

原生SQL数据集

原生SQL数据集类似SQL数据集，是通过在文本输入区中直接输入各类数据库方言表达式定义数据集条件和内容的一种数据集。

一般在SQL数据集嵌套执行性能低或是SQL数据集不支持的数据集语句（如Syabse数据库子数据集不支持Order by），我们可以使用原生SQL数据集来实现。

原生SQL数据集与SQL数据集的主要区别如下：

- SQL数据集是一类封装结构的原生SQL数据集语句，在解析SQL语句过程中对最外层进行了包装：如“select area from area”解析后的执行语句为“select T84.area as F85 from (select area from area) T84”；而原生SQL数据集没有对最外层进行包装，是按照“所见即所得”的方式进行解析并执行。
- SQL数据集不支持“select * from”的语句；而原生SQL数据集支持该语句。

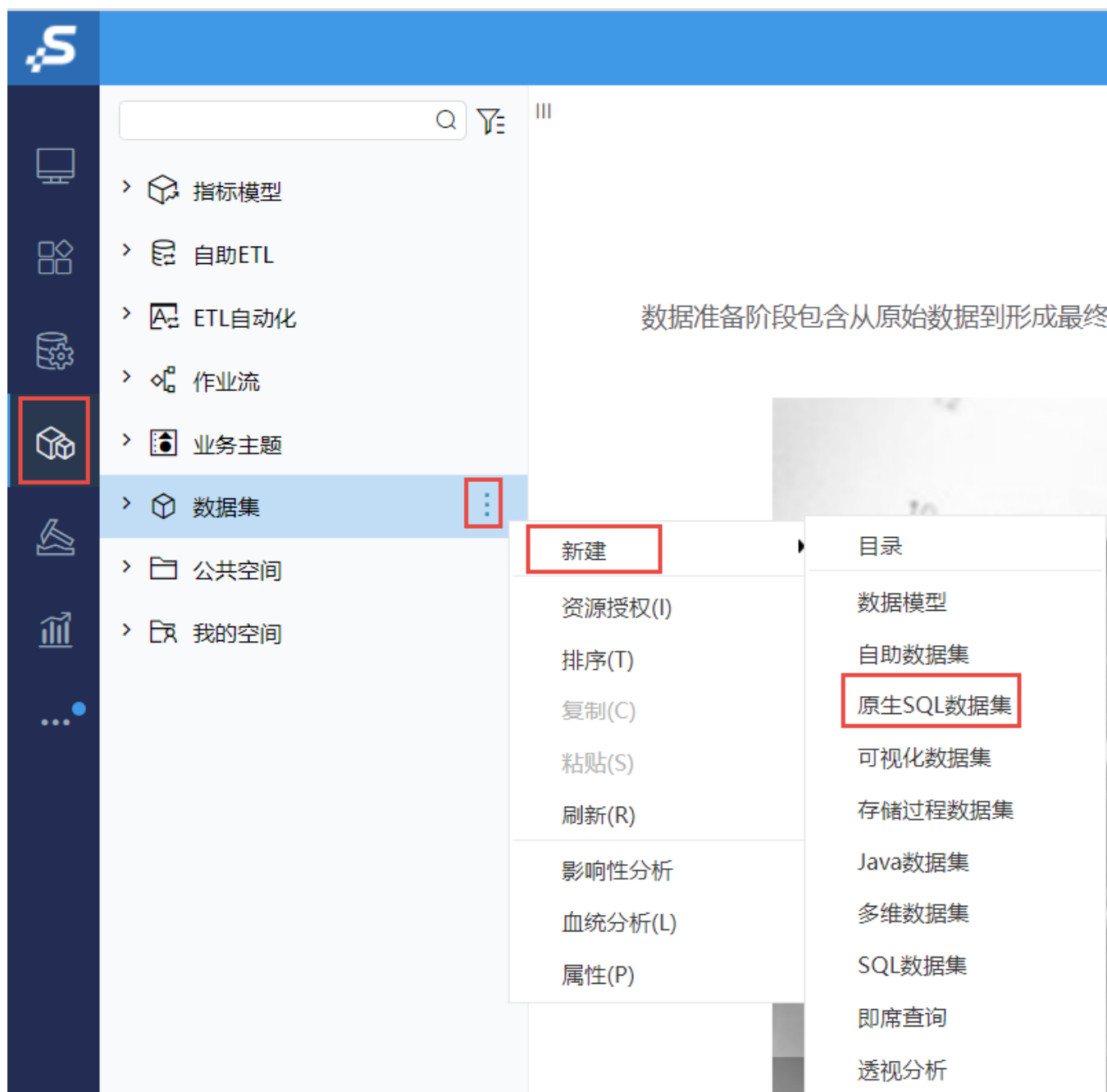
- 操作入口
- 功能说明
- 界面介绍
- 可选资源区
- SQL文本区
 - 工具栏
- 属性区
 - 输出字段
 - 检测输出字段
 - 其他说明：



