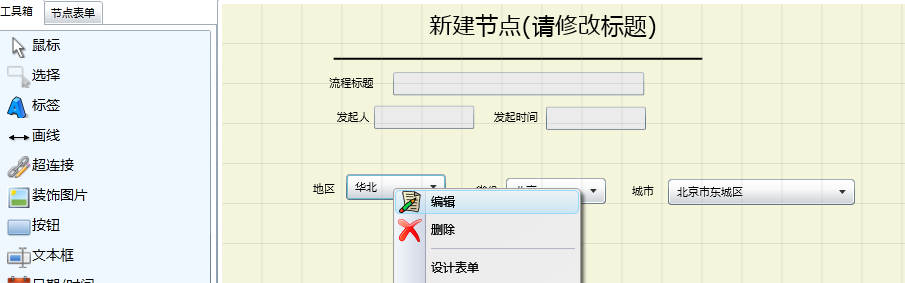


我们分析一下，地区与省份是关联的，省份与城市是关联的。这样的业务规则我们可以抽象出来两两关联，就可以解决了无限级关联的问题。

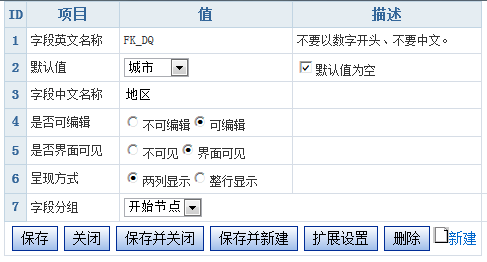
**配置步骤如下：**

第一步：配置片区与省份的关联。

在片区控件上点右键编辑它=》扩展属性。



点编辑进入下一步骤。



点扩展设置进入下一步骤。



点新建

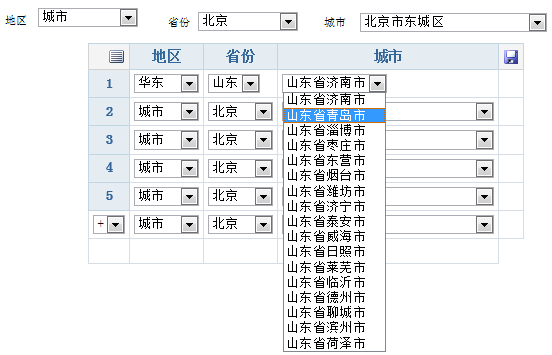


通过sql方式联动子下拉框，也就是片区联动省份。

重复上一步骤，让片区省份联动城市。 无限级的下拉框的联动，就可以制作成功了。

### 从表中的级联

从表的级联与主表的级连设置方式一样的。



上面是效果图。

### 常见问题

1. **在从表中设置级联：**没有正常设置下拉框的默认值，导致整个页面被保存。例如：省份设置默认为山东，城市设置默认为北京，这样就会造成全页面的从表存盘问题。
2. 待续。。。

## 5.3表单字段的输入js脚本验证

标题：表单字段输入验证.

关键字: 表单设计器, 字段验证. workflow,ccform, ccflow.

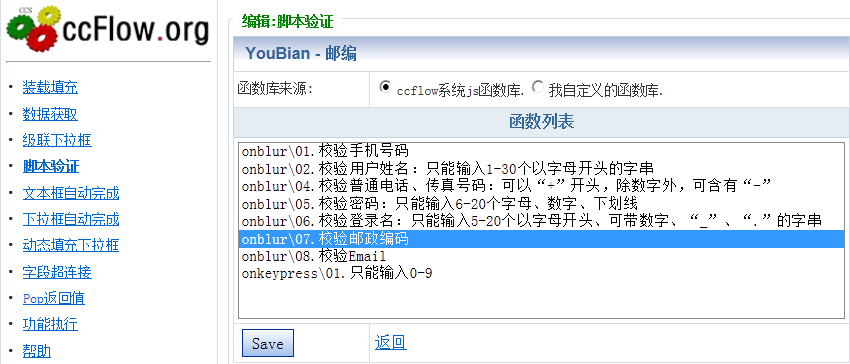
说明：对控件的用户输入采取输入控制，与输入后的校验。比如校验是否是e-mail, 是否是手机号，身份证号等等。

### 设置步骤

操作路径：字段属性, 扩展设置，脚本验证。

字段列表:

验证邮政编码设置:



函数库分为：系统函数库，与我自己定义的函数库。

系统函数库是ccflow系统自带的函数库，它包括一些基本的，通用类型的验证，比如：电话、邮件、邮政编码。

如果验证需要特殊的验证，那就需要自定义函数库了，如何定义个性化的函数请参考下一章节。

点保存按钮就可以了，系统将会自动返回字段列表。

这样的一个脚本验证就完成了，把它非常方便了为用户的输入完成了验证设置。

### 如何设置自定义验证函数?

按照如下步骤设置：

1. 建立一个js脚本文件。
2. 根据此脚本文件的事件，D:\ccflow\VisualFlow\DataUser\JSLib 对应的文件目录里。  
   比如 ：onblur 事件函数放在onblur目录里，onkeypress事件函数放在onkeypress目录里。
3. 函数名必须为小写,事例如下：

|  |
| --- |
| function isMobil(ctrl) {  var patrn = /^[+]{0,1}(\d){1,3}[ ]?([-]?((\d)|[ ]){1,12})+$/;  if (!patrn.exec(ctrl.value))  {  alert('非法的手机号码.');  return false;  }  return true  } |

