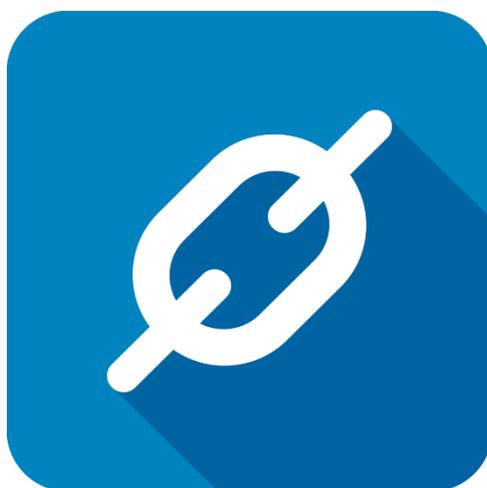


TotalLINK

产品手册



上海朝识智能科技有限公司

2017 年 5 月

移动应用

目 录

移动应用.....	2
1 系统登录.....	5
2 用户主菜单.....	6
2.1 首页界面操作.....	7
2.2 收藏界面操作.....	8
2.3 我的界面操作.....	9
3 系统功能说明.....	16
4 数据列表.....	17
5 无参数数据列表.....	18
6 内容显示-折叠模式	19
7 内容显示-展开模式	21
8 内容显示-简单列表	23
9 内容显示-自定义	25
10 模型显示内容控制.....	27
11 内容显示-高级设置	29
12 定义操作-数据挖掘	30
13 数据挖掘模型的定义.....	31
14 参数选择列表.....	34
15 参数选择列表的模型定义.....	35
16 为选择传递参数.....	36
17 选择列表显示的内容.....	39
18 扫描输入.....	40
19 语音输入.....	41
20 附加功能操作.....	42
21 带参数的附加功能操作.....	45
22 附加功能的参数“再”确认	47
23 附件功能操作的额外参数定义.....	50
24 附加功能处理后的数据刷新.....	52
25 图表显示.....	54
26 百度 ECHARTS3 集成.....	55
27 图表模型设计.....	61

28	地图数据示例.....	65
29	组合图表数据示例.....	70
30	多批次数据加载.....	72
31	数据刷新.....	75
32	数据处理操作.....	76
33	数据处理-添加.....	77
34	数据处理-删除.....	79
35	数据处理-编辑.....	80
36	预处理及后处理功能.....	82
37	外部功能列表.....	84
38	直接展现图表.....	86
39	数据挖掘直接显示图表.....	88

文档控制

■ 主要内容

本文介绍 TotalLINK app 基本功能及使用，请根据需要学习相关内容。

■ 更改记录

日期	版本	作者	备注
2017-05	1.0	Randy	初始发布
2023-07	2.0	Liz	修订内容，删除数据分享功能

■ 支持版本

非特殊说明的功能，默认前后版本都支持

仅支持V20版本及以后版本的功能点

1 系统登录

16:37 4G

中文



网址

账号

密码

自动登录

登录

[注册](#) | [关闭](#)

- TotalLINK 移动应用与服务端应用程序共享统一系统信息，包括：
 - 统一的用户权限管理
 - 统一的模型及数据
 - 统一的针对模型的各种操作功能
- 当前移动端分为 IOS 和 Android 两个版本，分别对应不同的平台，功能完全一致，**IOS 下载需要通过兑换码下载，TotalLINK 团队会提供对应的兑换码**
- 系统登录时，登录到 TotalLINK 的 WEB 服务端
- 用户登录时，可以使用与服务端相同的用户，也可以单独设置移动端用户

2 用户主菜单



- 用户登录后，显示在服务器端为用户分配的菜单组列表
- 底部按钮说明：
 - 首页：显示最近使用的模型、展现模型
 - 收藏：跳转到收藏夹列表
 - 我的：跳转到设置界面

2.1 首页界面操作

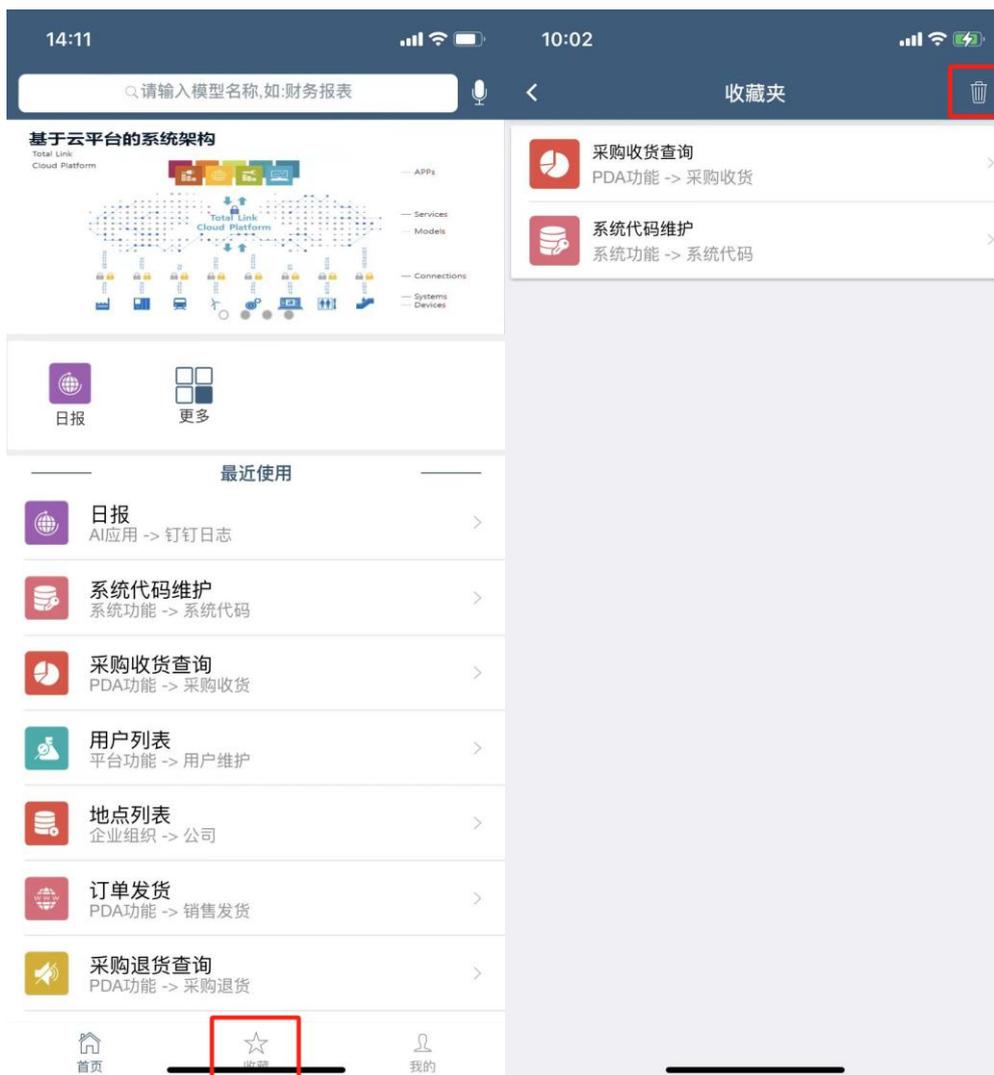
- 首页界面由轮播图、我的模型、最近使用列表构成
- 点击“更多”图标，进入全部模型界面
 - 长按模型图标对模型进行收藏，可以在收藏界面查看
 - 点击“编辑”按钮，将常用的模型添加到首页更多栏里面
- 最近使用列表显示最近使用的 8 个模型，选中其中一个模型列表向左滑动可对这个模型进行收藏





2.2 收藏界面操作

- 展现收藏夹列表
- 点击“删除”图标清除收藏夹列表



2.3 我的界面操作

- 点击“我的”，进入“设置”界面



- 展现内容如下图
 - 账号与安全，显示登录账户信息



- 用户操作手册，具体可参考《[系统功能说明](#)》
- 隐私
 - 清空历史数据
 - 清空缓存图片
 - 关闭定向推送功能



■ 通用

- 多语言，在 TotalLINK 设置好对应的翻译，可切换语种



- 扫描设置

- ◆ 始终扫描解析
- ◆ 广播模式:开启后需重启 APP，功能生效，**仅限安卓 PDA**
- ◆ 扫描后保存按钮：当该按钮处于 on 状态的时候，在扫描框扫描成功后，

则会自动保存，并提示是否继续添加，可选择继续添加，也可跳转回到查询结果界面



➤ 内容设置

- ◆ 记录上一次位置
- ◆ 返回首页清除记录
- ◆ 重置后聚焦
- ◆ 取消自动聚焦
- ◆ 连续添加按钮: 可以全局定义添加页面是否显示连续添加按钮。1 标识显示，0 隐藏，默认显示。

如果想单独修改模型的扫描模式，可以在模型 JS 模块中写入以下代码，表示隐藏连续添加按钮

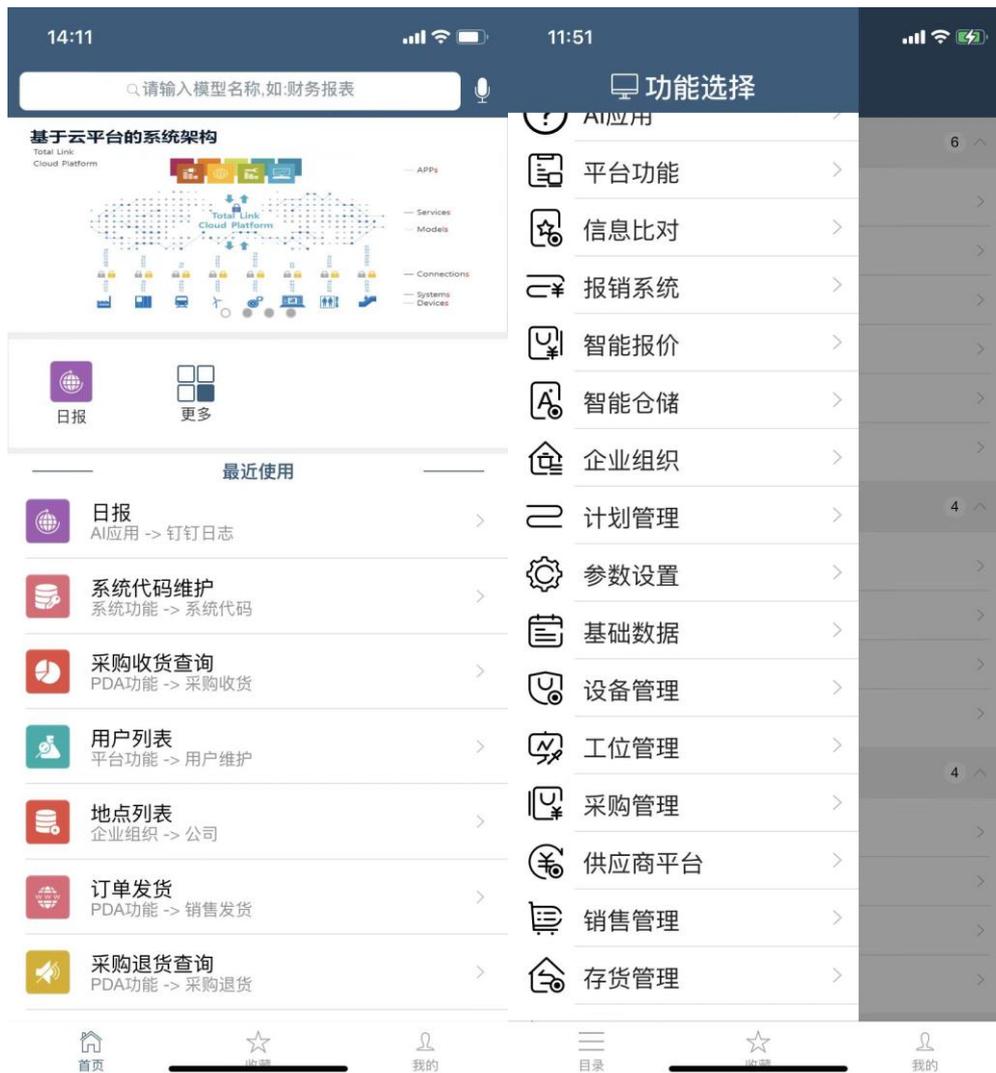
```
LINK_AUTO_ADD="0"
```



➤ 主页设置

- ◆ 应用图标，目前默认是应用图标模型

◆ 模型列表，老版的模型列表



应用图标模式

模型列表模式

- 界面尺寸设置，调整界面的字体，上下、左右边距



- 离线设置，在 total link-2.1.1 -101 版本才有

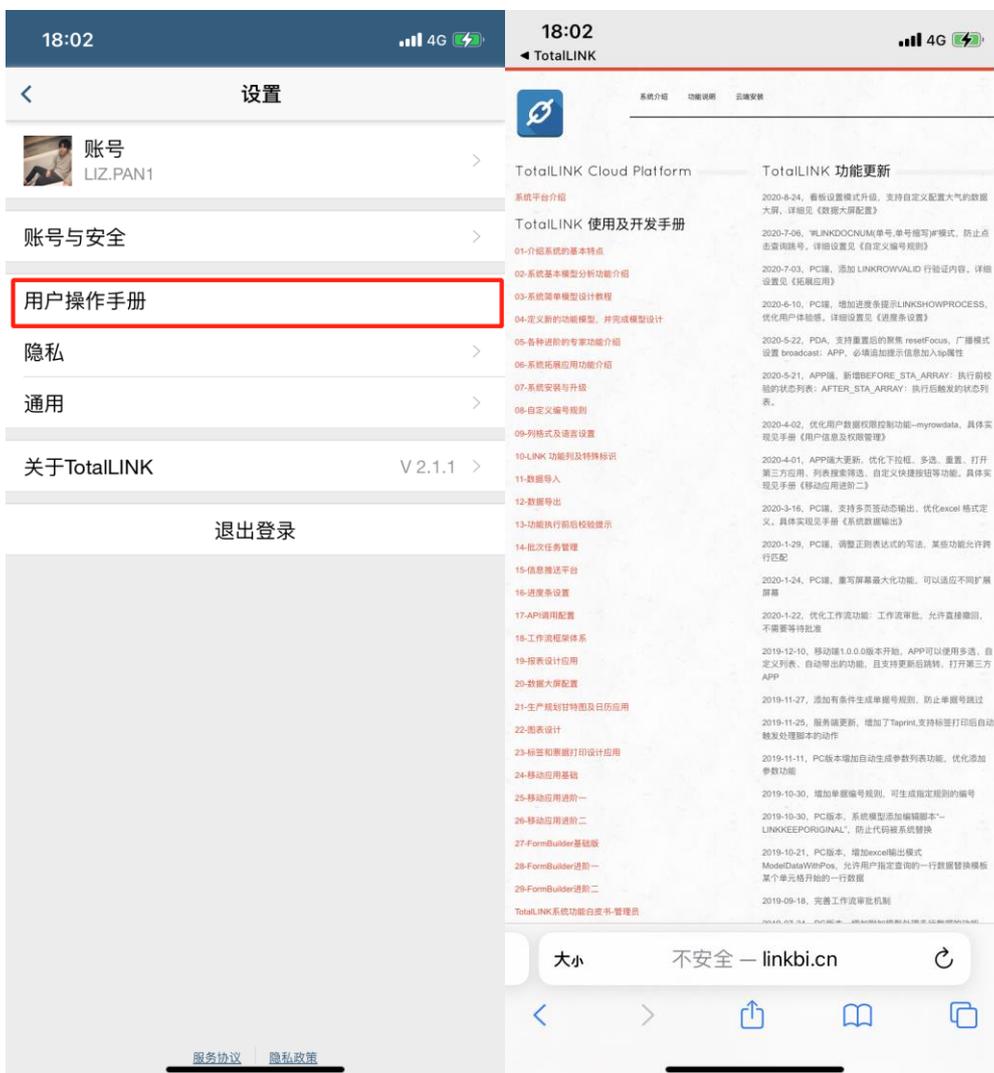
- ◆ 离线模式
- ◆ 下载并缓存数据
- ◆ 提交并同步数据
- ◆ 查看已缓存的模型



- 关于 TotalLINK，展现 TotalLINK 版本号
- 退出登录按钮



3 系统功能说明



- 本功能列出了系统的各类详尽的功能文档列表，包括本文档
- 具体的文档内容，可以至以下地址下载
 - <http://www.linkbi.com.cn>

4 数据列表



- 各具体的数据模型功能，通常是以数据列表的形式显示的
- 数据显示的内容与模型的定义有关，后文详述通过模型设计控制显示的内容

5 无参数数据列表

- 点击具体的功能时，根据模型定义，会出现两种情况
 - 一是该模型没有参数，则根据模型直接加载数据
 - 二是模型有需要填写的参数，则打开参数输入界面
- 有参数输入时显示的界面如下图所示