关于原生SQL数据集的创建入口、资源显示和使用，统一通过V10.5 license控制。如希望在V10.5版本中继续使用原生SQL数据集，则需更新授权包含原生SQL数据集的license文件。

操作入口

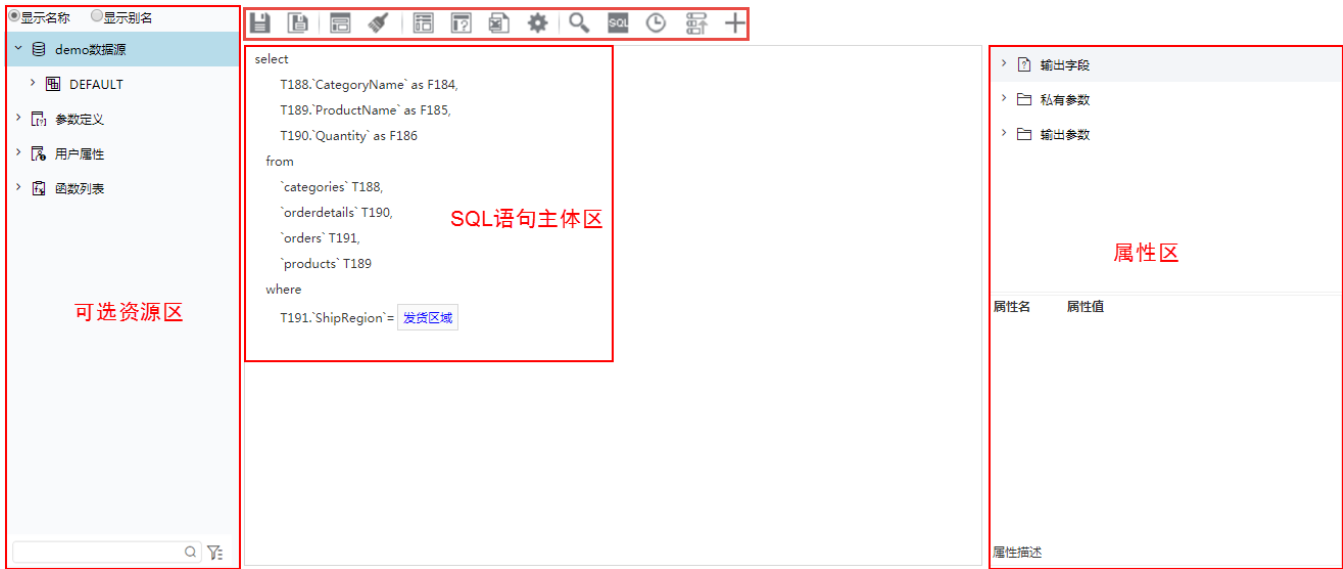
在系统导航栏中选择 **数据准备** ，在数据集的更多  中选择 **新建>原生SQL数据集** 。



功能说明

- 当原生SQL数据集中通过“select * from”语句返回结果集时，建议先通过“原生SQL数据集窗口 > 工作区 > 输出字段”的右键菜单选择 **检测输出字段** 命令检测出输出字段，再保存当前原生SQL数据集，确保在数据集定义的目录资源树上可以看到该数据集的输出结果。
- 原生SQL数据集中，在执行 **检测输出字段** 时，会从连接的数据库中读取输出字段的数据类型。
- 原生SQL数据集支持在同数据源下跨Schema进行查询，但是必须确保连接该数据源的用户对进行查询的Schema拥有相应的权限。