### 脚本验证的工作原理

系统函数的目录为：

D:\ccflow\VisualFlow\Data\JSLib

用户自定义的函数目录为：

D:\ccflow\VisualFlow\DataUser\JSLib

表单设计器自动产生的脚本文件位与

D:\ccflow\VisualFlow\DataUser\JSLibData

在你执行保存时，表单设计器就把所有的函数调取出来，生成一个脚本验证文件，此文件命名的格式为。 表单编号+”.js” 。 并存放在： D:\ccflow\VisualFlow\DataUser\JSLibData 目录下面。

表单载入时，就把此脚本文件自动的调入到界面上。

### 如何通过获取表单控件值？

我们知道在ccform表单设计器中有4种控件分别是dropdownlistbox, textbox,checkbox,rirdionbutton.



系统内置了三个函数，分别是获取相应对象的值，请参考以上图片。

ReqTB 获取TextBox类型字段的输入值。

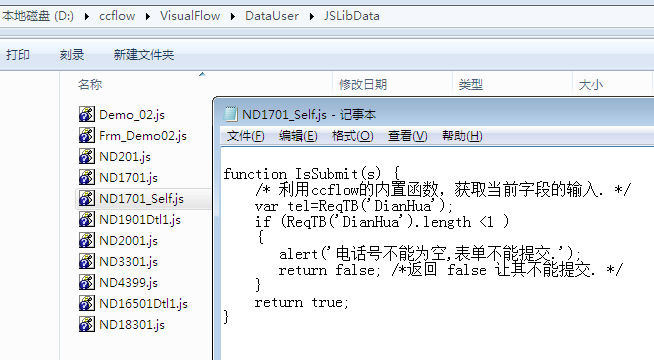
ReqDDL 获取DropDownListbox的选择值。

ReqCB 获取CheckBox的值。

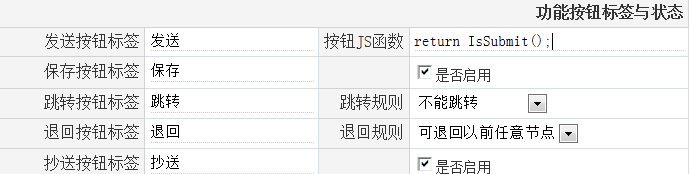
### 如何在提交前做一个完整的表单输入验证?

关键字: ccform 字段验证 工作流程引擎 提交前输入完整性验证

设置步骤：

1. 建立一个自定义的js文件，放在D:\ccflow\VisualFlow\DataUser\JSLibData\下面。
2. 命名格式为表单ID+”\_Self.js”。比如:ND1701\_Self.js
3. 在自定义js文件(ND1701\_Self.js)中创建一个一个校验函数。  
   

3, 打开该节点属性=》找到功能按钮标签与状态=》发送按钮标签=》按钮js函数文本框中填写: return IsSubmit();



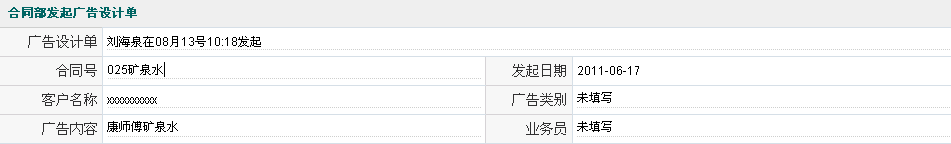
## 5.4自动完成

自动完成包括三大部分，主表中的自动完成，主从表的自动完成，从表中的自动完成。

### 5.4.1 演示环境说明

### 5.4.2主表中的自动完成

以如下图应用为例，来说明：



上图是用傻瓜表单设计器设计的，但是对自由表单设计器一样有效。

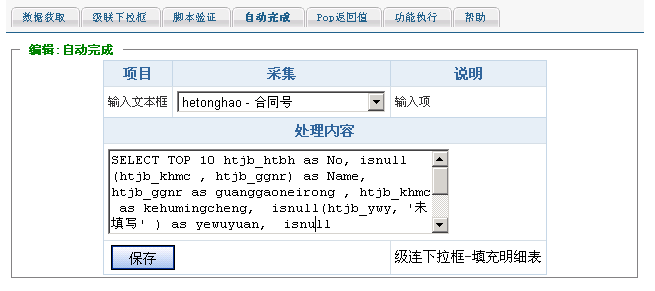
用户操作希望输入合同编号后，能够把客户名称，广告内容，业务员，广告类别自动的填写上。在合同号的控件中，输入合同编号希望有自动完成效果，其它几个控件是只读的。在输入合同号后，自动填写其它的数据内容。

