



步骤4：创建PI标签

以下将在PI Tag配置中使用的查询（数据库中表的名称为“ RxC”）：

```
/ SQL = " SELECT Timestamp as PI_TIMESTAMP, concat (TagName, '_ Temperature') as PI_TAGNAME1, Temperature as PI_VALUE1, 0 as PI_STATUS1, concat (Ta
```

执行上述查询时，结果是下表（将其称为RDBMS表）：

PI_Timestamp	PI_TagName1	PI_Value1	(No column name)	PI_TagName2	PI_Value2	(No column name)
2016-03-30 08:53:00.000	Pump3_Temperature	890	0	Pump3_Pressure	0	0
2016-03-30 09:50:00.000	Pump1_Temperature	730	0	Pump1_Pressure	0	0
2016-03-30 09:51:00.000	Pump2_Temperature	840	0	Pump2_Pressure	0	0
2016-03-30 09:52:00.000	Pump3_Temperature	850	0	Pump3_Pressure	0	0
2016-03-30 09:49:00.000	Pump1_Temperature	900	0	Pump1_Pressure	0	0
2016-03-30 09:52:00.000	Pump2_Temperature	900	0	Pump2_Pressure	0	0
2016-03-30 09:53:00.000	Pump3_Temperature	890	0	Pump3_Pressure	0	0
2016-03-30 10:05:00.000	Pump1_Temperature	1	0	Pump1_Pressure	0	0
2016-03-30 10:06:00.000	Pump2_Temperature	2	0	Pump2_Pressure	0	0
2016-03-30 10:07:00.000	Pump3_Temperature	3	0	Pump3_Pressure	0	0
2016-03-30 10:07:00.000	Pump1_Temperature	4	0	Pump1_Pressure	0	0
2016-03-30 10:06:00.000	Pump2_Temperature	5	0	Pump2_Pressure	0	0
2016-03-30 10:05:00.000	Pump3_Temperature	6	0	Pump3_Pressure	0	0
2016-03-30 10:16:00.000	Pump1_Temperature	10	0	Pump1_Pressure	10	0
2016-03-30 10:16:00.000	Pump2_Temperature	20	0	Pump2_Pressure	20	0
2016-03-30 10:16:00.000	Pump3_Temperature	30	0	Pump3_Pressure	30	0
2016-03-30 10:17:00.000	Pump1_Temperature	40	0	Pump1_Pressure	40	0

目标是将上述RDBMS表中的值分配给以下六个标记：

1. Pump1_Temperature
2. Pump2_Temperature
3. Pump3_Temperature
4. Pump1_Pressure
5. Pump2_Pressure
6. Pump3_Pressure

可以通过Point Builder插件使用PI System Management Tools (SMT) 创建PI标签。使用以下标记配置：

分销商/主标签

常规选项卡	
名称	RxCDistributor
点源	RDB
点类型	Float32 (或此标记的任何数据类型)
Exdesc	/ SQL = " SELECT Timestamp as PI_TIMESTAMP, concat (TagName, '_ Temperature') as PI_TAGNAME1, Temperature as PI_VALUE1, 0 as PI_STATUS1, concat (TagName, '_ Pressure') as PI_TAGNAME2, Pressure as PI_VALUE2, 0 as PI_STATUS2, 0 as PI_STATUS2 FROM RxC WHERE Timestamp >? ORDER BY Timestamp ASC;" P1 = TS
经典标签	
LOCATION1	1 (接口ID)

LOCATION2	RxC不适用
LOCATION3	-2
LOCATION4	1 (扫描类1)
location5	0 (启用例外报告)

目标/数据标签

常规选项卡	
名称	Pump1_Temperature, Pump1_Pressure, Pump2_Temperature, ... 注意: 如果不使用RDBMS表标签名称, 你必须列出这个名字作为ExDesc领域的别名 (/别名="标记名"), 这样的分销商标签可以决定哪些值发送到哪些标签。
点源	RDB
点类型	Float32 (或此标记的任何数据类型)
Exdesc	
经典标签	
LOCATION1	1 (接口ID)
LOCATION2	RxC不适用
LOCATION3	0
LOCATION4	1 (扫描类别1) (除非使用/ ALIAS, 否则应匹配主标记)
location5	0 (启用例外报告)

在此示例中, 表中没有状态列, 因此该列在查询中被替换为“0” (“0”表示状态=良好)。必须在SELECT语句中具有STATUS的值 (如果为0, 则为Status_ColumnTitle)。

上面的示例是基于扫描的。要配置标签以基于触发标签进行更新, 请参见“其他说明”下的**事件驱动标签**。

警告: TS占位符是查询返回的最新时间戳, 可与单个标记查询一起使用, 以仅返回尚未扫描的最新行。但是, 使用“标记分配”策略, TS占位符将带有当前时间的时间戳。因此, 如果接口正在快速扫描, 则TS占位符将在每次接口扫描时更新 (这是分配给Distributor标记的查询执行时间和时间戳)。在这种情况下, 如果某些标签的更新速度比扫描时间慢, 则可能会丢失某些行。

为避免上述问题, 可能采取的措施如下:

- 将P1设置为等于“标记名” / TS, 其中“标记名”是查询表中预期更新最慢的标记的名称。
- 在查询表中创建一个额外的列, 该列将在每次扫描后更新。在查询中添加一条语句, 以标记已扫描的每一行。包含第二条语句以仅扫描尚未更新的行。
- 在查询中创建另一个表, 其中TagName列和Timestamp列。每次扫描后将更新此表, 以仅显示每个TagName的最新时间戳。该查询会将此添加的表与原始查询的表进行比较, 以仅传递带有时间戳的行, 而不是创建的表中的时间戳。
- 添加一条WHERE语句, 该语句将仅在指定的时间范围内返回值, 在该时间段内所需的所有数据项将更新。可能会有一些重复的数据; 用Loc5设置或/ RDB接口参数控制它。例如, 如果所有标记都将在最近四个小时内确定更新, 即使某些标记多次更新, 则将SQL查询配置为始终返回最近四个小时的数据。

补充笔记**长SQL查询**

对于较长的SQL查询, 可以将查询另存为文件到接口计算机上的SQL Files Directory (在ICU的接口配置中指定了SQL Files Directory)。该文件必须具有扩展名.sql, 并且应包含查询中的所有内容。配置点时, 在“**仪器标签**”字段中指定文件名。可以在Exdesc字段中指定占位符规范。

事件驱动标签

可以配置RDBMS标记, 以便接口仅在触发标记更新时查询数据库。配置点时, 在“**源**”标签字段中指定此触发标签。另外, 位置4应该设置为“0”。触发器标签也可以在Exdesc字段中指定为/ EVENT = tagname或/ TRIG = tagname

找不到您想要的东西吗?



开启新案



查看联系方式