界面介绍



“原生SQL数据集”界面主要分为以下几个区域：

- 可选资源区：该区列出了所有可被原生SQL数据集使用的资源。
- SQL文本区：输入原生SQL数据集表达式的区域。
- 属性区：属性区用于设置数据集相关资源的属性。
- 工具栏：列出了所有可用于编辑原生SQL数据集的工具按钮。在“原生SQL数据集”界面中可以进行局部参数设置、参数缺省值设置。

可选资源区

可选资源区列出了所有可被原生SQL数据集使用的资源。用户只需将各资源拖到相应的区域，如将参数拖拽到where条件下和字段进行关联。

可选资源包含：参数、函数、用户属性。

显示别名/名称：该区针对“关系数据源”、“联合数据源”和“指标数据源”下的各资源：表、字段、视图、存储过程、表副本和视图副本，提供了“显示名称”和“显示别名”的功能。默认以“别名”显示。

模糊搜索：我们可以该区最下方的搜索引擎对相关资源快速搜索定位。可以在搜索区输入所需字段的真名或是别名进行模糊搜索。

SQL文本区

SQL文本区使用 [SQL语句编辑器](#) 输入SQL语句及其对象。

原生SQL数据集输入的注意事项：

- 在原生SQL数据集中编写SQL语句时，不能含有特殊字符，否则会报错。目前已知的特殊字符有“`”，单个“\”，以及非成对出现的“{[“，”]”。
- 原生SQL数据集可以使用的资源：参数、函数、用户属性（使用时请用括号括起来）
- 如果有多条语句，用分号和回车隔开；
- 表关系视图的定义对原生SQL数据集不起效；
- 数据行权限设置对原生SQL数据集不起效。
- 设置参数缺省值，在原生SQL数据集的设置方法为：

```
where {[ T32. CategoryName = CategoryName ]} and {[ T5.OrderDate=> 开始日期  
andT5.OrderDate<= 结束日期 ]};;
```

工具栏

工具栏上放置一些常用功能按钮。



工具栏的按钮说明如下表，对应的功能点击超链接跳转到详细分析：

按钮名称	说明
------	----

保存	保存数据集到默认路径。
另存	保存数据集到用户指定路径。
视图	用于区域的显示和隐藏。包含显示/隐藏资源区、显示/隐藏属性区。
格式化SQL	对用户编写的SQL语句按照一定的格式进行展现，便于阅读。
定义多重表头	对原生SQL数据集的表头进行修改添加，形成多重表头。
参数布局	调整参数在原生SQL数据集对应的报表中的位置。
Excel模板设置	把当前数据集的数据结果填充到用户自定义的Excel模板中。
高级设置	用于设置当前数据集的缓存策略和分页策略。
预览数据	对新建或修改的数据集进行预览。
查看SQL	查看该数据集的SQL语句。
显示估计的执行计划	对数据库内部的SQL执行过程进行的解释。
数据抽取	用于将当前数据集的数据抽取到MPP。
创建	可基于当前数据集快速创建透视分析、仪表分析、灵活分析以及自助数据集。

属性区

属性区用于设置数据集相关资源的属性。它主要包含两个部分：数据集输出区和属性面板。

- 数据集输出区：该区显示所有数据集输出的相关资源，该区通常与“属性面板”结合起来用于对这些资源的属性进行管理，包含：[输出字段属性](#)、[原生SQL数据集#检测输出字段](#)、[私有参数](#)、[输出参数](#)。
- 属性面板：该区需要与“工作区”结合使用，单击工作区中的任一资源，即在属性面板中列出该资源的相关属性信息。用户可以在属性面板中重新设置资源的相关属性。