我们看看运行的效果：



配置步骤：

打开表单设计器-》找到合同号字段-》编辑属性-》扩展设置-》自动完成。



我们把复制出来的sql内容显示如下：

|  |
| --- |
| SELECT TOP 10 htjb\_htbh as No, isnull(htjb\_khmc , htjb\_ggnr) as Name, htjb\_ggnr as guanggaoneirong , htjb\_khmc  as kehumingcheng, isnull(htjb\_ywy, '未填写' ) as yewuyuan, isnull(htjb\_ggxs, '未填写' ) as guanggaoleibie  FROM dbname.dbo.guanggao\_htjbxx WHERE htjb\_htbh LIKE '@Key%' |

这个sql是一个结果集合：它必须有No,Name 两个列，这两个列是用来控制显示在自动填写的下拉框中。其它的列名需要与主表的字段对应上才能自动填充上去。

### 5.4.2主从表的自动完成

在上一节中我们讲到了主表的自动完成，现我在们讲主从表的自动完成。

以以下通用需求为例来说明：

一个合同号跟着几个广告线路，在输入合同号后，主表的信息管理输出出来后，还要把从表的线路信息关联出来。这样的业务规则称为主从表的业务关联。

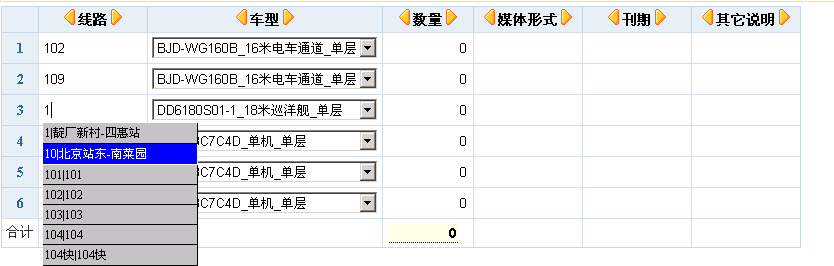


### 5.4.3从表中的自动完成

在从表中如果输入了线路名称需要把线路的其它信息在从表中反映出来。

这里有两种业务规则模式：

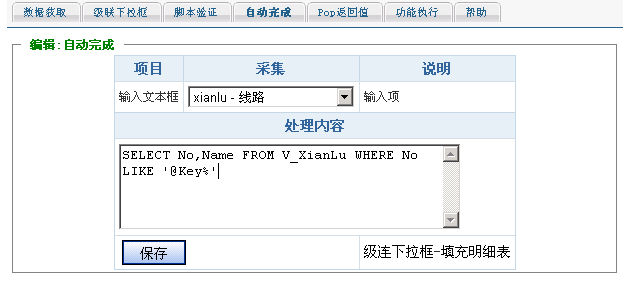
1. 一条从表记录对外键范围无要求的。  
   比如：输入一个合同编号，把合同名称，客户名称，客户地址，合同类型都显示这一个行的其它控件上。配置方式同主表的配置方式。
2. 一条记录对于外键字段的范围有要求。

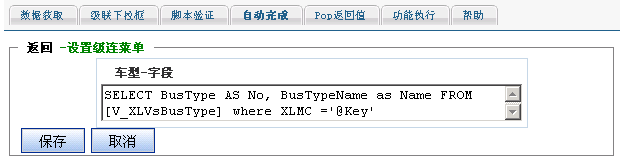
仍然以以上业务规则为例：一个公交公司有n多车类型，有一个表存放公交车类型。一个线路的公交车类型是有范围的，就是公交车类型表的一个子集。  