11:00

< 订单发货

扫描输入区

订单号 请在扫描输入区扫订单号

收货日期 2023年7月28日

收货描述 请输入收货描述

供应商 供应商编码或描述模糊搜索

产品 产品编码或描述模糊搜索

订单状态

完成

我 你 忘记 好 不 没有 文 是

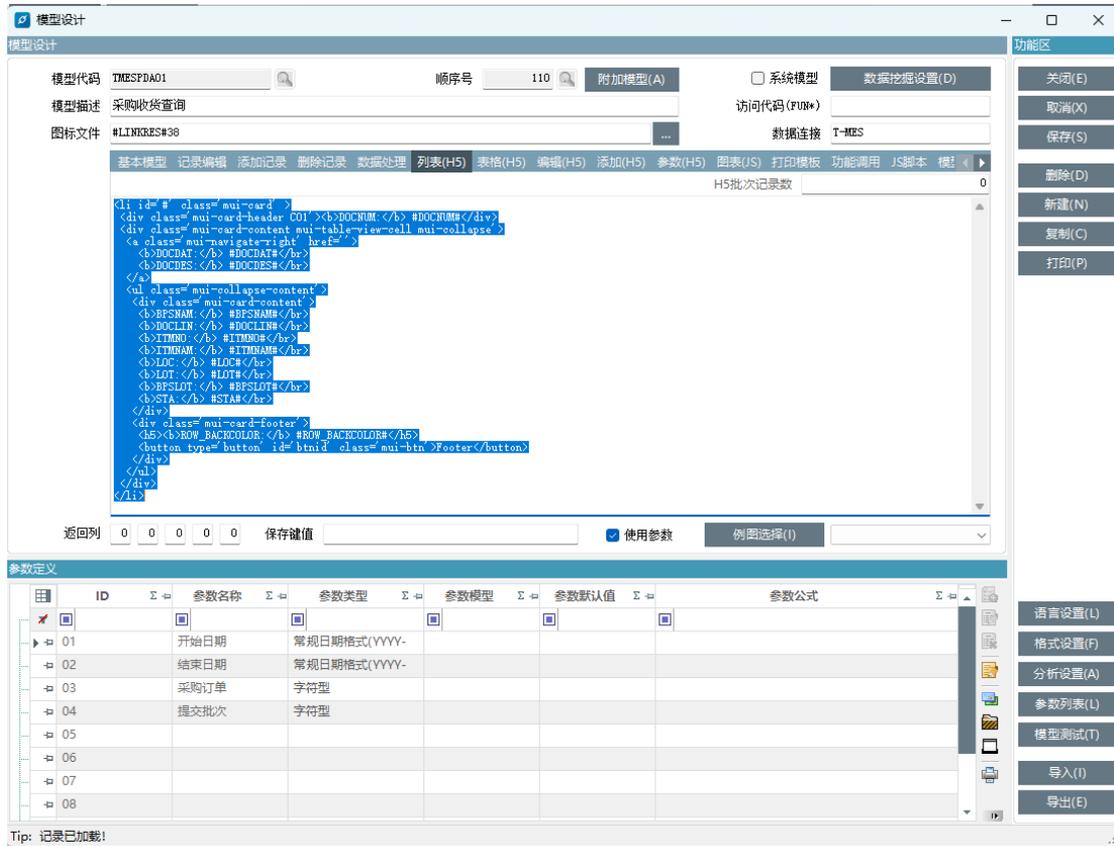
q w e r t y u i o p

a s d f g h j k l

z x c v b n m

123 空格 换行

6 内容显示-折叠模式

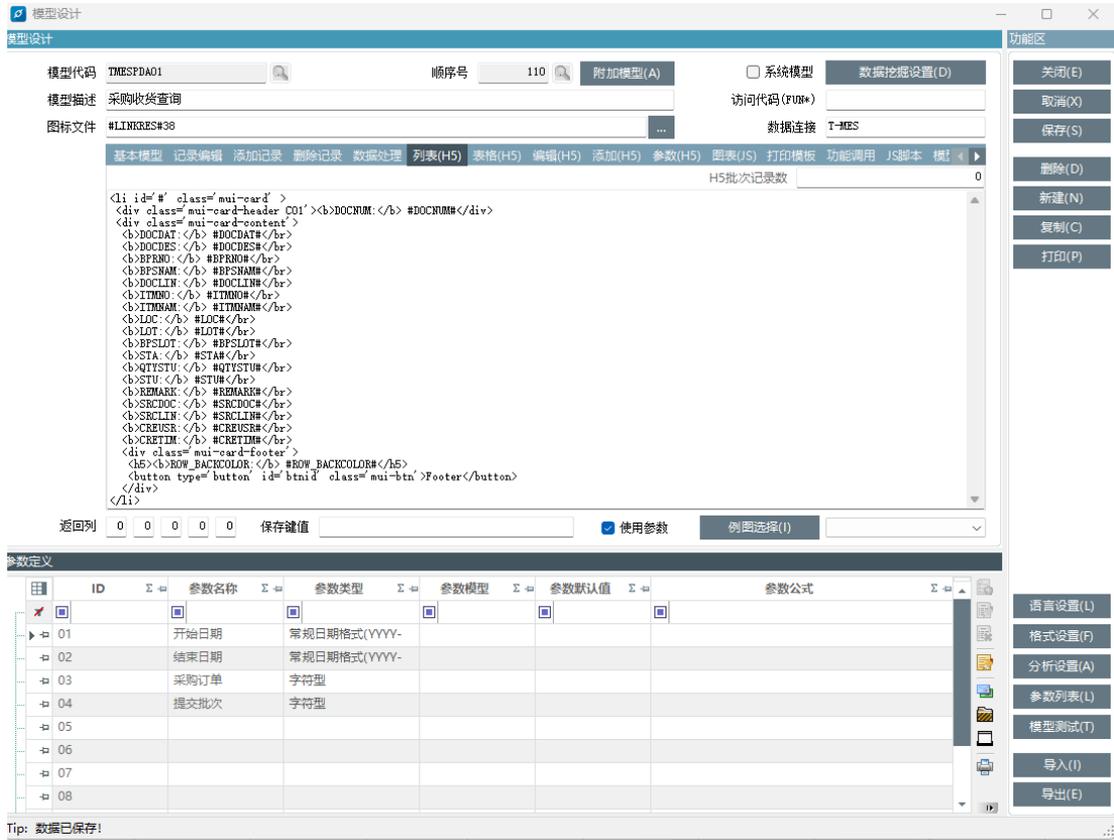


- 对于数据显示的内容，系统默认可以生成两种类型的显示列表
 - 折叠模式
 - ◆ 这种模式的列表内容，在移动端展示的时候，默认以第一列的内容作为标题，其后各列的内容作为展开的内容显示
 - ◆ 这也是在用户没有自定义显示列表内容的情况下，系统自动生成的显示格式
 - ◆ 通过系统功能的“右键菜单-生成 H5 折叠列表”实现自动生成 H5 嵌入脚本
 - ◆ 如果 H5 嵌入脚本中已经有内容，系统不再生成，需要先删除保持脚本内容空白然后才可以此功能
 - ◆ 用户可以根据需要对列表表头及内容进行调整
 - ◆ 调整是通过嵌入 H5 语法实现的

- 默认的折叠显示效果如下图所示



7 内容显示-展开模式

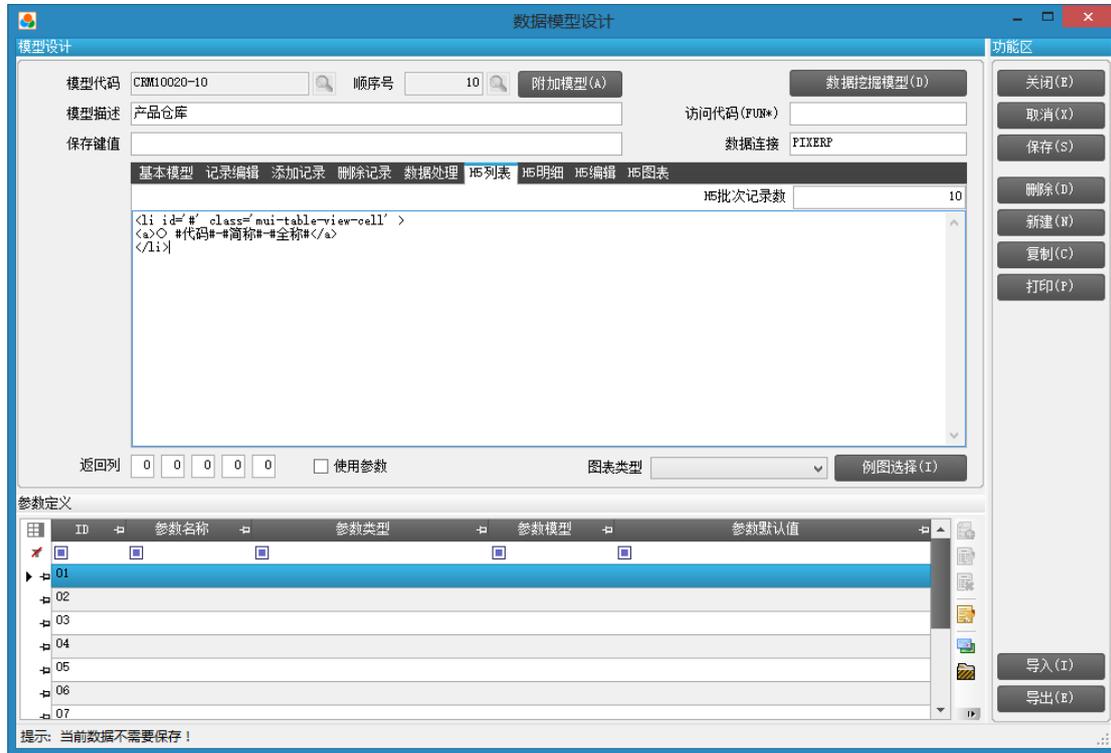


- 根据需要，可以生成自动展开的内容，即“卡片模式”
 - ◆ 这种模式的列表内容，在移动端展示的时候，默认以第一列的内容作为标题，其后各列的内容作为卡片的内容直接展开显示
 - ◆ 通过系统功能的“右键菜单-生成 H5 展开列表”实现自动生成 H5 嵌入脚本
 - ◆ 如果 H5 嵌入脚本中已经有内容，系统不再生成，需要先删除保持脚本内容空白然后才可以此功能
 - ◆ 用户可以根据需要对列表表头及内容进行调整
 - ◆ 调整是通过嵌入 H5 语法实现的

- 默认的展开列表显示如下图所示



8 内容显示-简单列表



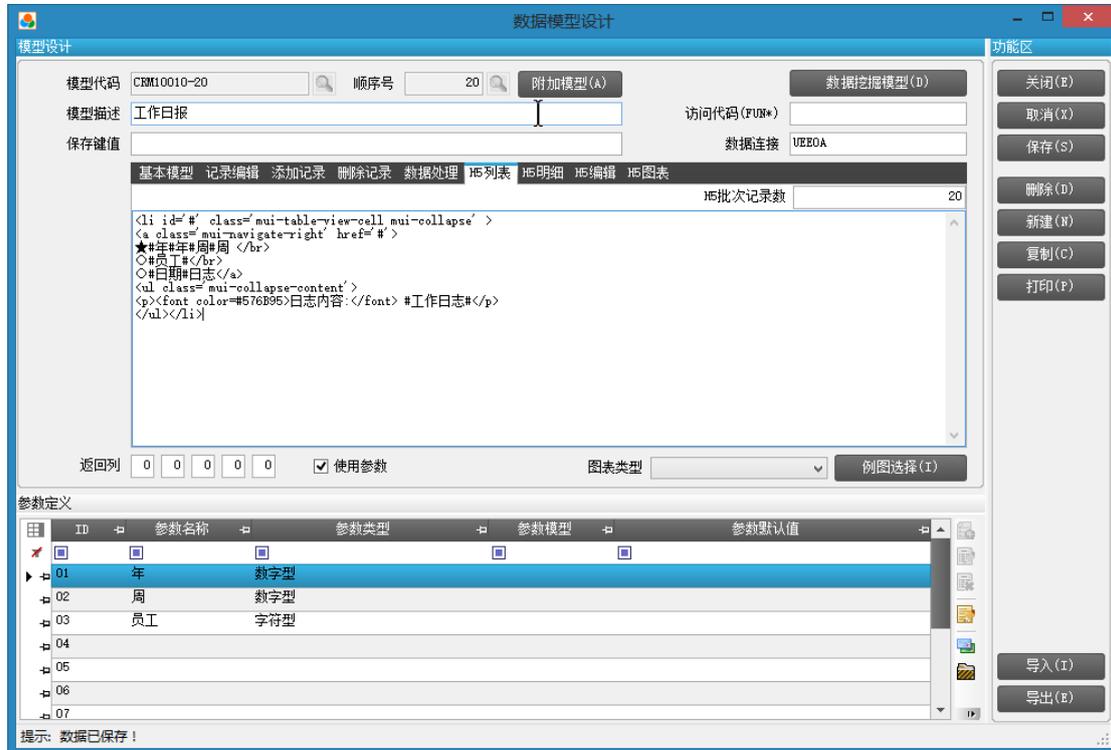
- 根据需要，用户仍可以通过自行设计脚本的方式实现其他显示模式
 - 比如直接显示一个简单的列表，不需要包含展开的内容，可以采用下面的H5脚本

```
<li id='#' class='mui-table-view-cell' >
<a>○ #代码#-#简称#-#全称#</a>
</li>
```

- 显示效果如下图所示



9 内容显示-自定义



- 对于显示的内容，标题可以不只一行，根据需要调整嵌入的 H5 脚本代码实现即可。
- 通过这种模式，可以实现类似主从表内容的显示
- 参考脚本如下：

```

<li id='#' class='mui-table-view-cell mui-collapse' >
<a class='mui-navigate-right' href='#'>
★##年##年##周##周 </br>
○##员工##</br>
○##日期##日志</a>
<ul class='mui-collapse-content'>
<p><font color=#576B95>日志内容:</font> #工作日志##</p>
</ul></li>
    
```

- 显示效果如下图所示：



10 模型显示内容控制



- 灵活运用模型设计，可以做到诸如利用数据控制显示颜色等强大的功能
- 比较本例模型的脚本如下：

```

SELECT FCYNAM_0 名称
      , FCY_0 LINKSUB10_地点
      , CRY_0 国家
      , MFGFLG_0 - 1 AS LINKBOOLEAN 生产
      , SALFLG_0 - 1 AS LINKBOOLEAN 销售
      , PURFLG_0 - 1 AS LINKBOOLEAN 采购
      , WRHFLG_0 - 1 AS LINKBOOLEAN 存货
      , FINFLG_0 - 1 AS LINKBOOLEAN 财务
      , FINRSPFCY_0 AS 对应财务地点
      , LEGCPY_0 所属公司
      , BIDNUM_0 LINKROWIDCOLOR
FROM   FACILITY
ORDER BY FCY_0
    
```

- 其中的一个字段为“LINKROWIDCOLOR”
 - 以 LINKROWID 开头的目的是使得该列默认成为隐藏列

- 对应显示的嵌入 H5 脚本为:

```
<li id='#' class='mui-table-view-cell mui-collapse'>
<a class='mui-navigate-right' href='#'>
<font color=#LINKROWIDCOLOR#>#LINKSUB10_地点# - #名称#</font></a>
<ul class='mui-collapse-content'>
<p><font color=steelblue>存货/财务:</font> #LINKBOOLEAN 存货#/#LINKBOOLEAN 生
产#</p>
<p><font color=steelblue> 销售 / 采购 / 生产 :</font> #LINKBOOLEAN 销售
#/#LINKBOOLEAN 采购#/#LINKBOOLEAN 生产#</p>
<p><font color=steelblue>公司/财务地点:</font> #所属公司#/#对应财务地点#</p>
</ul></li>
```

- 在脚本中，通过引用#LINKROWIDCOLOR#作为字体的颜色，对显示的内容进行控制
- 在实际应用中，很容易实现基于不同条件显示为不同颜色等操作
- 下图为颜色控制的另一个例子