输出字段

F184

F185

F186

私有参数

输出参数

属性名	属性值
属性描述	

输出字段

在数据集输出区会列出所有的输出字段。如下图，选中字段，在下面属性区可以对部分属性进行设置。

输出字段

F184

F185

F186

私有参数

输出参数

属性名

属性值

标识

BizViewOutField.l8a8a4c53016c07ef07efe06d0

名称

F184

别名

F184

描述

数据类型

字符串

数据格式

<字符串-默认值>

转换规则

否

脱敏规则

下面是对字段属性的详细说明：

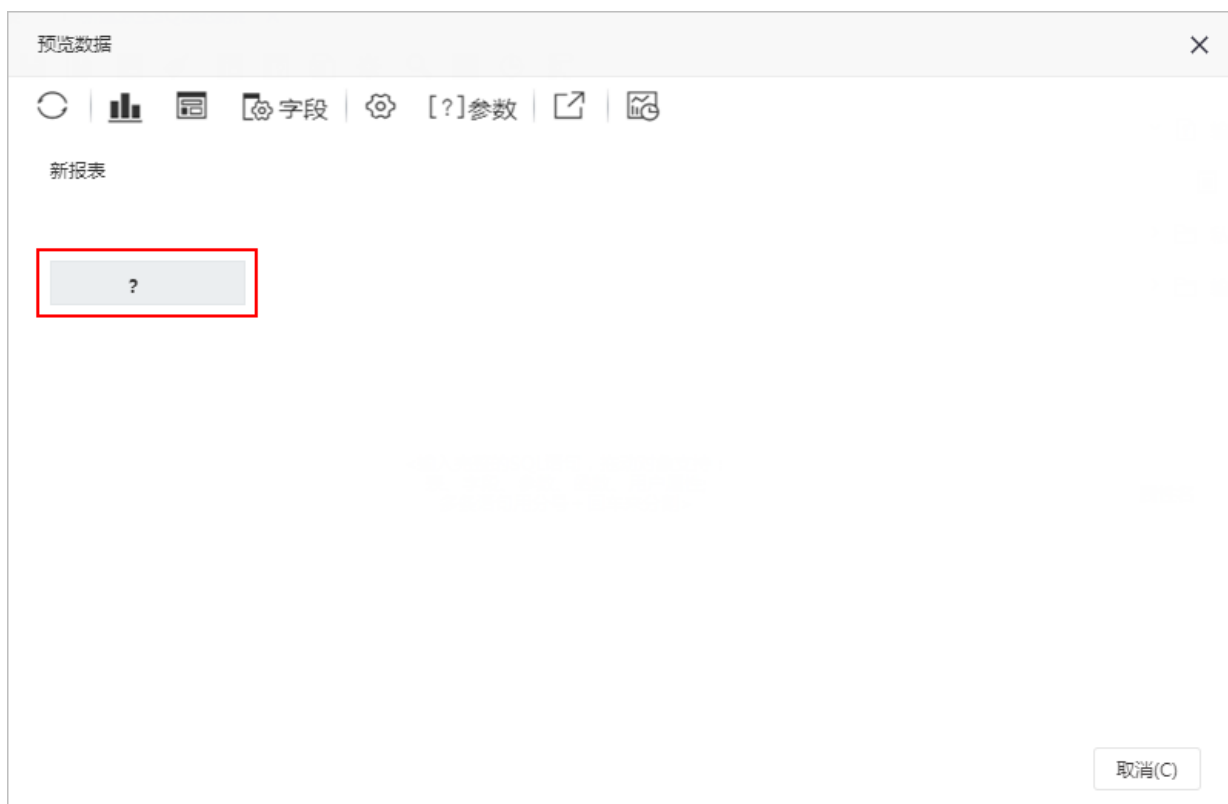
对象	设置项	说明
输出字段	标识	字段的全局唯一标识，由字段类型和其ID号共同构成。不允许重新设置。
	名称	字段的真名。不允许重新设置。
	别名	字段的别名，即显示的名字。允许重新设置。
	描述	对字段的相关描述，被用于其提示信息中。允许重新设置。
	数据类型	字段的数据类型，主要包含：整数、浮点、字符串、日期、时间、日期时间、文本、对象、集合。允许重新设置。
	数据格式	根据数据类型选择输出字段的数据格式，该格式选项在 数据格式 中进行设置。允许重新设置。
	转换规则	选择是否对该字段进行转换规则，若设置转换规则，其所选的转换规则在 转换规则 中定义。
	脱敏规则	数据脱敏 是指对某些敏感信息通过脱敏规则进行数据的变形，实现敏感隐私数据的可靠保护。

检测输出字段

新建的原生SQL数据集，必须要设置 **检测输出字段**，否则在右侧“数据集输出区”面板上“输出字段”下的字段显示成了问号。

示例说明：

1、如下图所示，新建一个“原生SQL数据集”，我们可以看到右侧的“数据集输出”面板上“输出字段”下并没有把所有的字段列出来，而是显示成了一个问号。同时在“预览数据”时，看到的表格上也只有一列，列名也是问号。