如上效果图：当我输入个线路时，回车后系统就会把该线路下的车型过滤出来。

配置方式：

**第一步：**设置线路下拉框的自动完成。



**第二步：**点保存右边的 [级联下拉框] 连接，为车型下拉框设置过滤sql。  


## 5.5 字段超连接

字段超连接是把只读的并且是可见的字段不以控件方式显示，而是用户超连接的方式显示。

字段超连接支持傻瓜表单设计与自由表单设计器。表现在表单主表数据与从表数据上。

### 在主表表单设计上的表现



这个功能适用于数据挖掘，功能导航。

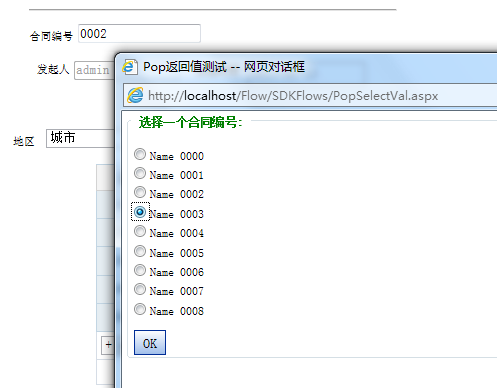
### 在从表的表现与设计



## 5.6 Pop返回值

### 事例：

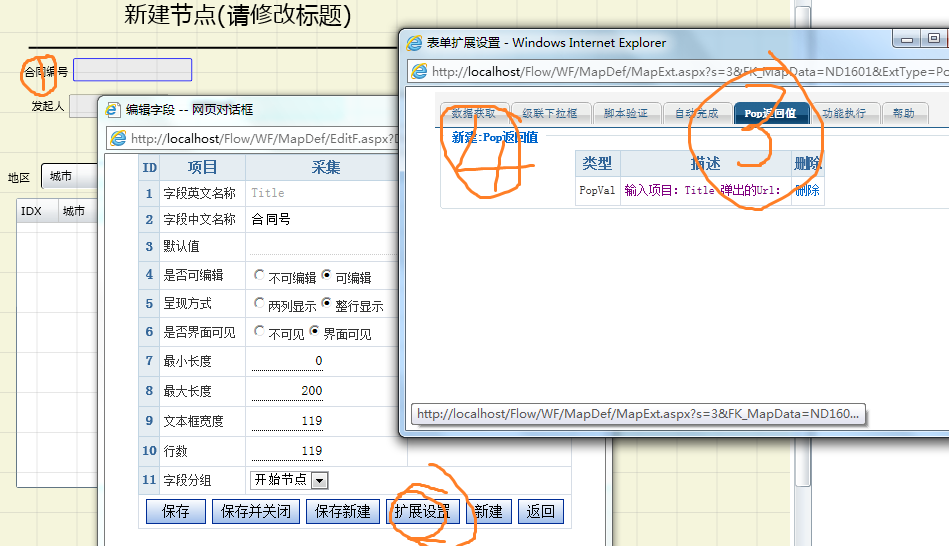
我们对弹窗取值，并不陌生，如下图：



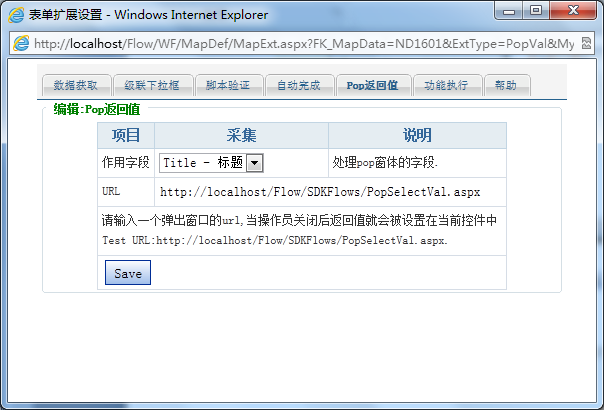
在一个数据控件上双击它，或者点旁边的icon 出来一个值的查询窗口，点确定后，选择的值返回到该控件上。

### 设置方式

字段-》编辑-》扩展属性-》



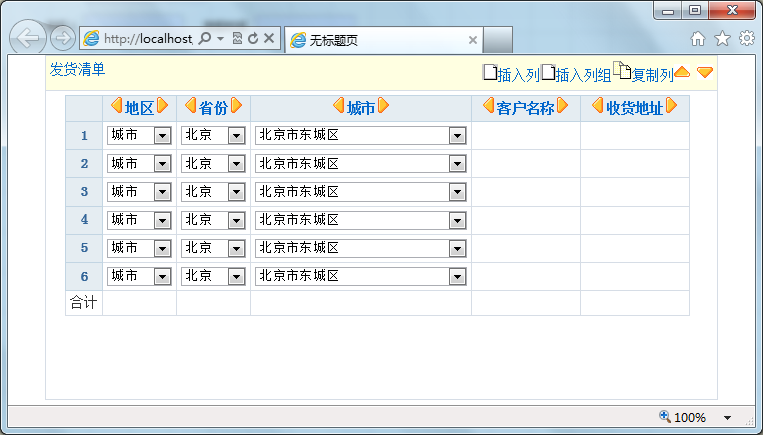
进入下图：



在url文本框中填写一个url。

# 6从表的设计

## 6.1从表的设计器



## 6.2从表的属性

|  |
| --- |
|  |
| **表英文名称：**  从表编号。  **表中文名称：**从表名称 **物理表名称:** 默认值可以改变默认与从表ID一致。  **操作权限：**详见下页介绍。  **对从表操作的权限控制:**  是否可以修改，增加行，删除行。 |

## 6.3从表属性说明

对于上图的从表容易理解的属性不在编写，本章节主要讲解难于理解的属性。

### 6.3.1操作权限控制

对从表的数据权限的控制，用与分合流中的权限控制。比如

操作员: 工作ID: 流程ID:

此属性是对于分合流的从表数据权限进行控制，

在分合流程中，每个一分支，填写完数据后，需要汇总到指定的节点中去。

### 6.3.2是否起用审核字段：

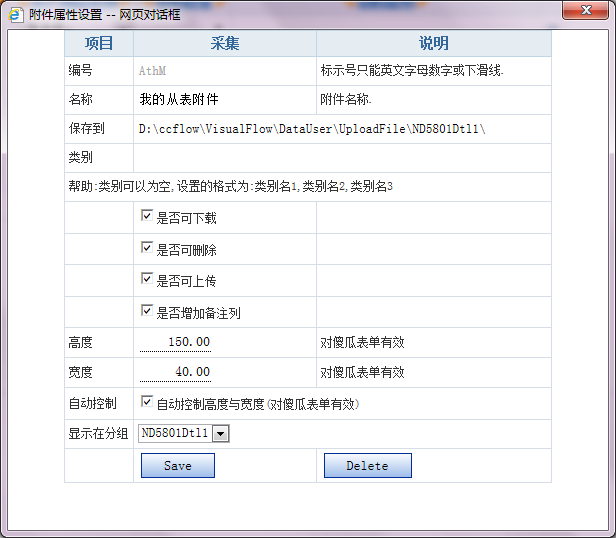
**此部分详见:**  
<< ccflow视频教程05-从表多级审核模式的流程开发.ppt>>  
<< ccflow视频教程05-从表多级审核模式的流程开发.wmv>>

### 6.3.3是否启用多附件

在现实应用中会经常出现在一个从表中每条数据也需要附件功能的支持，为此ccform 在从表属性中扩展了此功能，用户只需要启用这个设置就可以了。

如果启用了此功能，如下图，保存后在底部控制按钮会多显示一个附件属性按钮。点此按钮后可以进入编辑此附件属性功能。另外在最后显示出来附件按钮ICON.





其它说明：

1. 从表的附件也支持复制。
2. 从表的附件权限控制与主表附件权限控制相同。

运行效果：



### 6.3.4显示格式

显示格式分为表格方式与卡片方式，这两种方式应用于不同的场合。

#### 表格方式:



#### 卡片方式:

****

### 6.3.5越位处理

省略

## 6.4 从表事件



从表的事件的格式与主表的事件格式一样的描述。在实际工作中，我们经常用到从表保存完后要处理一些业务逻辑。

比如：要根据主表的字段更新一些从表的信息。

比如：在每条数据更新前后都要做安全性校验。

**特别说明：**

对于从表保存前，保存后两个事件，用@符号获取的变量是主表字段的变量。

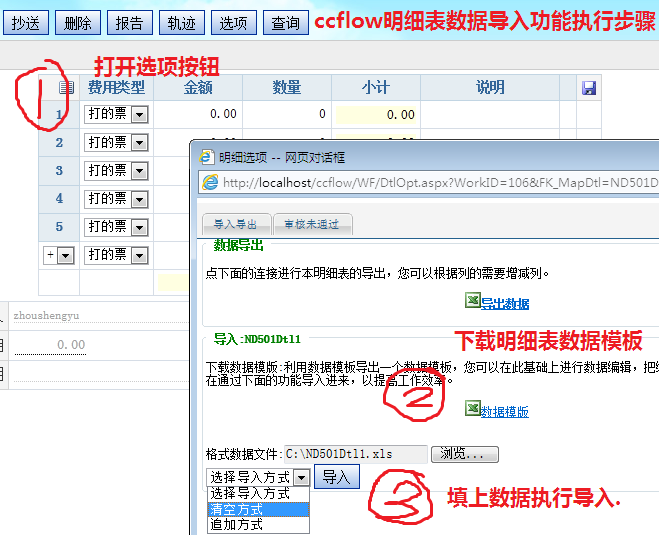
其它的事件是从表的字段。

@WebUser.No,@WebUser.Name@WebUser.FK\_Dept 全局字段约定不变。

## 6.5为从表设计导入数据模版

**应用场景：**对于大批量的明细表数据采集，在web上执行就会给用户带来不方便，用户迫切需要一个导入功能，比如事先把数据整理到excel中然后导入到表单明细表中。本章节介绍如何为ccform的明细表设置导入表单模板。

**执行效果：**

****

**用户点选项按钮，到处**

**制作步骤：**

第一步：根据明细表制作一个excel 表(注意:费用类型为枚举或者外键类型仍然可以导入。)  


第二步：把此模板文件按照该明细表的ID存储一个明细表ID+.xls 格式存储到  
D:\ccflow\VisualFlow\DataUser\DtlTemplete\ 下面.  
比如: D:\ccflow\VisualFlow\DataUser\DtlTemplete\ND501Dtl1.xls 文件.