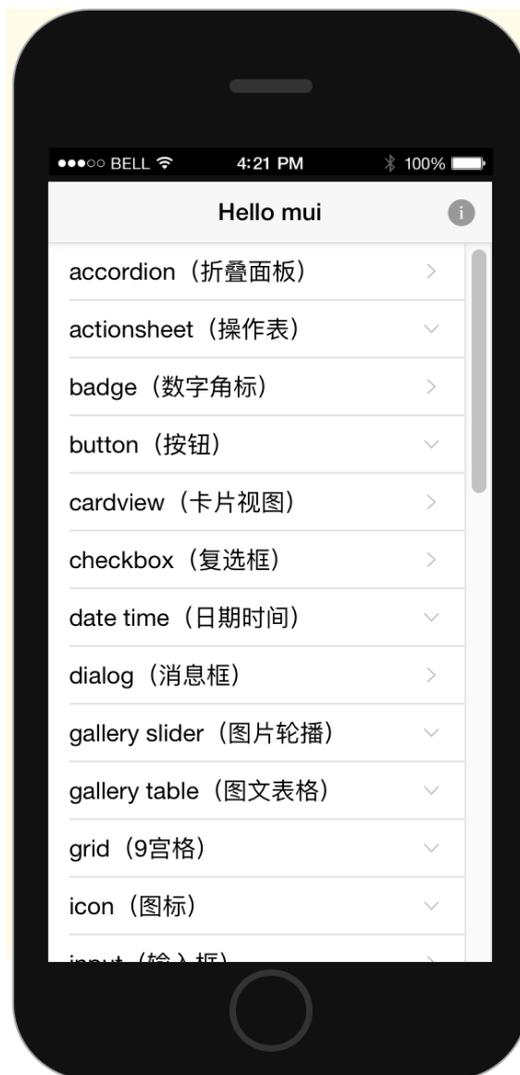


11 内容显示-高级设置

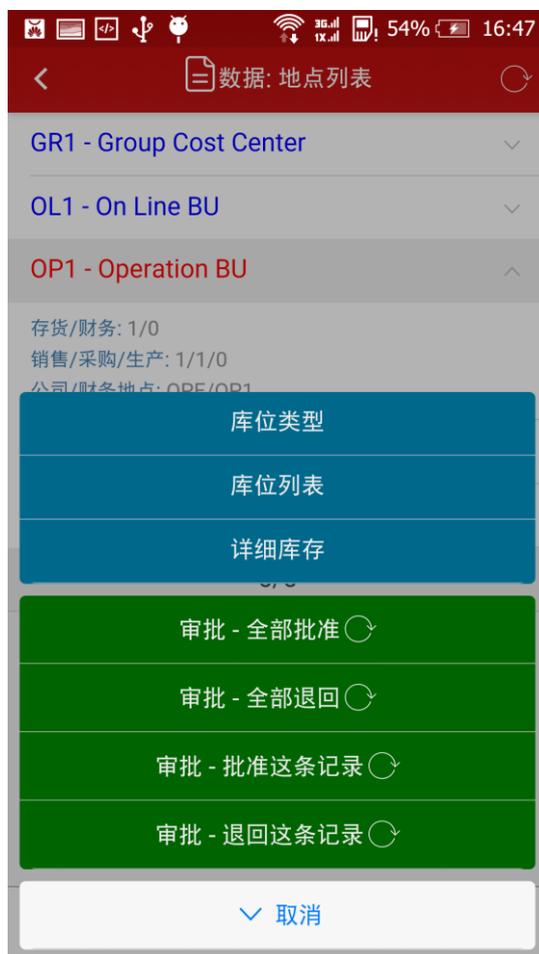
TotalLINK 平台的移动端显示采用 Dcloud 前端框架设计，支持标准的 H5 语法脚本嵌入。对于复杂的内容显示，可以参考该平台的相关内容。

相关内容可以参考：

<http://dev.dcloud.net.cn/mui/>

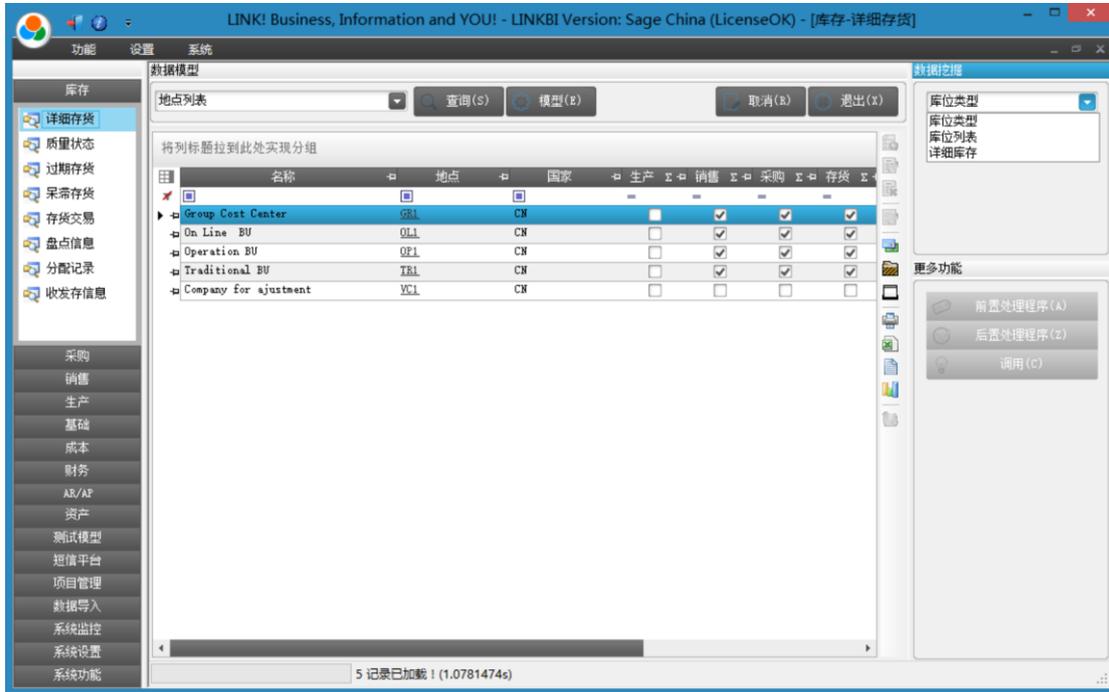


12 定义操作-数据挖掘

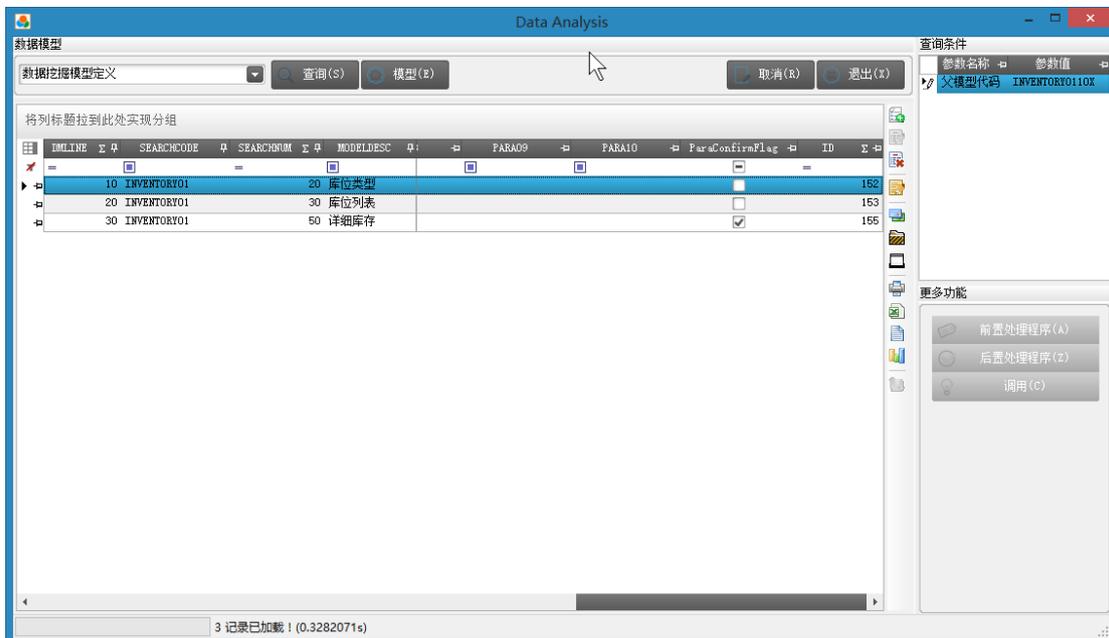


- 针对模型可以定义各种不同的“数据操作”
- 最常用的是数据挖掘操作
- 数据挖掘操作在移动端以“长按菜单”触发
- 比如本例中的三个蓝色按钮
 - 库位类型
 - 库位列表
 - 详细库存

13 数据挖掘模型的定义



- 对应系统的模型定义中的数据挖掘模型



- 上图所列的数据挖掘模型在移动端显示的模式如下图



- 在数据挖掘模型定义中，有一个选项为 `ParaConfirmFlag`，这个选项用于控制进行数据挖掘的时候是否弹出挖掘条件界面
 - 当 `ParaConfirmFlag` 为“否”的时候，直接传递条件，并加载数据
 - 当 `ParaConfirmFlag` 为“真”的时候，在加载数据之前弹出一个参数确认界面，允许用户“确认参数”
 - 这时用户也可以修改参数，并使用新填写的参数查询数据
 - 此时，数据挖掘的参数界面与打开有参数的模型的参数输入界面相同



参数: 详细库存

地点	OP1
库位类型	OOS
库位	请输入库位
产品种类	请输入产品种类
产品编码	请输入产品编码
产品名称	请输入产品名称 
质量状态	请输入质量状态

主屏 扫描 查询

14 参数选择列表



The screenshot shows a mobile application interface for parameter selection. The title bar is red and contains a back arrow, a document icon, and the text "参数: 详细库存". The status bar at the top shows the time 17:41 and various system icons. The form consists of several rows, each with a label and a value or input field:

地点	OP1
库位类型	OOS
库位	请输入库位
产品种类	请输入产品种类
产品编码	请输入产品编码
产品名称	请输入产品名称 
质量状态	请输入质量状态

Below the form, there are two buttons: "取消" (Cancel) and "确定" (Confirm). Underneath these buttons is a list of options for selection:

- GR1-Group Cost Center
- OL1-On Line BU
- OP1-Operation BU
- TR1-Traditional BU
- VC1-Company for ajustment

- 如图所示，在输入参数时，某些参数可以通过列表选择的方式输入
- 参数选择的列表同样来自模型定义
- 参数列表设置可参考文档《[LINK26 移动应用进阶二](#)》

15 参数选择列表的模型定义



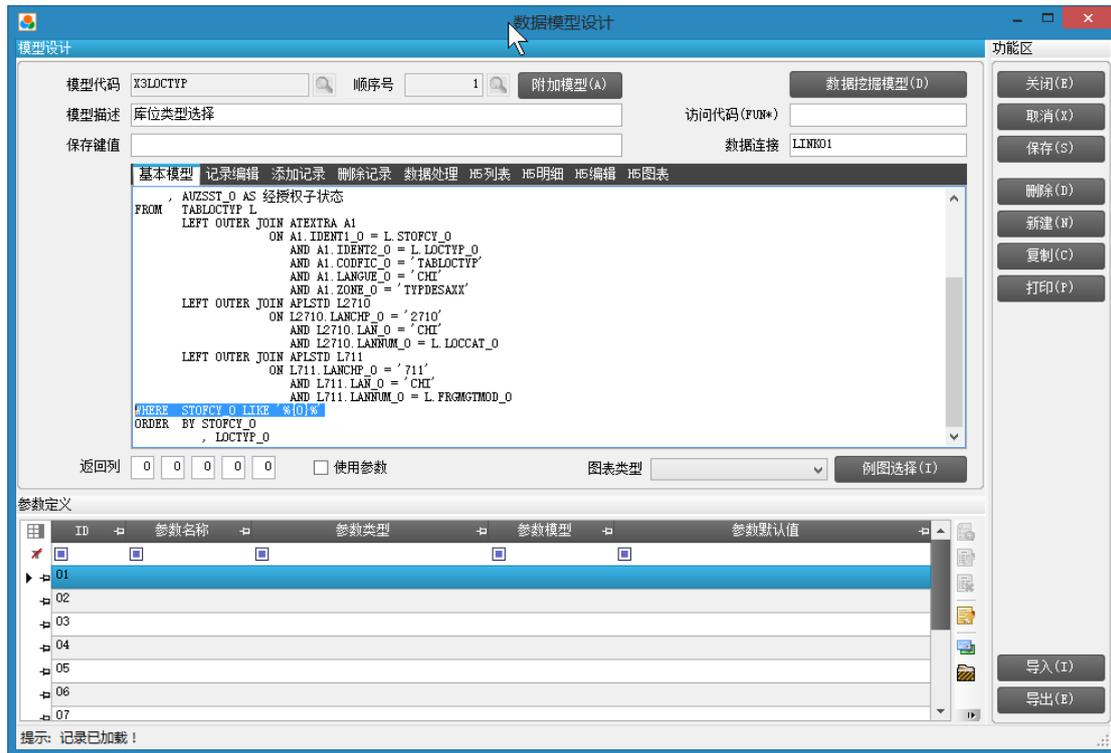
- 如图所示，对于参数 01-地点，是可以通过列表选择的
- 在模型定义时，指定参数类型为“字符型-移动端列表选择”

16 为选择传递参数



- 参数列表选择时，可以通过配置实现根据已经输入的内容或参数过滤选择的内容
- 比如，本例中的库位类型，显示的选择列表时“第一个参数：地点”所对应的库位类型

- 实现的方式如下图所示



- 在模型设计的时候，为查询“库位类型”的模型脚本设置了条件
- 条件中指定了“库位类型对应的地点”
- 与 TotalLINK 应用的 PC 端平台一致，以“{10}”表示当前已经输入的内容，可以控制选择列表显示的条目
- 相关内容可以参考模型设计文档

- 移动端的显示结果如图所示

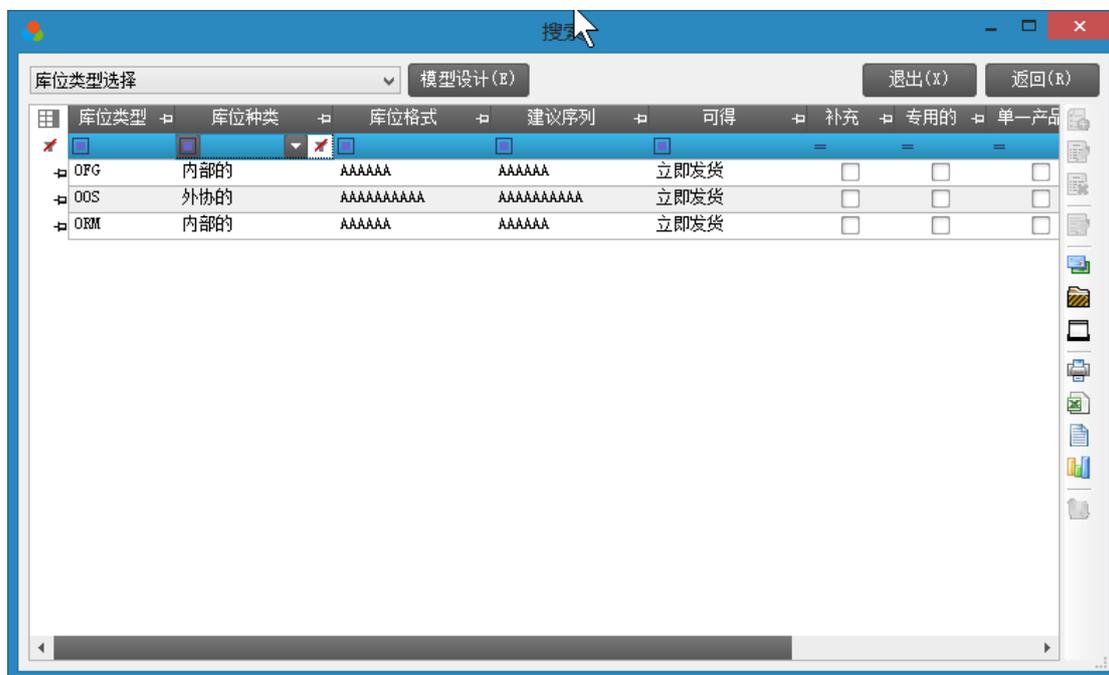
参数: 详细库存

地点	OP1
库位类型	OOS
库位	请输入库位
产品种类	请输入产品种类
产品编码	请输入产品编码
产品名称	请输入产品名称
质量状态	请输入质量状态

取消 确定

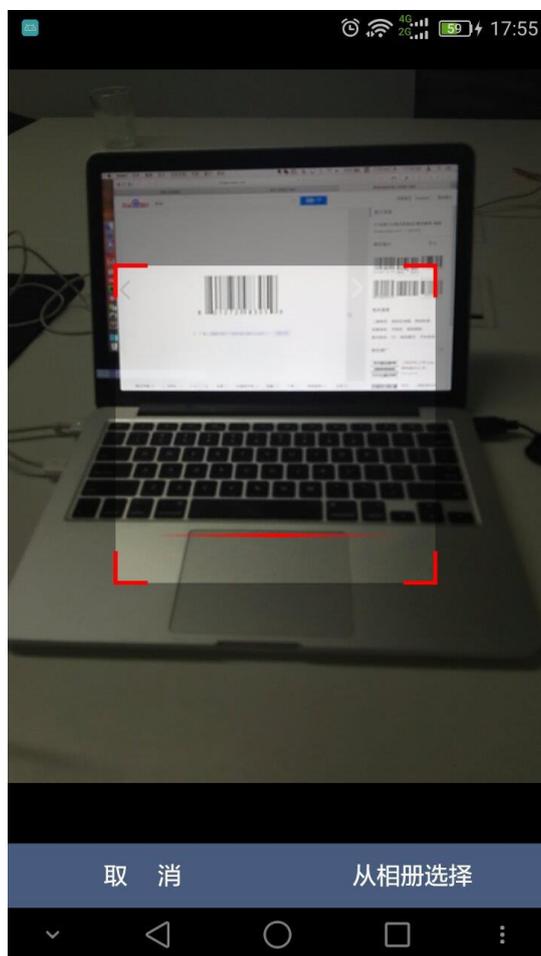
- OFG-内部的
- OOS-外协的
- ORM-内部的

17 选择列表显示的内容



- 通过模型定义，可以控制显示的内容
- 当前系统默认取模型显示数据的第一列作为列表返回值，以模型的“第一列-第二列”的内容作为列表的显示值
- 比如，本例中返回值为“OFG、OOS、ORM”等库位类型代码
- 对应的显示值列表为：
 - OFG-内部的
 - OOS-外协的
 - ORM-内部的

