文章
姚 鑫 · 十月 28, 2021 阅读大约需 7 分钟

第五十九章 SQL命令 JOIN（一）

第五十九章 SQL命令 JOIN（一）

基于两个表中的数据创建表的SELECT子句。

大纲

```
table1 [[AS] t-alias1] CROSS JOIN table2 [[AS] t-alias2] |
table1 [[AS] t-alias1] , table2 [[AS] t-alias2]

 table1 [[AS] t-alias1]
NATURAL [INNER] JOIN |
NATURAL LEFT [OUTER] JOIN |
NATURAL RIGHT [OUTER] JOIN |
table2 [[AS] t-alias2]

table1 [[AS] t-alias1]
[INNER] JOIN |
LEFT [OUTER] JOIN |
RIGHT [OUTER] JOIN |
FULL [OUTER] JOIN
table2 [[AS] t-alias2]
ON condition-expression

table1 [[AS] t-alias1]
[INNER] JOIN |
LEFT [OUTER] JOIN |
RIGHT [OUTER] JOIN |
table2 [[AS] t-alias2]
USING (identifier-commalist)
```

(上述联接语法用于SELECT语句FROM子句。可以在其他SELECT语句子句中使用其他联接语法。)

描述

联接是将两个表组合在一起以生成联接表的操作，可以选择遵守一个或多个限制条件。新表的每一行都必须满足限制条件。联接提供了将一个表中的数据与另一个表中的数据链接起来的方法，并且经常用于定义报表和查询。

有几种表示联接的语法形式。首选形式是在SELECT语句中指定显式联接表达式作为FROM子句的一部分。FROM子句联接表达式可以包含多个联接。

注意： SQL还支持在SELECT语句SELECT-ITEM列表、WHERE子句、ORDER BY子句和其他地方使用箭头语法(->)的隐式联接。指定隐式联接以执行表与另一个表中的字段的左外联接；指定显式联接以联接两个表。这种隐式联接语法可以很好地替代显式联接语法，或者与显式联接语法一起出现在同一查询中。但是，在组合箭头语法和显式连接语法方面有一些重要的限制。这些限制如下所述。

IRIS使用复杂的优化算法来最大化连接操作的性能。它不一定按照表的指定顺序联接表。相反，SQL优化器根据每个表的Tune Table数据(以及其他因素)确定表连接顺序。因此，在复杂SQL查询中使用表之前，必须先针对表运行调优表，这一点很重要。

在大多数情况下，SQL优化器策略提供最佳结果。但是， IRIS还提供联接优化关键字，如%FIRSTTABLE、%INORDER和%FULL，可以在FROM关键字之后立即使用这些关键字来覆盖特定查询的默认优化策略。

JOIN 定义

IRIS支持多种不同的连接语法形式。但是，这许多公式涉及以下五种类型的联接。

ANSI连接语法	句法上的等价于
CROSS JOIN	与符号表示相同：FROM子句中的Table1、Table2(用逗号分隔的表列表)。
INNER JOIN	与JOIN相同。符号表示：“ = ” (在WHERE子句中)。
LEFT OUTER JOIN	与左连接相同。箭头语法(->)还执行左外部联接。
RIGHT OUTER JOIN	与右连接相同。
FULL OUTER JOIN	与FULL JOIN相同。

除非另有说明，否则所有连接语法都在FROM子句中指定。

- 交叉连接是将第一个表的每一行与第二个表的每一行交叉的连接。
  这将产生一个笛卡尔积，即一个具有大量数据重复的、逻辑上全面的大型表。
  通常这种连接是通过在FROM子句中提供一个逗号分隔的表列表来执行的，然后使用WHERE子句来指定限制性条件。
  %INORDER或%STARTTABLE优化关键字不能用于交叉连接。
  尝试这样做会导致SQLCODE -34错误。
- INNER JOIN是将第一个表的行与第二个表的行连接起来的连接，不包括在第一个表中没有在第二个表中找到相应行的任何行。
- 左OUTER JOIN和右OUTER JOIN在大多数方面功能相同(语法相反)，因此经常统称为单向外部连接。
  单向外部连接是将第一个(源)表的行与第二个表的行链接在一起的连接，包括第一个表的所有行，即使第二个表中没有匹配。
  这将导致第一个(源)表的某些字段可能与NULL数据配对。
- 在指定单向外部连接时，在FROM子句中命名表的顺序非常重要。
  对于LEFT OUTER JOIN，指定的第一个表是该连接的源表。
  对于RIGHT OUTER JOIN，指定的第二个表是连接的源表。
- FULL OUTER JOIN是将在两个表上执行左OUTER JOIN和右OUTER JOIN的结果组合在一起的连接。
  它包括在第一个表或第二个表中找到的所有行，并在两边的缺失匹配中填充null。

CROSS JOIN 注意事项

显式使用JOIN关键字比使用逗号语法指定交叉连接具有更高的优先级。
IRIS将t1,t2 JOIN t3解释为t1，(t2 JOIN t3)。

不能执行涉及本地表和通过ODBC或JDBC网关连接链接的外部表的交叉连接。
例如，FROM Sample.Person,Mylink.Person。
尝试这样做的结果是SQLCODE -161: “ 对SQL连接的引用必须构成整个子查询 ” 。
要执行此交叉连接，必须将链接表指定为子查询。
例如，FROM Sample。
Person，(SELECT * FROM Mylink.Person)。