第三步：打开明细表属性，启用设置可以导入功能。  


## 6.6 从表的多表头设计

**关键字: ccform 从表 多表头**

**需求背景：**

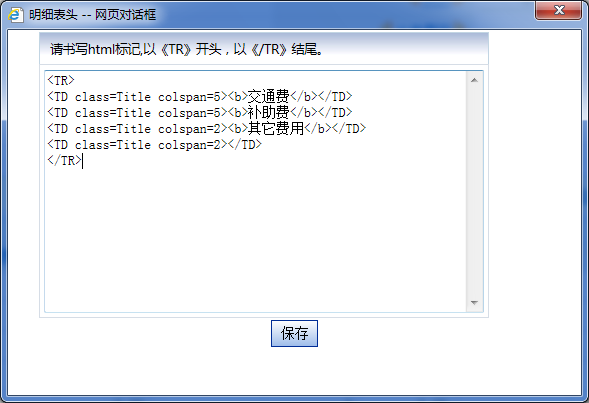
多表头是反映在复杂的报表中或者数据输入界面中，它是对用户的表头进行分组。

如下图：



**设计步骤：**

在设计模式下点多表头，在里面填写html脚本代码。



# 7一对多的关系设计

## 7.1什么是一对多的数据关系?

一对多的数据关系存储是工作中经常遇到，比如软件支持的操作系统，软件是一个表，操作系统是一个表，软件与支持的操作系统的关联信息我们称为一对多的关系存储。



## 7.2一对多的关系设计

设计器如下图：



**编号：**表单唯一的ID号。

**描述：**一对多的关系名称。

**主题数据源：**

就是用sql指定一个内容数据源，它需要三个列，编号，名称，分组依据，这里是操作员。

**分组数据源：**

就是用sql指定一个分组数据源，它需要二个列，编号，名称，这里是部门。

其它：Sql 支持变量比如: SELECT No,Name FROM PORT\_DEPT WHERE NO LIKE ‘@WebUser.FK\_Dept%’

## 7.3数据存储格式是什么

认识到ccflow数据存储格式，可以在此基础上进行数据需要。

所有表单的一对多的关系处理请见表Sys\_M2M.

# 8 一对多多的关系设计

## 8.1应用背景

## 8.2设计方式

## 8.3数据存储格式

# 9表单的单据打印

## 9.1单据打印总体概述

**应用场景：**

我在设计流程时间，经常遇到需要打印单据。比如：请假条，出库单、入库单。

这些单据需要打印出来，存档，或者递交给相关的办事人员。

**制作步骤:**

首先要明确单据打印的数据来源，明确表单的每个字段。

其次需要制作单据模板。

最后把此模板绑定到数据源上。

**单据打印功能：**

报表设计所见既所得

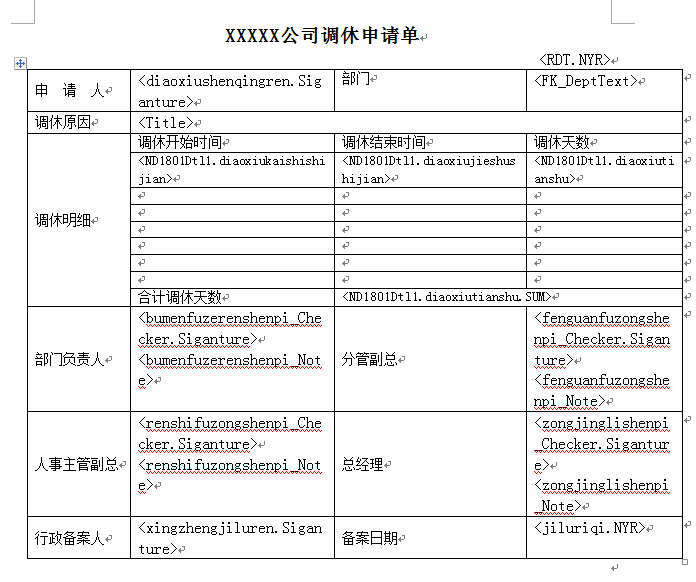
支持多表单数据源

支持字段的转意输出

支持从表的打印

## 9.2单据模板的制作

Ccflow单据模板是一个rtf格式的文件，如下图。



用尖括号表示变量，变量就是表单的字段。

## 9.3单据输出格式约定

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **单据模板事例**  -----------------------------------------------**基本的字段输出**------------------------------------------------  普通类型字段输出:  文本输出 ：<Title> 输出内容: 按文字输出.  文本输出 ：<RDT> 输出内容: 2011-05-11  日期文本输出 ：<RDT.NYR> 输出内容: 2011年05月11日  金额文本输出 ：<baoxiaojine> 输出内容: 101.123  金额人民币输出 ： <baoxiaojine.RMB> 输出内容: 101.12  金额人民币大写输出 ：<baoxiaojine.RMBDX> 输出内容: 壹百零壹元壹角贰分  布尔类型数据输出： <IsPass> 输出 0  布尔类型数据转意输出： <IsPassText> 输出 否  布尔类型数据转意输出： <IsPass.Yes> 输出 [X]  布尔类型数据转意输出： <IsPass.No> 输出 [V]    以性别为例  枚举类型输出: <XB> 直接输出 1  枚举类型输出标签: <XBText> 直接输出 男    以部门为例  外键类型输出: <FK\_Dept> 直接输出 01  外键类型输出标签: <FK\_DeptText> 直接输出 总经理室  **------------------------------------------------- 图片标记的输出 -------------------------------**  **-- 用于图片签名。**  签名：<Rec.Siganture>  **-- 手写版图片输出。**  手写版：<OID.BPPaint.MyPaintID> 格式为: OID+”.BPPaint.”+标识ID.  **图片输出:** <OID.Img@AppPath//DataUser//LogBiger.png>  多用于ICON输出, AppPath 是应用程序主目录, OID. 是为了能够取出参数。  **------------------------------------------------- 明细表的输出 -------------------------------**  **以下是表格的明细输出：**   |  |  |  |  | | --- | --- | --- | --- | | 报销内容 | 单价 | 数量 | 小计 | | <Dtl.BXNR> | <Dtl.danjia> | <Dtl.shuliang> | <Dtl.xiaoji> | |  |  |  |  | | 合计 |  | <Dtl.shuliang.SUM> | <Dtl.xiaoji.SUM> | |