18 扫描输入



- 如图所示，所有的输入单元格都可以进行扫描输入
- 可以扫描一维条码或者二维条码完成内容的输入

19 语音输入

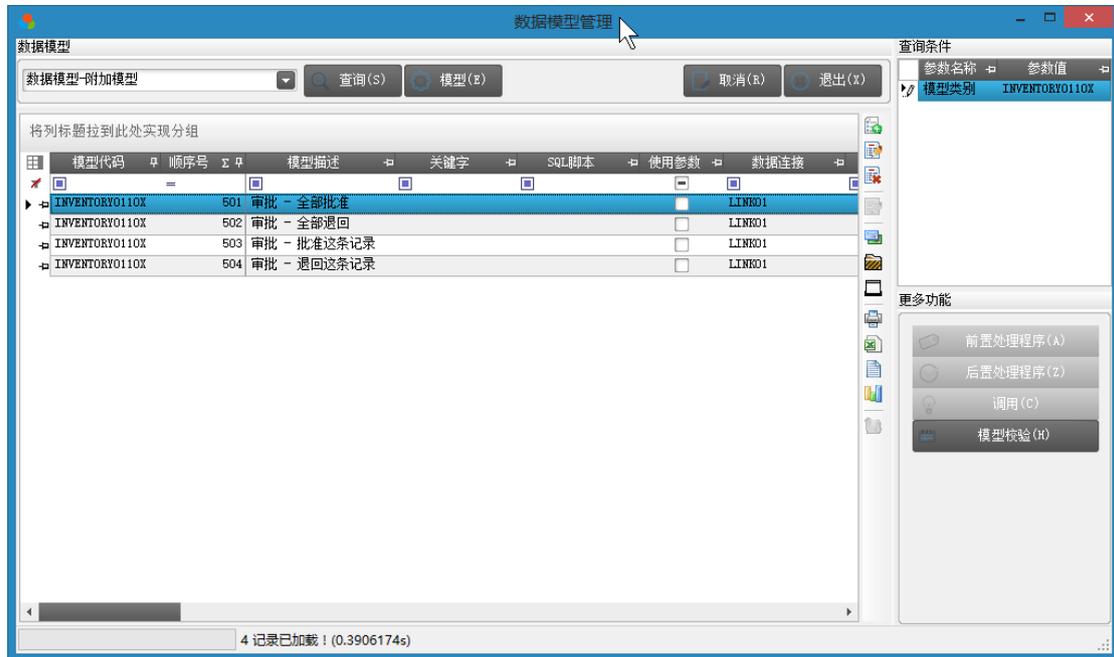


- 对于文本类型的内容，可以启用设备的语音输入模式
- 在模型定义时，对于参数，只需要指定参数类型为“字符型-移动端语音输入”即可

ID	参数名称	参数类型	参数模型	参数默认值
02	库位类型	字符型-移动端列表选择	X3LOCTYP	
03	库位	字符型	X3LOC	
04	产品种类	字符型	X3PRODCAT	
05	产品编码	字符型		
06	产品名称	字符型-移动端语音输入		
07	质量状态	字符型		
08				

- 当然，直接使用移动设备的语音输入法亦可方便完成语音输入

20 附加功能操作



- 通过附加模型，可以定义对于数据的各种功能操作，比如审批、加入购物车等等
- 对应的附加模型编号为 501-520
- 比如本例中，定义的附加功能包括
 - 审批-全部审批
 - 审批-全部退还
 - 审批-批准这条记录
 - 审批-退回这条记录
- 这里定义的附加功能在 PC 客户端使用时显示的效果如下

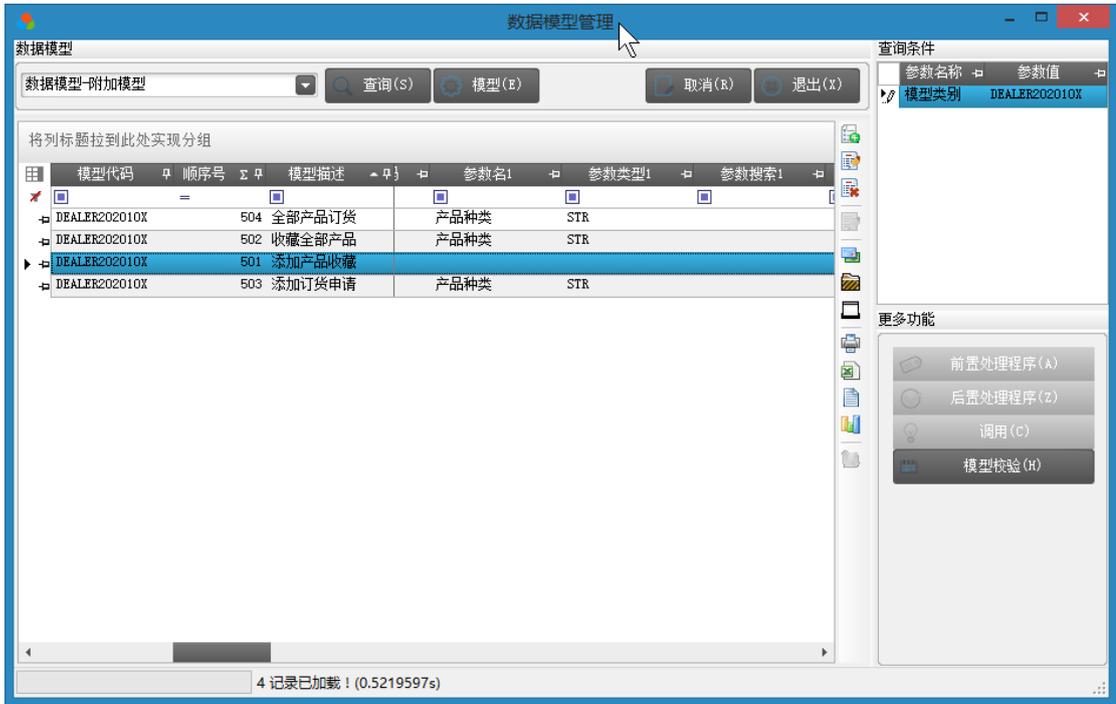


- 对应功能在移动端显示的模式如下图所示



21 带参数的附加功能操作

- 有些附加操作，需要在数据处理的时候，获取用户的输入操作
- 这种功能，可以利用模型的参数定义实现



- 如图所示，我们为数据模型定义了 4 个附加处理功能
- 在 PC 客户端进行附加处理操作的时候看到的界面如下



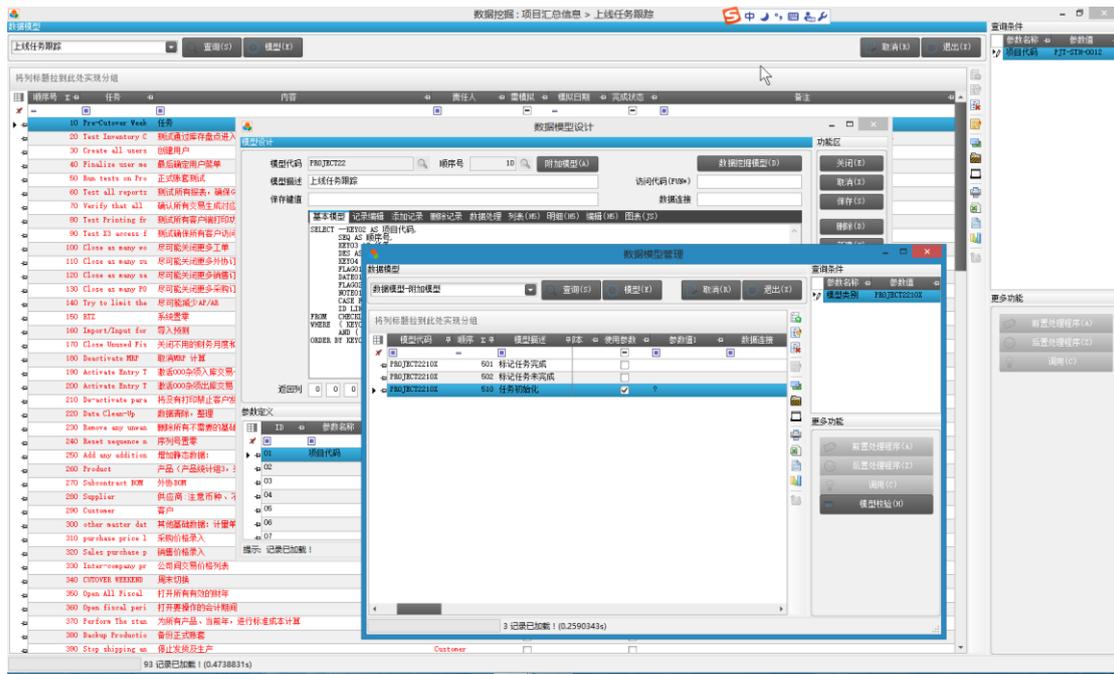
- 在移动端看到的界面如下



- 这里的四个模型中，可以看到后面两个在移动端界面显示的时候，标题中有“⊕”图标
- 该图标表示执行改功能时，需要依据用户输入数据进行操作

22 附加功能的参数“再”确认

某些情况下，对于附加功能，进行操作的时候，我们希望将用户传递的参数进行再次确认或更改。此时，可以对附加模型的参数用“?”标记即可。



- 如图所示，该模型定义了三个附加功能，分别是：
 - 标记任务完成
 - 标记任务未完成
 - 任务初始化
- 其中，第三个附加功能——任务初始化，有使用参数标记，并且其第一个参数的默认值用“?”标记
- PC 端执行的显示效果如下图

数据模型

上线任务跟踪

查询(S) 模型(E)

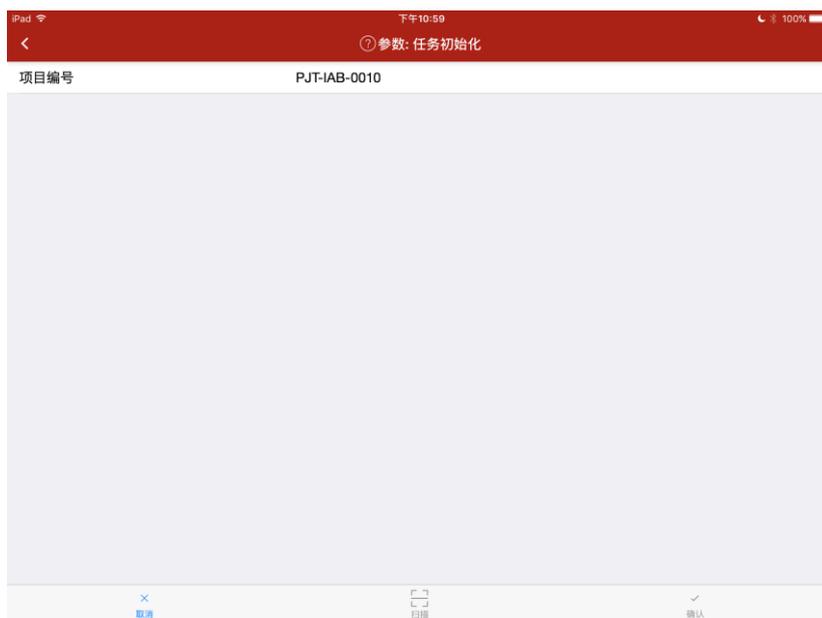
将列标题拉到此处实现分组

顺序号	任务	内容	责任人	需模拟	模拟日期
10	Pre-Cutover Week	任务		<input type="checkbox"/>	
20	Test Inventory C	测试通过库存盘点进入库存。		<input checked="" type="checkbox"/>	
30	Create all users	创建用户		<input type="checkbox"/>	
40	Finalize user me	最后确定用户菜单		<input type="checkbox"/>	
50	Run tests on Pro	正式账套测试		<input checked="" type="checkbox"/>	
60	Test all reports	测试所有报表, 确保Go-live 用到的报表可用	Customer	<input checked="" type="checkbox"/>	
70	Verify that all	确认所有交易生成对应正确的财务凭证	Customer/Sage	<input checked="" type="checkbox"/>	
80	Test Printing fr	测试所有客户端打印功能	Customer	<input checked="" type="checkbox"/>	
90	Test X3 access f	测试确保所有客户访问系统	Customer	<input type="checkbox"/>	
100	Close as many wo	尽可能关闭更多工单	Customer	<input type="checkbox"/>	
110	Close as many su	尽可能关闭更多外协订单	Customer	<input type="checkbox"/>	
120	Close as many es	尽可能关闭更多销售订单	Customer	<input type="checkbox"/>	

- 移动端执行的时候, “长按” 菜单显示的功能如下图



- 这时, 系统在执行相应的附加功能的时候, 会弹出界面, 允许用户确认已经填入的参数, 在确认后可以继续操作执行相应的功能



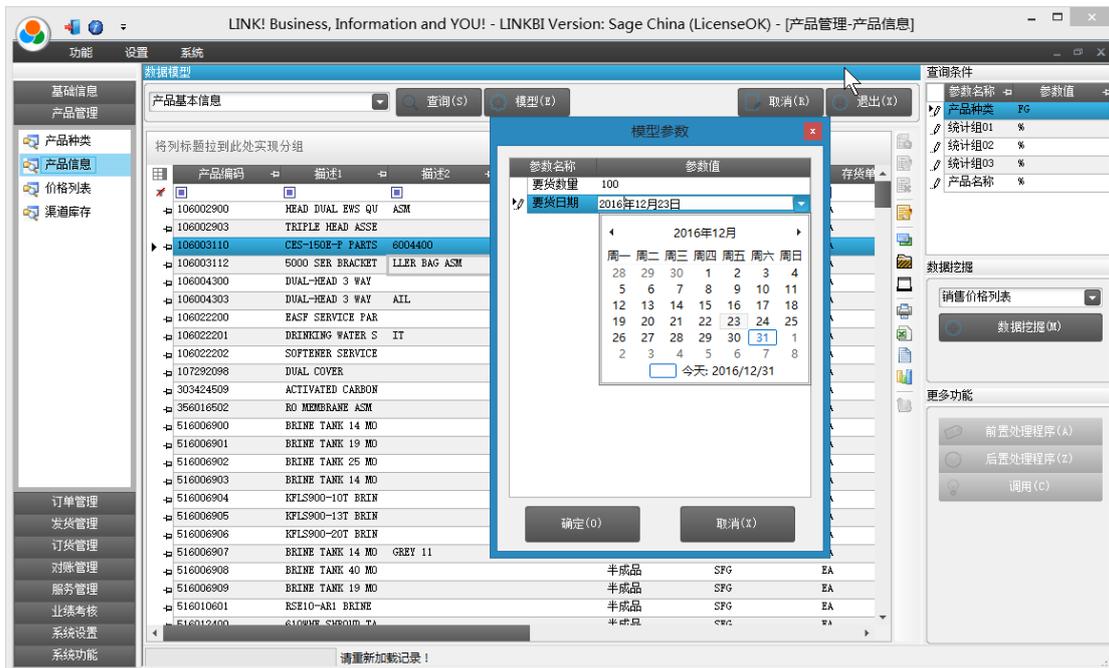
- 说明：
 - 对于没有“?”标记的参数，系统仍然会将父模型的对应参数值传递填入，只是不需要用户再次确认。