自然连接

NATURAL JOIN是以NATURAL关键字为前缀的INNER JOIN、LEFT OUTER JOIN或RIGHT OUTER JOIN。
在连接前加上单词NATURAL，说明正在连接具有相同名称的两个表的所有列。
由于NATURAL连接对具有相同名称的所有列自动执行相等条件，因此不可能指定on子句或USING子句。
尝试这样做会导致SQLCODE -25错误。

对于NATURAL连接的两个操作数，只支持简单的基表引用(不支持视图或子查询)。

只能将NATURAL连接指定为连接表达式中的第一个连接。

NATURAL连接不会合并名称相同的列。

FULL JOIN不能以NATURAL关键字作为前缀。
尝试这样做会导致SQLCODE -94错误。

ON 子句

内连接、左外连接、右外连接或全外连接都可以有ON子句。
ON子句包含一个或多个条件表达式，用于限制连接操作返回的值。
带有ON子句的连接可以在连接表达式中的任何位置指定。
带有ON子句的连接可以为连接的任一操作数指定表、视图或子查询。

ON子句由一个或多个条件表达式谓词组成。
其中包括SQL支持的大多数谓词。
但是，不能使用FOR SOME %ELEMENT集合谓词来限制连接操作。

可以使用AND、OR和NOT逻辑操作符关联多个条件表达式。
AND优先于OR。
括号可以用来嵌套和分组条件表达式。
除非用括号分组，否则使用相同逻辑运算符的谓词严格按照从左到右的顺序执行。

ON子句有以下限制:

- 带有ON子句的连接只能使用ANSI连接关键字语法。
- 带有ON子句的连接不能使用NATURAL关键字前缀。
  这将导致SQLCODE -25错误。
- 带有ON子句的连接不能接受USING子句。
  这将导致SQLCODE -25错误。
- ON子句不能包含箭头语法(- >)。
  这将导致SQLCODE -67错误。
- ON子句只能引用ANSI关键字JOIN操作中显式指定的表。
  在FROM子句中指定的其他表不能在ON子句中引用。
  这将导致SQLCODE -23错误。
- ON子句只能引用位于JOIN操作数中的列。
  多个连接中的语法优先级可能会导致ON子句失败。
  例如，查询SELECT * FROM t1,t2 JOIN t3 ON t1.p1=t3.p3失败，因为t1和t3不是join的操作数;
  t1连接t2 JOIN t3的结果集。
  SELECT * FROM t1 CROSS JOIN t2 JOIN t3 ON t1.p1=t3。
  p3或者SELECT * FROM t2,t1 JOIN t3 ON t1.p1=t3.p3。

```
SELECT *
FROM Table1
   LEFT JOIN Table2 ON Table1.k = Table2.k
   LEFT JOIN Table3 ON COALESCE(Table1.k,Table2.k) = Table3.k
```

使用FULL OUTER JOIN或RIGHT OUTER JOIN的类似示例也有这个限制。

ON子句索引

为了获得最佳性能，ON子句中引用的字段应该(在大多数情况下)具有关联索引。

ON子句可以使用只满足某些连接条件的现有索引。
在多个字段上指定条件的ON子句可以使用只包含这些字段子集的索引作为下标，以部分满足连接;
IRIS将直接从表中测试其余字段的连接条件。

ON子句中引用的字段的排序规则类型应该与它在相应索引中的排序规则类型匹配。
排序规则类型不匹配可能导致索引不被使用。
但是，如果连接条件位于%EXACT字段值上，但只有排序字段值上的索引可用，
IRIS可以使用该索引来限制要检查的行以获取准确值。

在一些非常特殊的情况下，可能希望通过在ON子句条件前面加上%NOINDEX关键字来防止索引的使用。

USING 子句

INNER JOIN、LEFT OUTER JOIN或RIGHT OUTER JOIN可以有USING子句。
对于使用USING子句的连接的操作数，只支持简单的基表引用(不支持视图或子查询)。
带有USING子句的连接只能指定为连接表达式中的第一个连接。
使用USING子句的连接不能使用NATURAL关键字前缀或ON子句。

USING子句列出一个或多个列名，列名由逗号分隔，用括号括起来。
括号是必需的。
只允许显式的列名;
%ID不允许。
重复的列名被忽略。
USING子句不会合并名称相同的列。

USING子句是表示ON子句中表达的相等条件的一种简单方式。因此：t1 INNER JOIN t2 USING (a,b)等价于t1.a=t2.a和t1.b=t2.b上的T1内连接T2

#SQL #Caché

源URL:
https://cn.community.intersystems.com/post/%E7%AC%AC%E4%BA%94%E5%8D%81%E4%B9%9D%E7%AB%A0-sql%E5%91%BD%E4%BB%A4-join%EF%BC%88%E4%B8%80%EF%BC%89