2、我们在SQL编辑器中可能会输入复杂的SQL语句，Smartbi自身是无法解析SQL语句并自动判断有哪些输出字段的。它只能利用数据库的功能，先去执行该SQL语句，根据数据库返回的结果集来判断有哪些输出字段。因此，我们需要在“输出字段”节点上点击鼠标右键，然后点击“检测输出字段”菜单项。



待输入的SQL语句执行成功后，在“输出字段”节点下，就会列出所有的字段。选择某字段，可以设置别名、数据格式，以及其他各种属性。



其他说明：

数据集处理需通过antlr等第三方工具包统一解析sql语句的语法树，因为mongo不是使用sql语法，目前产品只能解析处理部分常用的MongoDB查询指令，其他指令需要在find之前前面加上return，通过eval指令执行runCommand脚本函数支持。

使用时需注意如下：

1) 目前直接支持的指令关键字有：（即find命令如果只使用了下面关键字不需要在前面加return）

常规指令：

find、getCollection、limit、sort、projection、skip、toArray等；

查询条件：

\$gt、\$gte、\$lt、\$lte、\$eq、\$neq、\$in、\$nin、\$and、\$or、\$nor、\$not、\$regex等；

正则表达式部分字符（如字符```）与`expression antlr`冲突，find查询支持直接使用正则表达式和使用`new RegExp`对象，但是不能包含冲突字符。如果是比较复杂的正则表达式建议添加`return`使用脚本查询。

2) 目前支持的数据类型

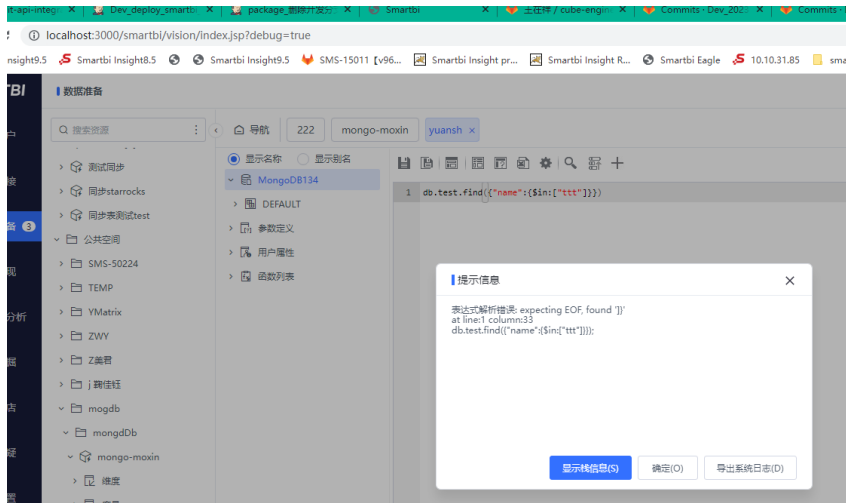
数值、字符、布尔、日期、正则、内嵌文档、常规数组；

产品只支持MongoDB规整结构的数据结构，即要求返回每行的数据结构是一样的，每行不一样只解析第一行的数据结构，结果可能会存在数据缺失。

3) 非单行指令，如需要设定变量等操作，都需要添加`return`使用脚本查询；

4) `]]`、`{[`等特殊字符的组合`antlr`第三方工具包无法正常解析，需要通过在中括号和大括号之间添加换行，进行格式化规避处理；

如直接执行`db.test.find({"name":{"in":["ttt"]}})`，会报错：



添加换行后即正常：

```
db.test.find(
{"name":{
$in:["ttt"]
}}).toArray()
```

5) 如果MongoDB开通了`auth`认证鉴权，使用`return`关键字执行脚本查询，MongoDB业务库配置的用户需要开通执行`eval`指令的权限，如有`anyAction`或`anyResource`的权限，如果没有执行`eval`指令的权限会提示报错，如下图所示



开通权限命令：

```
db.createRole({role:"executeFunctions",privileges:[{resource:
```



```
{anyResource:true},
actions:["anyAction"]},roles:[]})
db.grantRolesToUser("yourusername", [
{ role: "executeFunctions", db: "admin" }
])
```