23 附件功能操作的额外参数定义

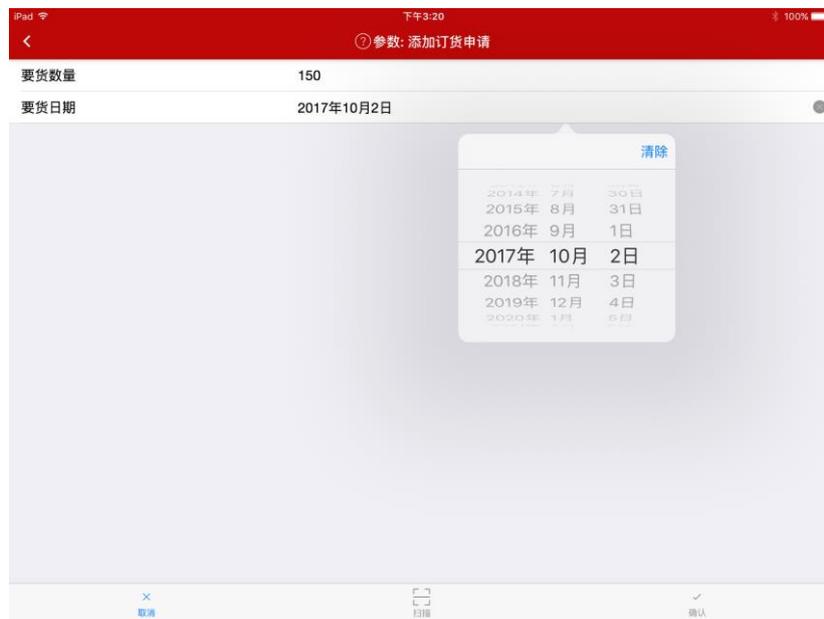


- 附件功能中, 如果需要使用参数, 与主模型的参数定义模式相同
 - 说明
 - ◆ 如果附加模型的参数与主模型的参数完全一致, 也可以保留参数空白
 - ◆ 此时, 子模型自动获取主模型的参数
 - 对于需要在主模型传递的参数之外, 增加用户输入数据的情况, 可以通过定义主模型之外的额外参数的形式实现
- 额外参数定义
 - 本例中, 在用于选择订货操作时, 需要传递两个额外的数据
 - ◆ 要货数量
 - ◆ 要货日期
 - 需要用户输入的参数数据, 在参数默认值中用“?”表示即可

- 该模型在 PC 客户端执行时，显示的效果如下图



- 该模型在移动端执行的时候，显示的效果如下



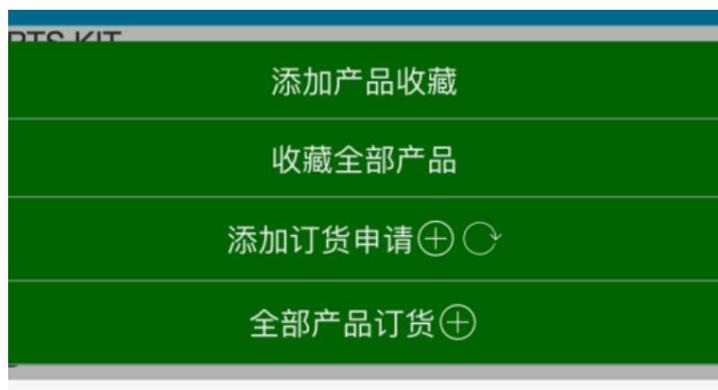
24 附加功能处理后的数据刷新

- 在附加模型的脚本中，可以通过指定关键字的方式，触发功能操作完成后的数据刷新
- 数据刷新关键字为--LINKREFRESH，可使用该关键字进行模型跳转，具体如下

--LINKREFRESH:模型代码 #顺序号 (#引用字段 1#,#引用字段 1#{0},{1})



- 如图所示，如果脚本的第一行为--LINKREFRESH，则执行完数据操作的脚本之后，会刷新主模型的数据
- 在移动端，该模型标题中会包含一个类似于“带箭头的圆圈”的数据刷新符号



- 在 PC 客户端，显示为标题前的一个“√”

106004300	DUAL-HEAD 3 WAY	
106004303	DUAL-HEAD 3 WAY	AIL
106022200	EASF SER	
106022201	DRINKING	
106022202	SOFTENER	
107292098	DUAL COV	<input checked="" type="checkbox"/>
303424509	ACTIVATE	
356016502	RO MEMBRANE ASM	
516006900	BRINE TANK 14 MD	

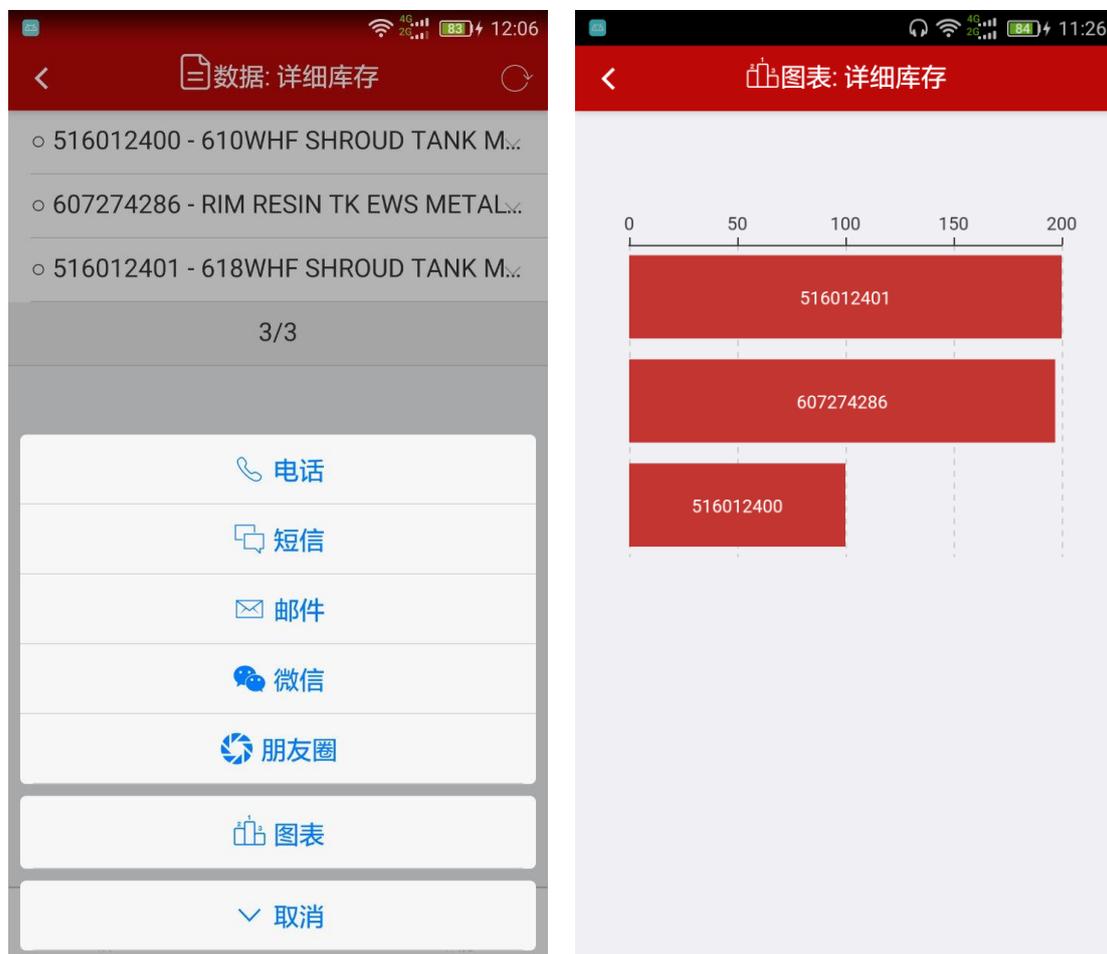
添加产品收藏

收藏全部产品

添加订货申请

全部产品订货

25 图表显示



- 通过定义模型的图表脚本，可以在移动端显示丰富的图表
- 图表的实现通过模型定义，TotalLINK 系统提供了强大的嵌入 javascript 脚本功能
- 在模型定义中，按照 javascript 的语法定义显示的基本模式
- 可以通过类似##COLNAME##的方式嵌入动态的数据
- 详细内容请参考后文的[图表模型设计](#)部分
- 图片中的数据分享（电话、短信、邮件、微信、朋友圈）不在新版本中开放

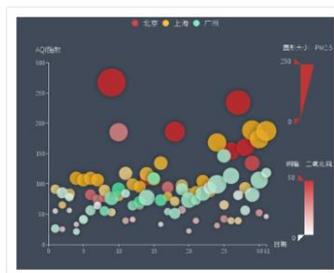
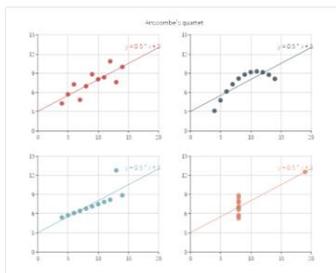
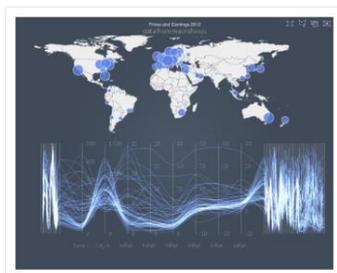
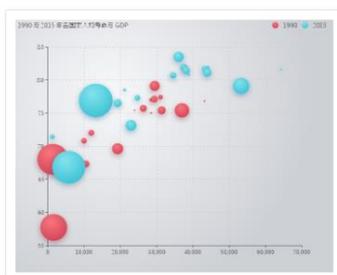
26 百度 ECHARTS3 集成

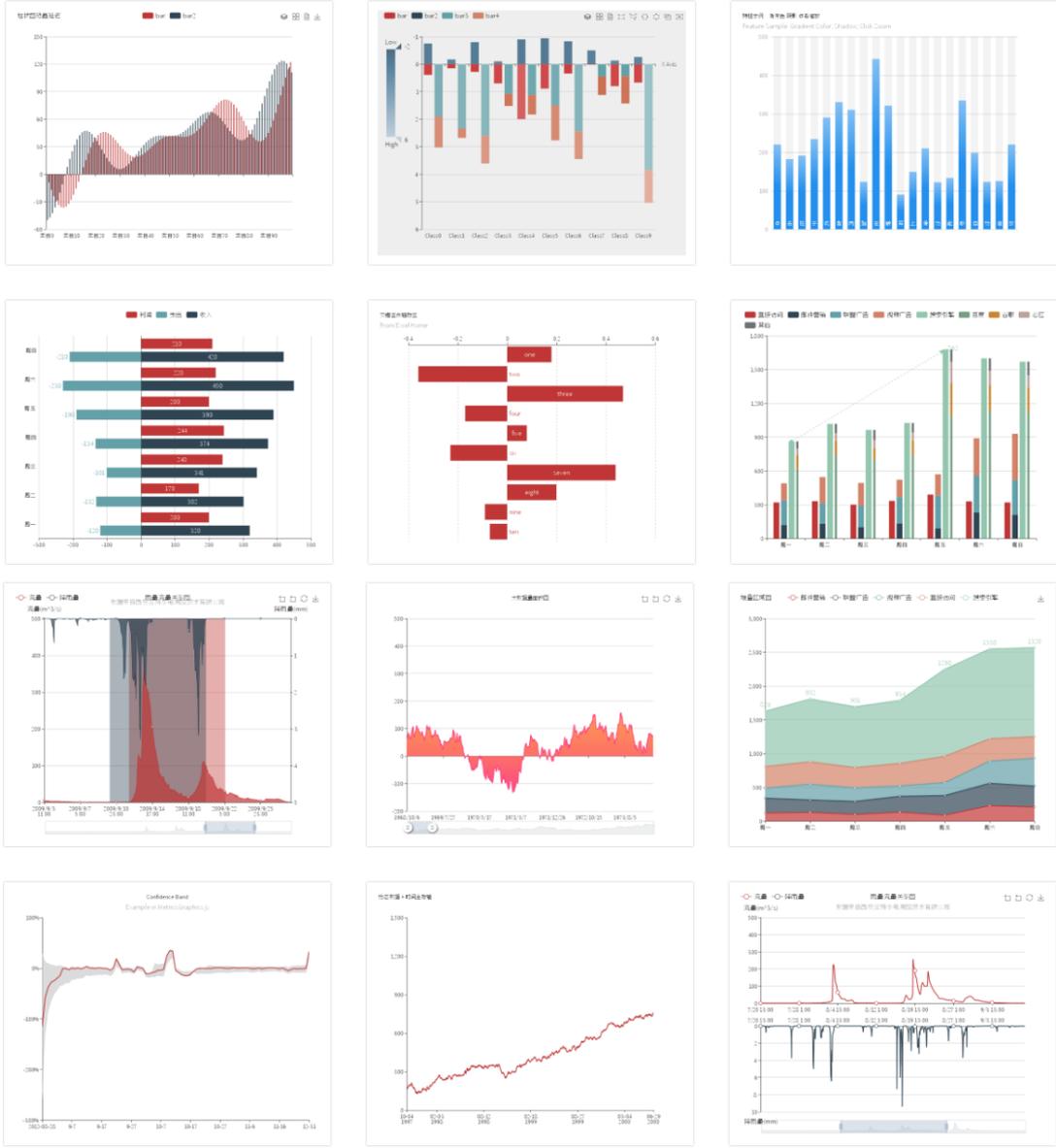
更丰富的可视化效果

新增更多图表类型，更好的满足不同数据的处理需求 更多的搭配方案让你的数据呈现方式更个性和完美

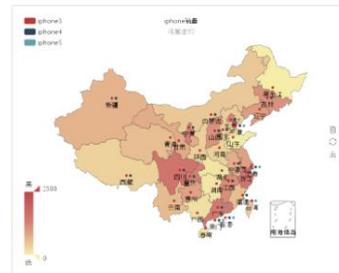
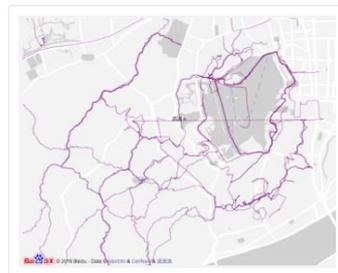
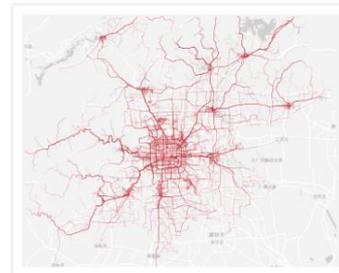
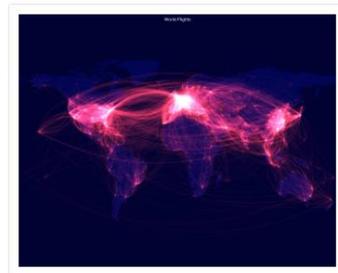
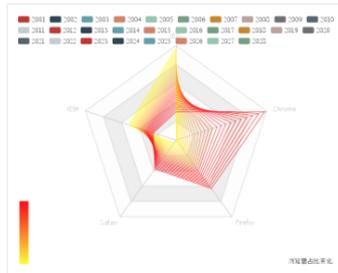
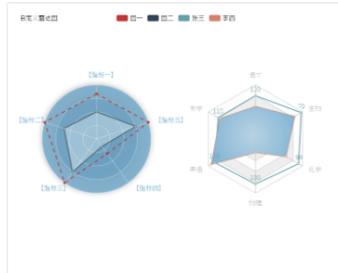
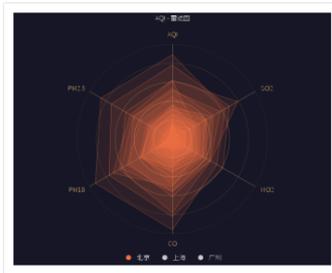


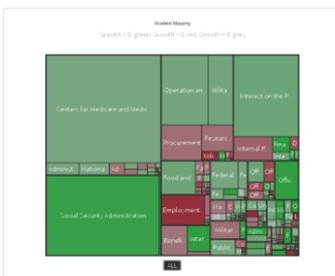
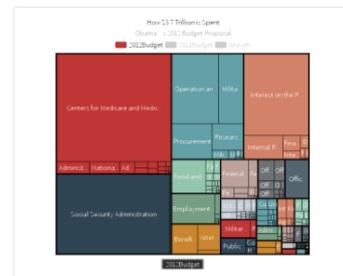
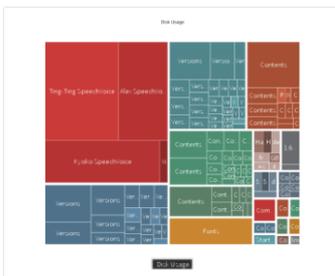
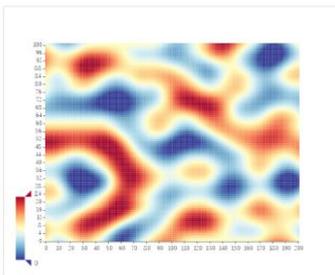
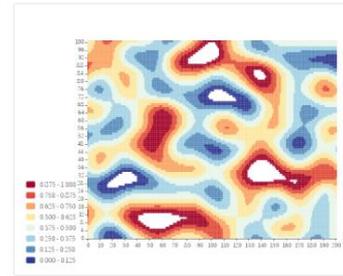
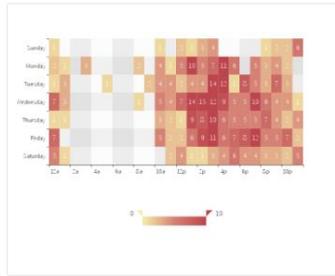
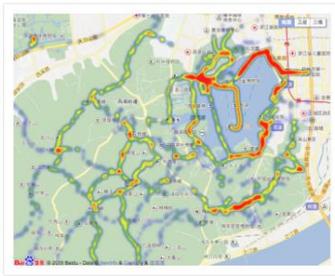
- 系统集成百度 eCharts3，可以实现各种丰富的图表