## 9.4单据生成的工作原理

**首先：**系统把表单模板的所有的内容都读取到内存里，然后遍历每个字段串，找到尖括号的位置，并且找到尖括号的变量。以确定这些变量的集合，并把它放在缓存里。

**其次：**根据变量集合获取它的变量值，替换模板内容。

**最后：**把替换后的模板生成一个以.doc 结尾的文件后缀文件，保存到指定的位置。

技术处理难点：

1. 对从表的处理。
2. 对模糊变量的修复处理。

## 9.5单据表单的分类

单据打印分为流程单据打印与节点表单打印, 设置打印内容这两种情况没有区别.

节点表单单据打印需要在节点属性里上传，而流程表单的打印只需要按照约定设置就可以了。

### 9.5.1节点表单

#### 节点表单设置步骤

1, 制作表单模板。

2, 打开节点属性，把表单绑定到节点上。

#### 节点单据打印物理表与文件存储

**配置表单时**

1, 在上传单据模板时，系统把把单据模板存储在.   
 D:\ccflow\VisualFlow\DataUser\CyclostyleFile

2, 单据模板表存储表 WF\_BillTemplete

**生成单据时:**

1, 生成文件位置 D:\ccflow\VisualFlow\DataUser\Bill\.   
存储格式为.

1. 在单据注册表里增加一条记录 WF\_Bill.

### 9.5.2 流程表单

#### 流程表单单据设置方式

1, 打开 D:\ccflow\VisualFlow\DataUser\CyclostyleFile\FlowFrm\

2, 按照流程编号建立一个目录比如: 001 所有该流程下的表单都在这里面.

D:\ccflow\VisualFlow\DataUser\CyclostyleFile\FlowFrm\001

3, 找到节点编号, 比如我们对107 节点设置打印, 建立在流程编号目录下建立目录.

D:\ccflow\VisualFlow\DataUser\CyclostyleFile\FlowFrm\001\107

所有的流程表单模版都放在这里面.

4, 表单命名规则: 流程图表单编号+'.'+ 自定义的表单名称+'.rtf'.

比如: D:\ccflow\VisualFlow\DataUser\CyclostyleFile\FlowFrm\001\107\00001.企业简介.rtf

5, 以上说明了对于单个表单中设置一个打印的单据, 打印的格式是按照rtf 格式的内容进行.

系统就会取表单 00001 的表单数据，替换 00001.企业简介.rtf 里面的内容进行执行打印.

6, 如果一个单据中有需要多个表单数据支持. 约定的格式为. D:\ccflow\VisualFlow\DataUser\CyclostyleFile\FlowFrm\001\107\all.企业简介.rtf

它的是意思就是把整个流程表单数据都放入里面.

## 9.6表单设计常见的问题

### 如何解决乱码的问题？

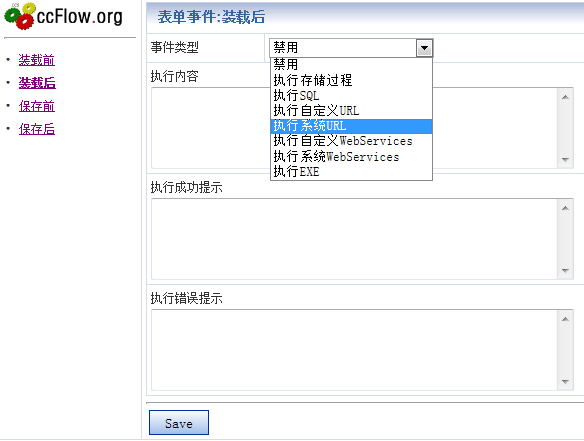
单据中的乱码是由单据模板中的字符的格式产生的，出现乱码就是模板修饰中有西文字符。最简单的办法，是全选，然后设置中文字体。

1. 模板里面不要有西文字符，西文字符容易产生打印出来的单据乱码，解决办法是全选然后选择中文字体。
2. 尽量不要用word来编辑尤其是高版本的word，要用写字板来编辑，对于word模版文件要执行全选，copy。粘贴到已经打开的空写字版文件里。
3. 如果变量替换不出来，有两个原因：1，拼写错误，ccflow没有特别说明都是区分大小写。2，格式错误：用记事本打开你的模板文件，搜索关键的字符： 不符合<youfield>的就把周围的格式删除了它。
4. 用案例模板以文件位于: D:\ccflow\VisualFlow\DataUser\CyclostyleFile\单据打印样本.rft