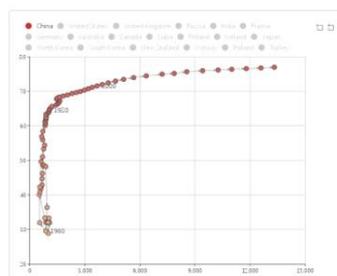
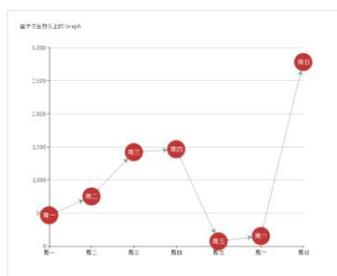
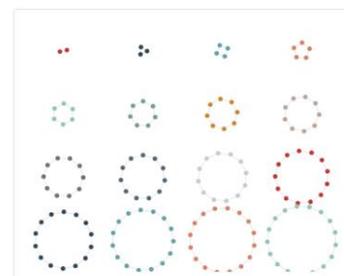
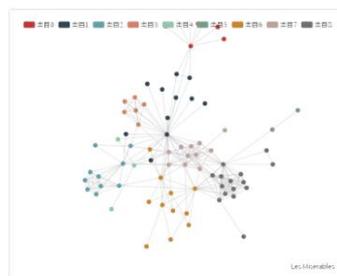
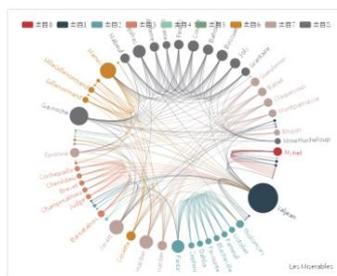
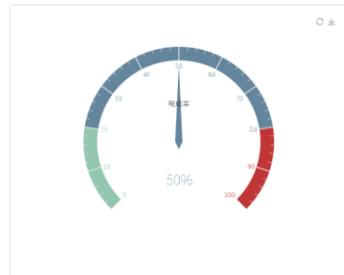
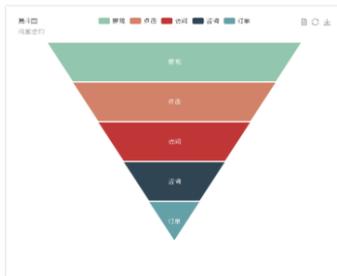
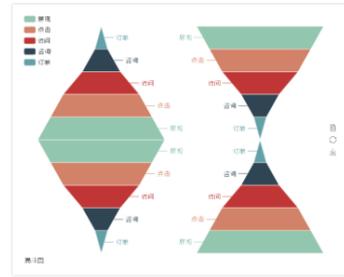
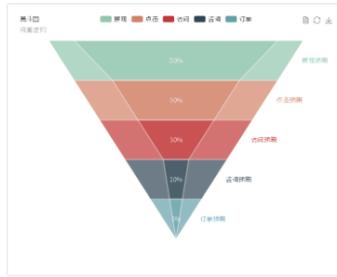
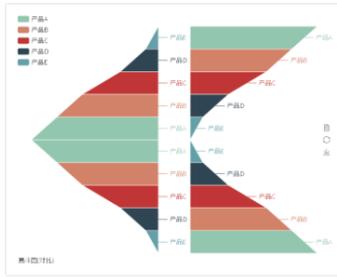












27 图表模型设计



- 移动端图表展示，是通过对模型定义图表展示脚本实现的。
- 如图所示，该模型用于查询库存数据
- 可以利用产品编号、存货数量生成条形图的图表
- 模型设计功能如上图所示
- 图表模型脚本（JS）设计使用嵌入 Javascript 脚本实现，本例中，模型脚本如下：

```

var labelRight = {{
    normal: {{
        position: 'right'
    }}
}};
option = {{
    tooltip: {{
        trigger: 'axis',
        axisPointer: {{ // 坐标轴指示器，坐标轴触发有效
            type: 'shadow' // 默认为直线，可选为：'line' | 'shadow'
        }}
    }}
}}
    
```

```
    }},
    grid: {{
      top: 80,
      bottom: 30
    }},
    xAxis: {{
      //type: 'value',
      position: 'top',
      splitLine: {{ lineStyle: {{ type: 'dashed' }} }},
    }},
    yAxis: {{
      //type: 'category',
      axisLine: {{ show: false }},
      axisLabel: {{ show: false }},
      axisTick: {{ show: false }},
      splitLine: {{ show: false }},
      data: [##产品编号##]
    }},
    series: [{{
      //name:'生活费',
      type: 'bar',
      //stack: '总量',
      label: {{
        normal: {{
          show: true,
          formatter: '{{b}}'
        }}
      }}
    }},
    data: [##存货数量##]
  }}
];
```

`myChart.setOption(option);`

- 本例中，图表所涉及到的数据分别来自数据列：“产品编号”、“存货数量”
- 在进行脚本设计时，通过类似“##数据列名##”的方式获取动态数据
 - 根据数据类型的不同，可以有三种模式
 - ◆ 带双引号的数据：“##数据列名##”
 - 使用实际数据替换后，形成的结果类似于：
“516012400”, “607274286”, “516012401”
 - ◆ 带单引号的数据：'##数据列名##'
 - 使用实际数据替换后，形成的结果类似于：

```
'516012400','607274286','516012401'
```

- ◆ 不带引号，单纯数字格式的数据：##数据列名##
 - 使用实际数据替换后，形成的结果类似于：
100.00,197.00,200.00

- 图表模型脚本的最后一句脚本，通常为：`myChart.setOption(option);`用于使用 `option` 的数据形成移动端展示的结果
- 本例，通过移动端发送图表请求，形成 JS 脚本的结果如下：

```
var labelRight = {
  normal: {
    position: 'right'
  }
};
option = {
  tooltip: {
    trigger: 'axis',
    axisPointer: { // 坐标轴指示器，坐标轴触发有效
      type: 'shadow' // 默认为直线，可选为：'line' | 'shadow'
    }
  },
  grid: {
    top: 80,
    bottom: 30
  },
  xAxis: {
    //type: 'value',
    position: 'top',
    splitLine: { lineStyle: { type: 'dashed' } },
  },
  yAxis: {
    //type: 'category',
    axisLine: { show: false },
    axisLabel: { show: false },
    axisTick: { show: false },
    splitLine: { show: false },
    data: ['516012400','607274286','516012401']
  },
  series: [{
    //name:'生活费',
    type: 'bar',
    //stack: '总量',
    label: {
```

```
        normal: {
            show: true,
            formatter: '{b}'
        }
    },
    data: [100.00,197.00,200.00]
}
};
myChart.setOption(option);
```

- 脚本的转义处理
 - 脚本中的“{”、“}”符号与 dotnet 平台的参数替换功能的符号标记“{0}”、“{1}”等会出现混淆，需要统一使用花括号的转义模式，用“{{”代替“{”，用“}}”代替“}”
 - 当脚本中需要使用到参数“{0}”、“{1}”等作为图表的“标题、副标题”等数据时，需要使用上述“转移字符替换”；如果脚本中不使用“{0}”、“{1}”等参数，可以不需要进行转移替换

28 地图数据示例

- 根据需要，可以将销售数据等与地理位置有关的信息在地图上显示



- 如图所示，显示的是全国各地城市的空气质量 PM2.5 的数据。
- 其实现的模型设计如下：

- 基本模型，用于获取各地的 PM2.5 数据及城市的坐标等

```
SELECT 1 AS 编号, N'海门' AS 城市, N'{{name: "海门", value: 9}}' AS 空气质量,
N'"海门":[121.15,31.89]' AS 坐标
UNION
SELECT 2, N'鄂尔多斯', N'{{name: "鄂尔多斯", value: 12}}', N'"鄂尔多斯":[109.781327,39.608266]'
UNION
SELECT 3, N'招远', N'{{name: "招远", value: 12}}', N'"招远":[120.38,37.35]'
UNION
SELECT 4, N'舟山', N'{{name: "舟山", value: 12}}', N'"舟山":[122.207216,29.985295]'
UNION
SELECT 5, N'齐齐哈尔', N'{{name: "齐齐哈尔", value: 14}}', N'"齐齐哈尔":[123.97,47.33]'
UNION
SELECT 6, N'盐城', N'{{name: "盐城", value: 15}}', N'"盐城":[120.13,33.38]'
UNION
SELECT 7, N'赤峰', N'{{name: "赤峰", value: 16}}', N'"赤峰":[118.87,42.28]'
```

```

UNION
SELECT 8,N'青岛',N'{{name: "青岛", value: 18}}',N'"青岛":[120.33,36.07]'
UNION
SELECT 9,N'乳山',N'{{name: "乳山", value: 18}}',N'"乳山":[121.52,36.89]'
UNION
SELECT 10,N'金昌',N'{{name: "金昌", value: 19}}',N'"金昌
":[102.188043,38.520089]'
UNION
SELECT 11,N'泉州',N'{{name: "泉州", value: 21}}',N'"泉州":[118.58,24.93]'
UNION
SELECT 12,N'莱西',N'{{name: "莱西", value: 21}}',N'"莱西":[120.53,36.86]'
UNION
SELECT 13,N'日照',N'{{name: "日照", value: 21}}',N'"日照":[119.46,35.42]'
UNION
SELECT 14,N'胶南',N'{{name: "胶南", value: 22}}',N'"胶南":[119.97,35.88]'
UNION
SELECT 15,N'南通',N'{{name: "南通", value: 23}}',N'"南通":[121.05,32.08]'
UNION
SELECT 16,N'拉萨',N'{{name: "拉萨", value: 24}}',N'"拉萨":[91.11,29.97]'
UNION
SELECT 17,N'云浮',N'{{name: "云浮", value: 24}}',N'"云浮":[112.02,22.93]'

```

- 图片模型，用于动态获取数据，并设置图表的格式

```

var data = [##空气质量##];
var geoCoordMap = {{##坐标##}};

var convertData = function (data) {{
  var res = [];
  for (var i = 0; i < data.length; i++) {{
    var geoCoord = geoCoordMap[data[i].name];
    if (geoCoord) {{
      res.push({{
        name: data[i].name,
        value: geoCoord.concat(data[i].value)
      }});
    }}
  }}
  return res;
}};

option = {{
  backgroundColor: '#404a59',

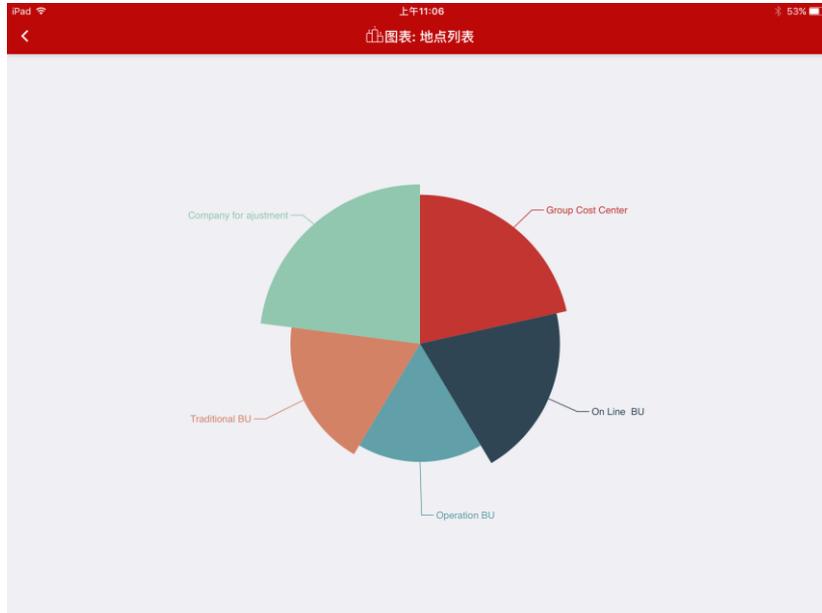
```

```
title: {{
  text: '全国主要城市空气质量',
  subtext: 'data from PM25.in',
  sublink: 'http://www.pm25.in',
  left: 'center',
  textStyle: {{
    color: '#fff'
  }}
}},
tooltip : {{
  trigger: 'item'
}},
legend: {{
  orient: 'vertical',
  y: 'bottom',
  x: 'right',
  data: ['pm2.5'],
  textStyle: {{
    color: '#fff'
  }}
}},
geo: {{
  map: 'china',
  label: {{
    emphasis: {{
      show: false
    }}
  }},
  roam: true,
  itemStyle: {{
    normal: {{
      areaColor: '#323c48',
      borderColor: '#111'
    }},
    emphasis: {{
      areaColor: '#2a333d'
    }}
  }}
}},
series : [
  {{
    name: 'pm2.5',
```

```
type: 'scatter',
coordinateSystem: 'geo',
data: convertData(data),
symbolSize: function (val) {{
    return val[2] / 10;
}},
label: {{
    normal: {{
        formatter: '{{b}}',
        position: 'right',
        show: false
    }},
    emphasis: {{
        show: true
    }}
}},
itemStyle: {{
    normal: {{
        color: '#ddb926'
    }}
}}
}},
{{
    name: 'Top 5',
    type: 'effectScatter',
    coordinateSystem: 'geo',
    data: convertData(data.sort(function (a, b) {{
        return b.value - a.value;
    }}).slice(0, 6)),
    symbolSize: function (val) {{
        return val[2] / 10;
    }},
    showEffectOn: 'render',
    rippleEffect: {{
        brushType: 'stroke'
    }},
    hoverAnimation: true,
    label: {{
        normal: {{
            formatter: '{{b}}',
            position: 'right',
            show: true
        }}
    }}
}}
```

```
    }  
  },  
  itemStyle: {  
    normal: {  
      color: '#f4e925',  
      shadowBlur: 10,  
      shadowColor: '#333'  
    }  
  },  
  zlevel: 1  
}  
]  
};  
myChart.setOption(option);
```

29 组合图表数据示例



- 上图所示图表的 js 脚本如下:

```
option = {{
  series: [{
    //name: '访问来源',
    type: 'pie',
    radius: '55%',
    roseType: 'angle',
    data: [##CHARTDATA##]
  }]
}};
myChart.setOption(option);
```

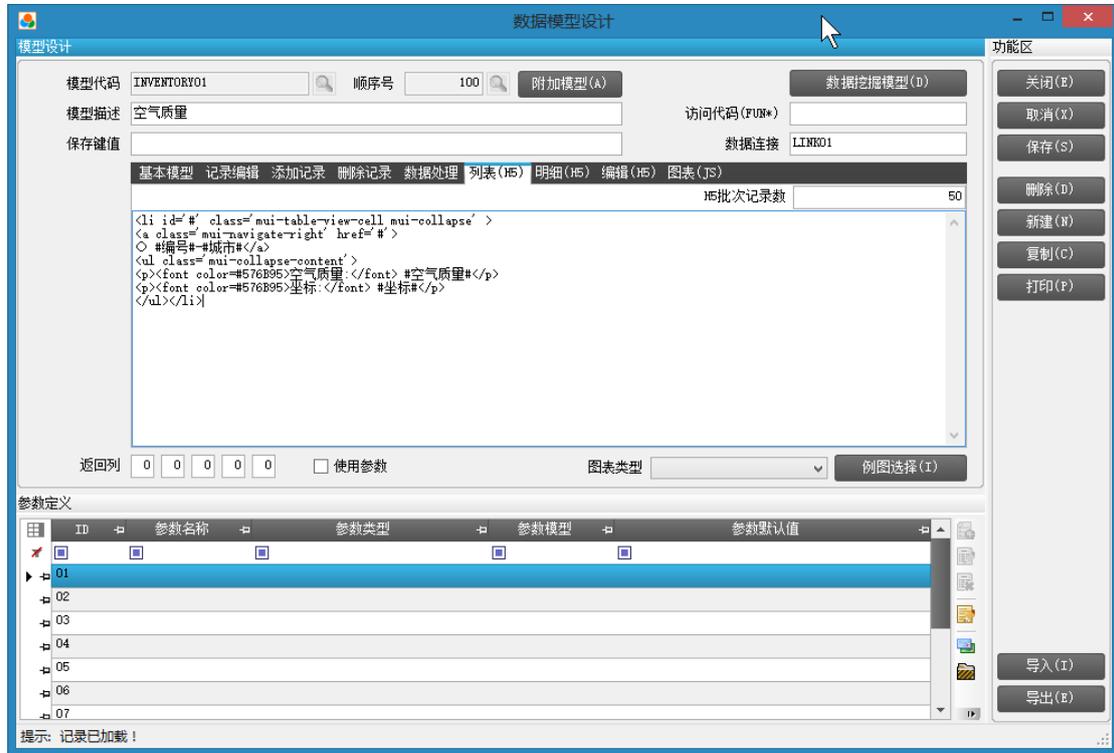
- 其中的数据，可以通过模型设计通过一个 CHARTDATA 列获取
- 模型的基本数据获取脚本如下:

SELECT FCYNAM_0	名称,
FCY_0	LINKSUB10_地点,
CRY_0	国家,
MFGFLG_0 - 1	AS LINKBOOLEAN 生产,
SALFLG_0 - 1	AS LINKBOOLEAN 销售,
PURFLG_0 - 1	AS LINKBOOLEAN 采购,
WRHFLG_0 - 1	AS LINKBOOLEAN 存货,
FINFLG_0 - 1	AS LINKBOOLEAN 财务,

```
FINRSPFCY_0 AS 对应财务地点,  
LEGCPY_0 所属公司,  
BIDNUM_0 LINKROWIDCOLOR,  
N'{{value:1' + Cast(ROWID AS VARCHAR(10))  
+ N'5,name:'' + FCYNAM_0 + ''}}' AS CHARTDATA  
FROM FACILITY  
ORDER BY FCY_0
```

- 该例中，用于图表的数据是通过各列数据信息组合而成的，可根据项目的实际情况进行处理

30 多批次数据加载



- 在模型设计的时候，指定批次记录数，可以用来控制每批次显示的数据条数
- 比如本例中，设定批次记录数为 50，在移动端获取数据的时候，每批次加载 50 条记录

- 第一次加载记录的时候，自动加载一个批次的数据



- 数据的最后一行，显示当前已经加载的记录数和总记录数
- 没有完整加载的情况下，点击最后一行会自动加载后面一个批次的数据



- 直到所有的数据都加载后，最后一条显示为灰色



- 在任何时候，系统共享功能中的“图表”显示的数据与当前已经加载的数据一致

31 数据刷新



- 正如在前面例子中看到的，在数据栏的右上角有个“数据刷新的按钮”
- 点击此按钮可以重新从第一批加载数据

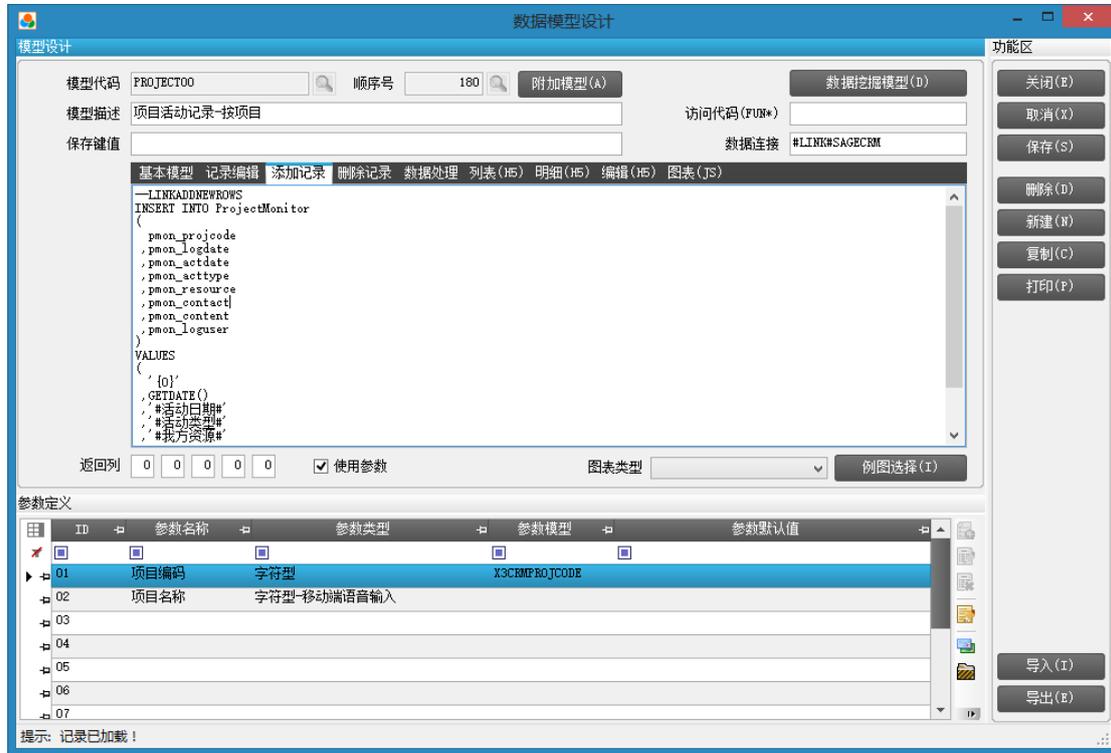
32 数据处理操作

- 与数据模型的添加记录操作对应，也可以在移动端执行数据的添加操作



- 在屏幕的右下角可以执行“添加、编辑、删除”的操作

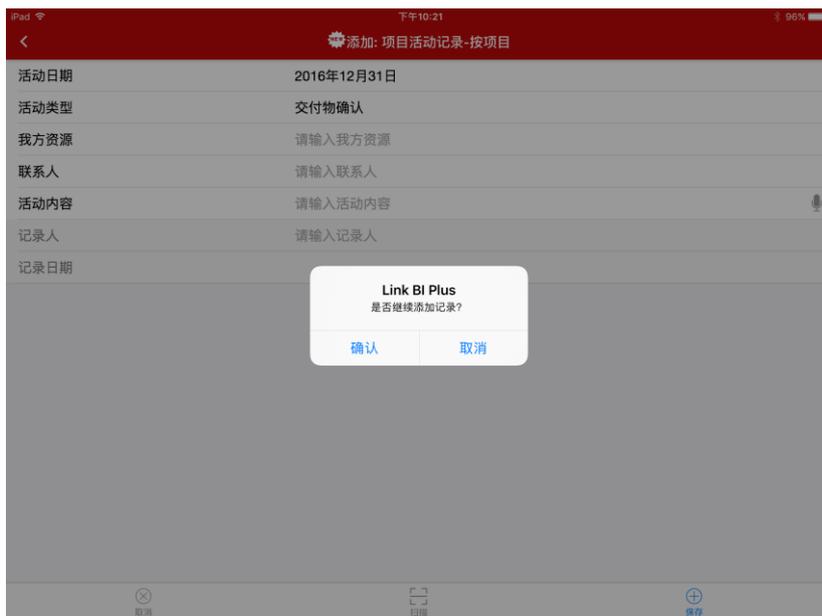
33 数据处理-添加



- 该模型定义了数据的添加操作
- 执行“添加操作”时移动端显示的界面如图所示



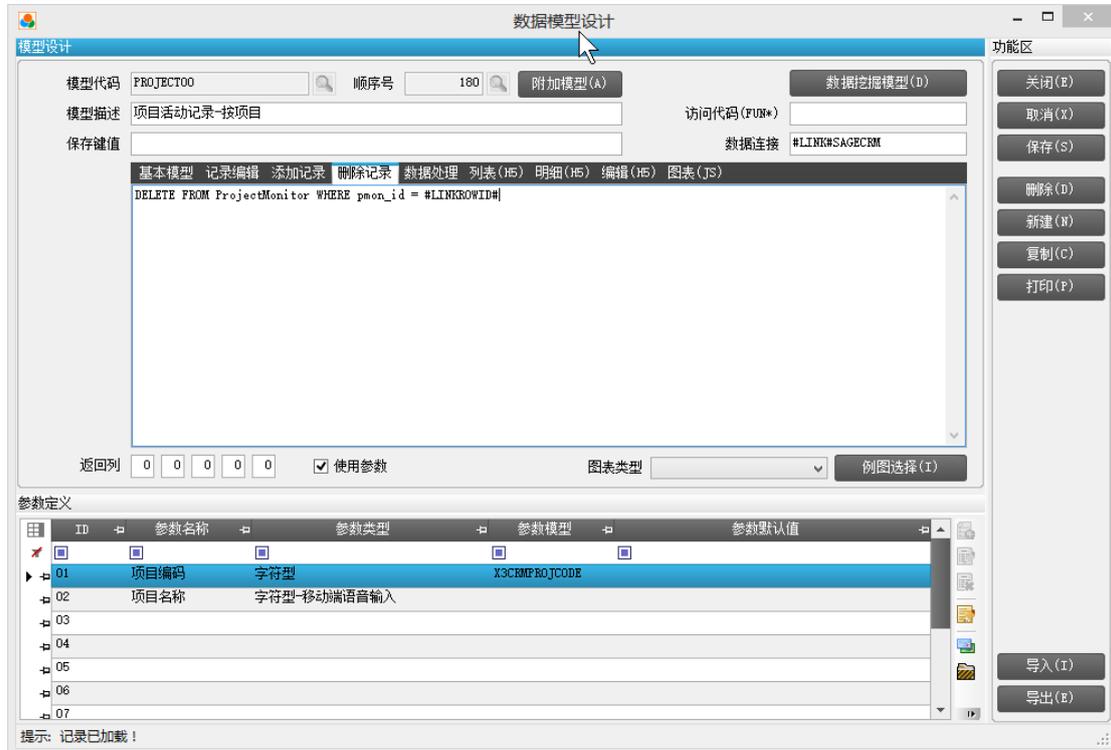
- 如果需要编辑的数据列定义了选择模型，在移动端也会自动激活列表选择功能



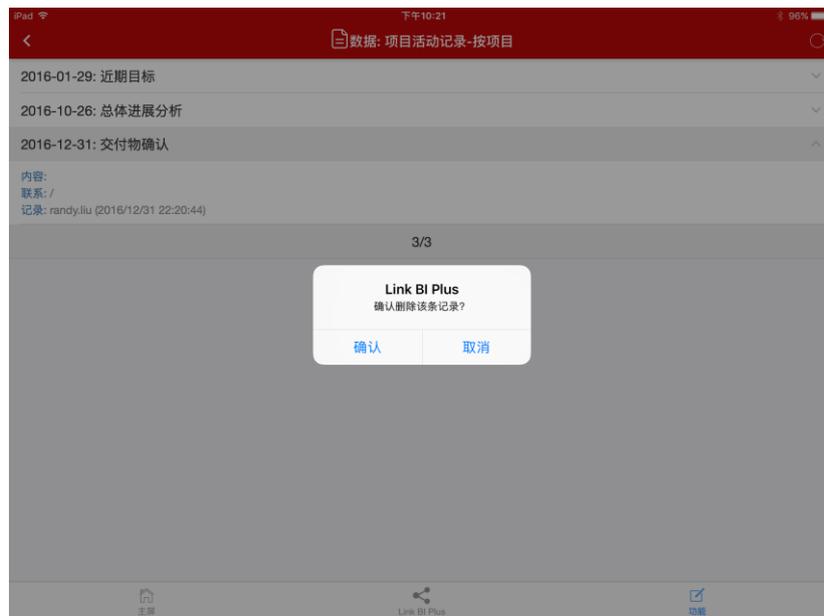
- 完成数据填写后，可以点击右下角的“保存”按钮，执行添加保存操作
- 数据保存完成后，系统会提示是否继续添加记录，用户根据需要选择“确认”或“取消”
- 完成数据添加后，返回数据界面，就可以看到添加的记录了



34 数据处理-删除

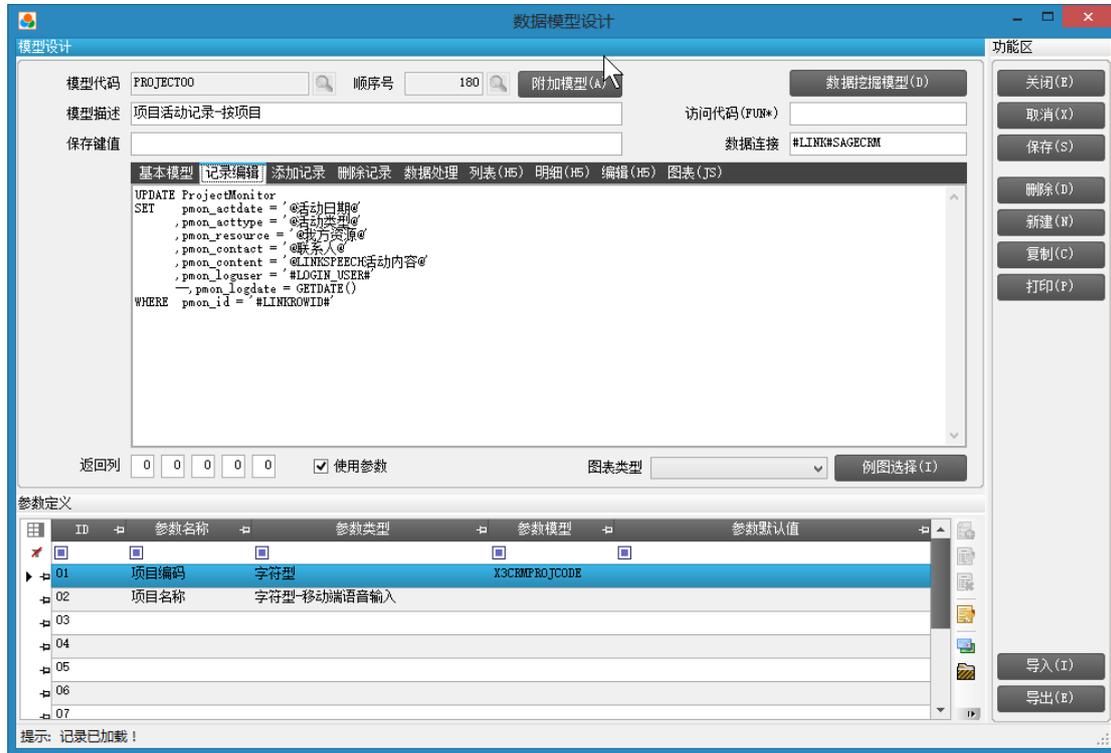


- 如图，该模型定义了数据的删除操作

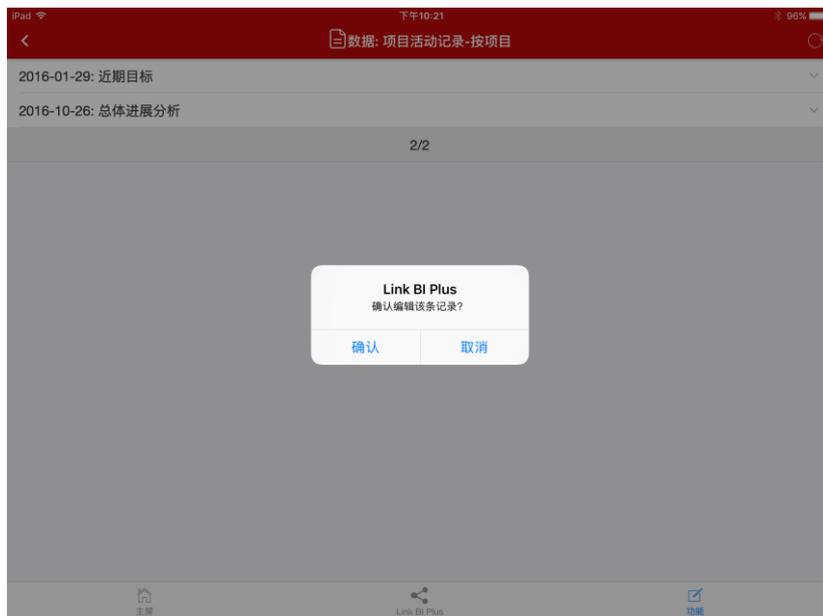


- 在移动端操作时，可以执行“删除”操作
- 根据系统提示完成操作即可

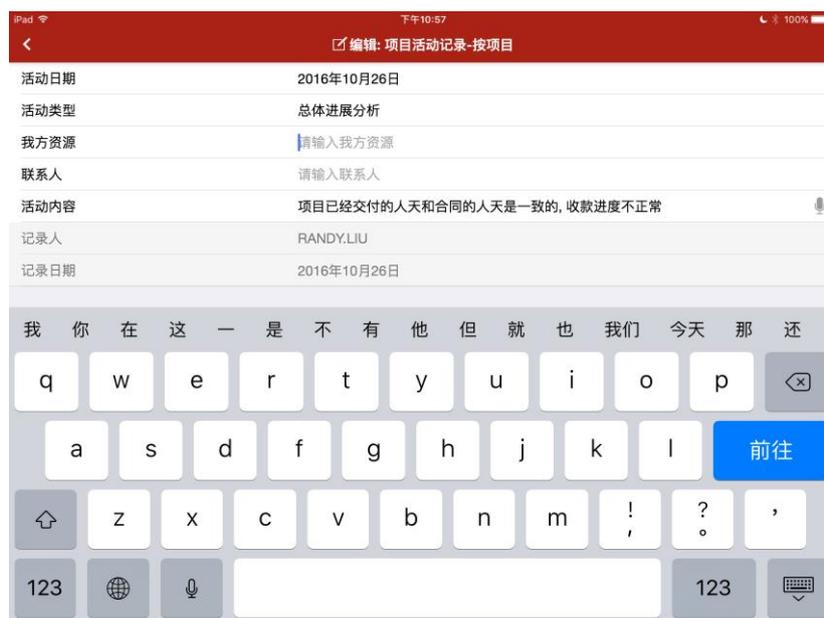
35 数据处理-编辑



- 该模型定义了数据的编辑操作



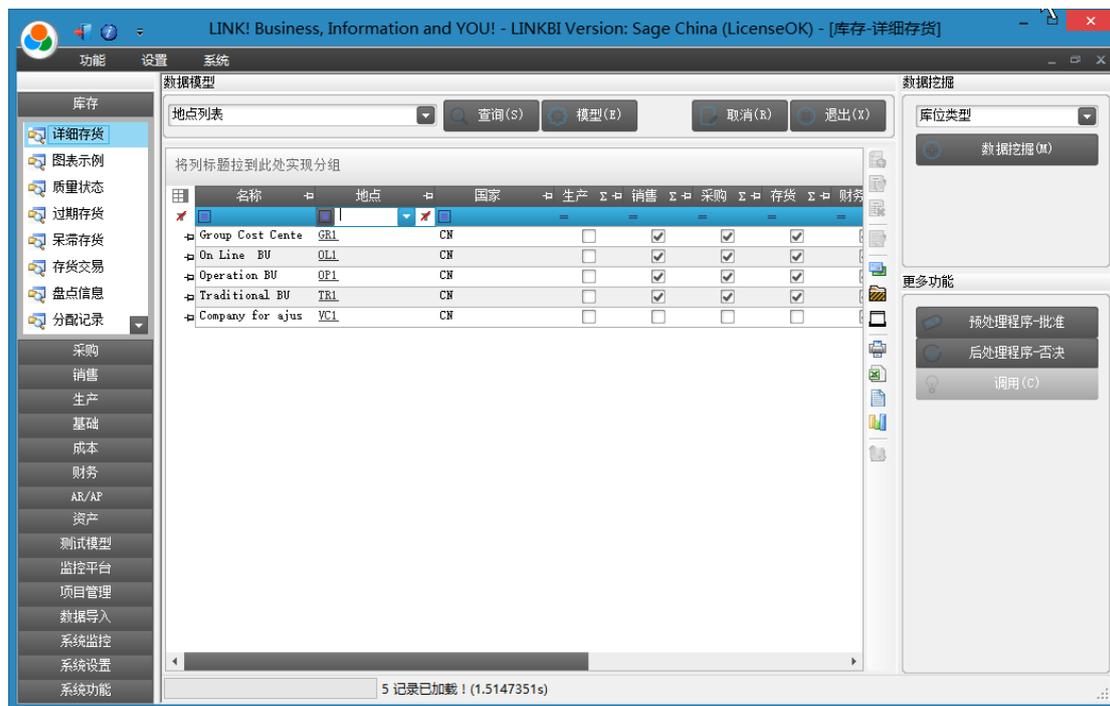
- 根据系统提示操作，系统会打开当前记录的编辑功能



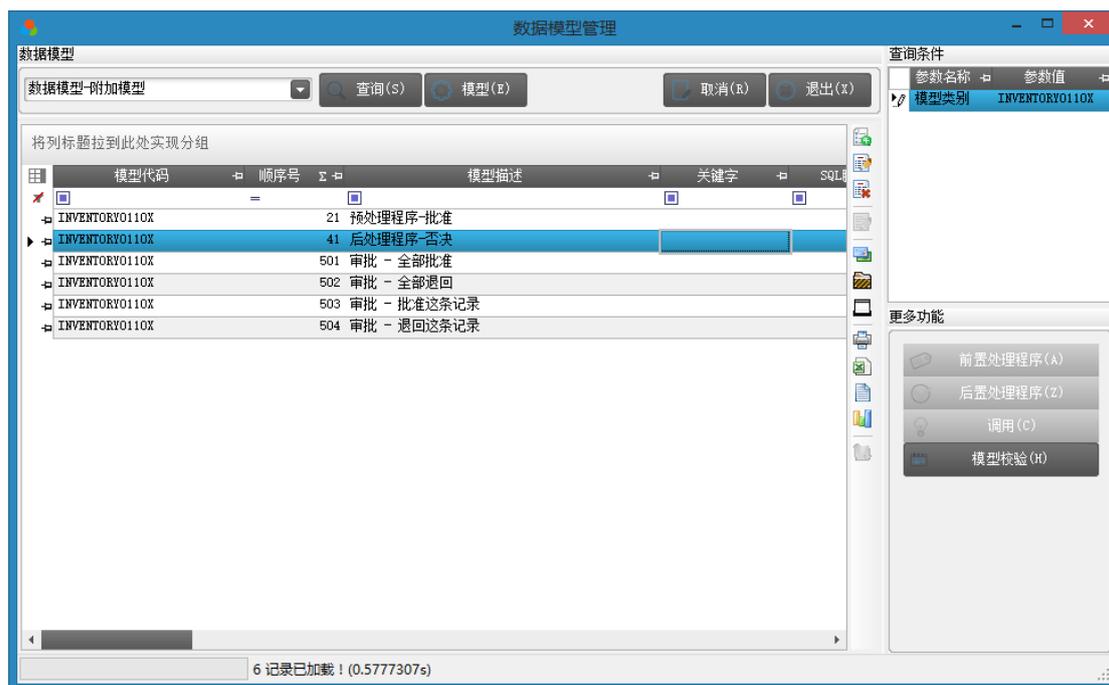
- 用户完成相应内容的编辑操作后，保存即可。

36 预处理及后处理功能

对于每个模型，都可以定义预处理及后处理功能，用于对数据集或者对数据的某一行记录进行操作。



- 如图所示，该模型定义了“预处理”和“后处理模型”，分别是：
 - 预处理程序-批准
 - 后处理程序-否决

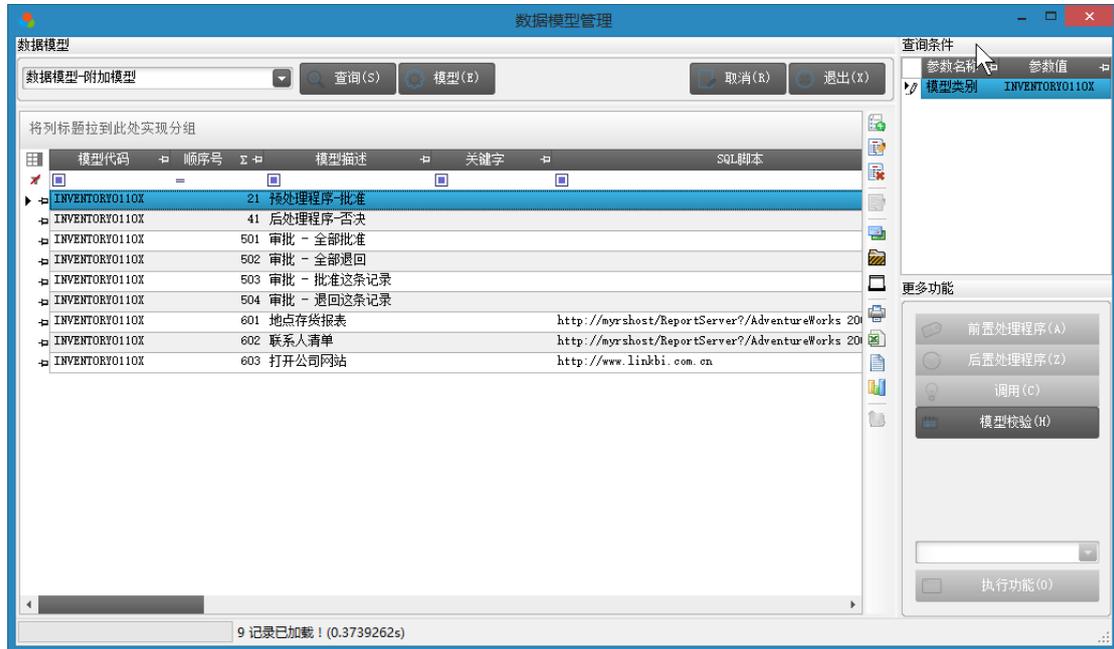


- 预处理及后处理功能的详细说明请参见模型设计的相关文档



- 在移动端执行时，功能显示如上图所示
- 点击操作相应的功能即可

37 外部功能列表



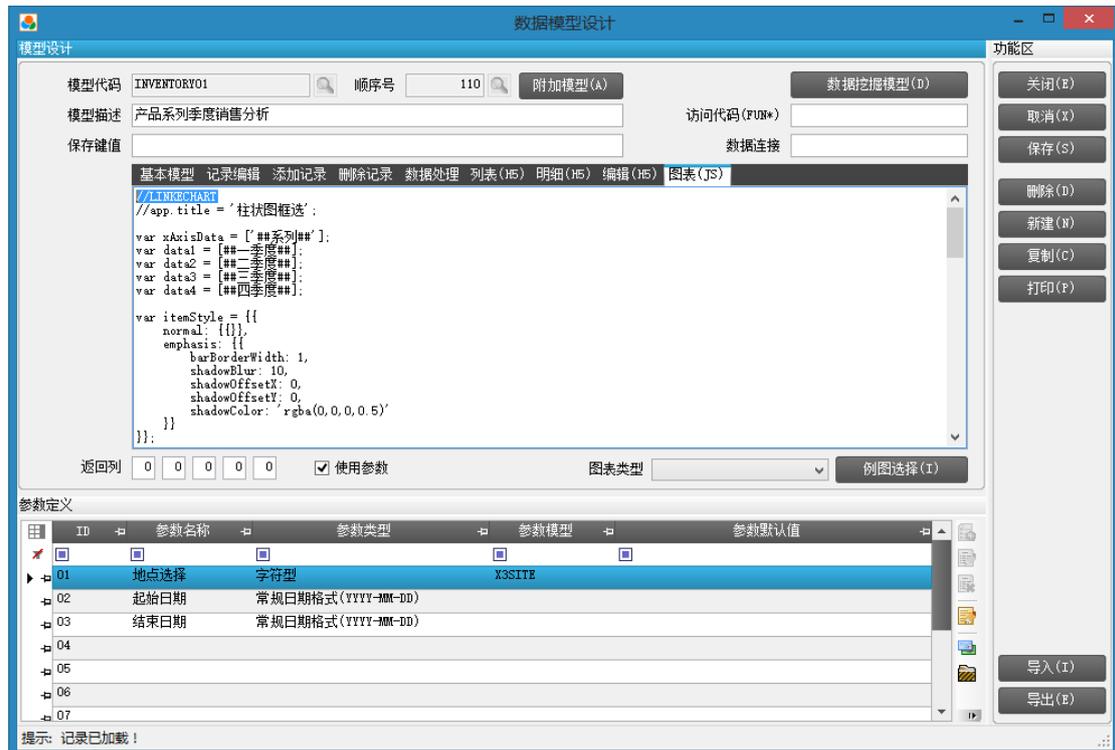
- 如图所示，定义的附加模型编号在 601 与 620 之间的时候，可以执行对应的外部功能，比如打开一个系统应用或者外部网址等
- 此功能可以用于与其他的应用集成，比如集成 Microsoft Reporting Services 应用等
- 在执行相应功能的时候，系统会自动将当前模型的主参数以及当前行记录的数据作为参数传递给相应的功能



- 移动端执行相应的功能的时候，界面显示如上图所示

38 直接展现图表

- 某些情况下，如果需要不通过加载数据转图表的方式，而直接展现图表时，可以通过在图表脚本中使用“//LINKECHART”标记实现

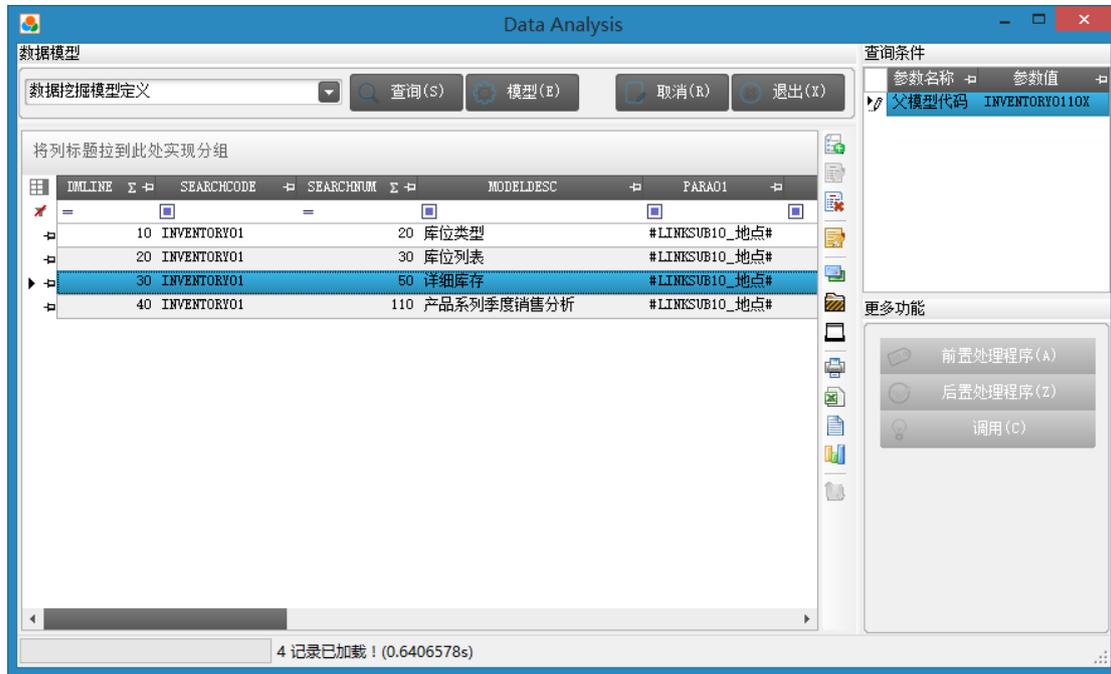


- 直接展现为图表的内容，在菜单列表中显示的时候，以不同的颜色显示菜单条



- 如图所示，“空气质量”和“产品系列销售分析”是直接显示为图表内容的
- 如果模型的图表数据加载时需要用户输入参数，仍然先打开参数输入界面
- 在参数输入完成后，查询功能即可直接加载为图表

39 数据挖掘直接显示图表

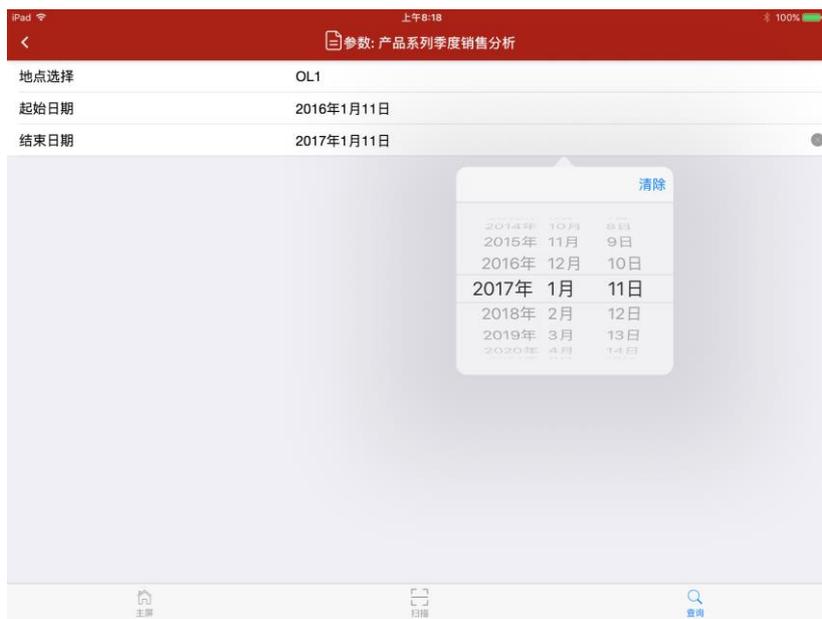


- 如图所示，数据挖掘模型中也可以指定直接显示图表的模型
 - 40-INVERTORY01-100，产品系列季度销售分析



- 直接显示图表的数据挖掘模型在移动端显示的时候，弹出菜单以不同的颜色区分
 - 请参考图中的“产品系列季度销售分析”
- 如果图表不需要参数，此时可以直接打开图表

- 对于有需要参数确认的“数据挖掘模型”，首先打开参数输入界面



- 根据需要输入参数后，执行查询操作，即可直接得到图表