# 10表单事件

## 10.1事件概述

设计图如下：



表单事件是在表单装在前后，保存前后发生的事件，它是编辑业务逻辑的接口。所以掌握表单事件非常有必要。

每个表单事件都有一样的如下4个属性：

**事件类型：**

业务逻辑处理的依托载体，你是用存储过程，还是用sql.。。。 来实现你的功能。

**执行内容：**

就是执行的主体，书写时支持全局变量@WebUser.No,@WebUser.Name 。支持本表单的字段变量@+字段名。

**执行成功时：**

执行成功时提示的信息，信息内容支持变量，同执行内容一样。

**执行失败时：**

执行失败时提示的信息，信息内容支持变量，同执行内容一样。

**事件类型解释：**

**禁用：**

不启用

**执行存储过程：**

请在执行内容里填写一个存储过程名称，注意表达式支持变量。

**执行SQL:**

请在执行内容里填写SQL，注意表达式支持变量。

如果有多条sql ，可以用@GO 分开。比如：

DELETE WF\_Emp WHERE 1=2

@GO

DELETE WF\_Emp WHERE 2=3

**执行自定义URL.**

用户在表达自己的业务逻辑时，往往用代码编写比较顺手，在这样的应用条件下，你可以编写一个url 向这个url中传递不同的标记，返回不同的结果来，返回空表示顺利执行。

实例：在执行内容里设置一个http://myserver/Do.aspx?DoType=aaaaa，创建一个Do.aspx 根据DoType 标记这不同的内容处理。

如果顺利处理了就返回空，出现异常一定要返回: Error+”异常信息。”

处理返回值用: this.Response.Write("Error:"+msg); 方法.

Ccform 处理的机制是，使用 HttpWebRequest 类静默的执行URL ,然后获取返回的内容。如果检查到前几个字符是Error 就认为是异常ccform 就会抛出异常。

**执行系统URL**

为了方便期间ccform定义一个系统的url, 如果选择这种类型，执行内容里就可以为空。系统在执行的时，就会自动访问。

<http://myserver/ccflow/DataUser/AppCoder/FrmEventHandle.aspx>

如果在FrmEventHandle.aspx文件中实现业务逻辑里面的一个详尽的案例。

**执行自定义WebServices: (未完成)**

**执行系统WebServices: (未完成)**

**执行Exe. (未完成)**

## 10.2装载前事件

在这里您可以写事件，

## 10.3装载后事件

## 10.4保存前事件

## 10.5保存后事件

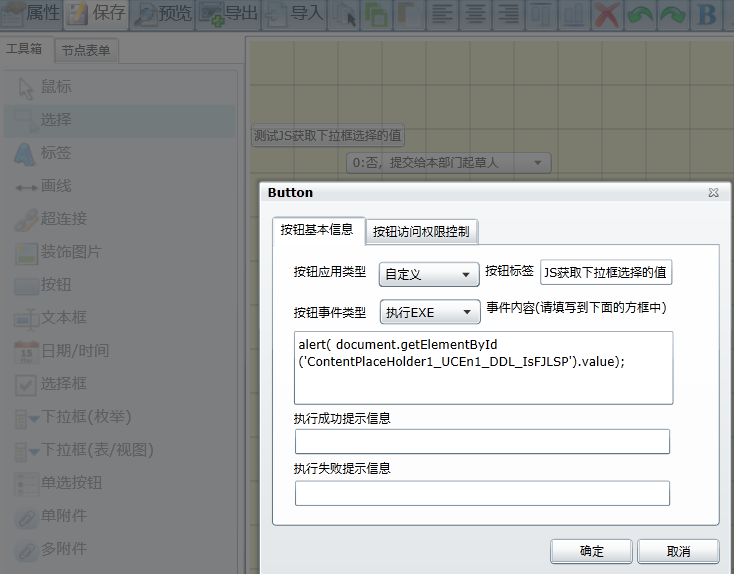
# 11技术文摘

## 11.1常见问题类

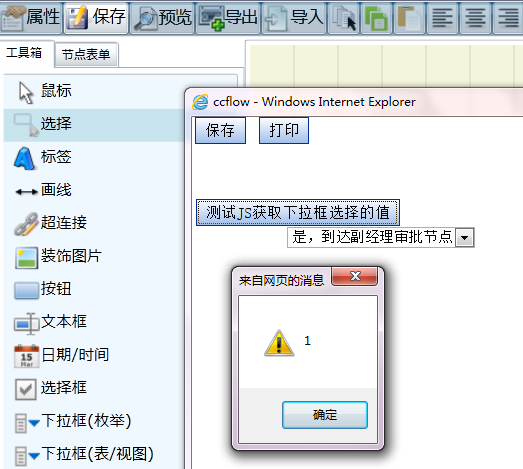
### 11.1.1在表单设计器中，如何获取自由表单下拉框的值？

**应用背景：**在表单的前台展现中，我们经常用到一个控件的js事件需要获取其它的控件的值，这里我们演示一下，如何在button 事件里获取一个下拉框的的选择的值。由此您可以开发灵活的js脚本应用了。

**准备工作：**在表单设计器的画布里，设置两个控件，一个button一个dropdownlistbox. 在button事件里做如下设置，如图所示。



**预览效果：**



### 11.1.2如何用自由表单设计器映射我的物理表?

首先建立一个物理表或者视图，这个表或者视图必须有No,Name两个列。

**=============OVER=============**