

### 文章

姚鑫 · 十二月 1, 2021 阅读大约需 6 分钟

## 第二章 SQL谓词的概述（二）

## 第二章 SQL谓词的概述（二）

### 谓词和%SelectMode

所有谓词都使用逻辑(内部存储)数据值进行比较。

但是，有些谓词可以对谓词值执行格式模式转换，将谓词值从ODBC或Display格式转换为Logical格式。其他谓词不能执行格式模式转换，因此必须始终以Logical格式指定谓词值。

执行格式模式转换的谓词确定是否需要从匹配字段的数据类型(如DATE%List)进行转换，并确定从%SelectMode设置进行转换的类型。

如果%SelectMode设置为逻辑格式以外的值(例如%SelectMode=ODBC或%SelectMode=Display)，则必须以正确的ODBC或Display格式指定谓词值。

- 相等谓词执行格式模式转换。  
IRIS将谓词值转换为逻辑格式，然后与字段值进行匹配。  
如果%SelectMode设置为逻辑格式以外的模式，则必须以%SelectMode格式(ODBC或Display)指定显示值与逻辑存储值不同的数据类型的数据类型的谓词值。  
例如，日期、时间和%list格式的字符串。  
因为IRIS会自动执行这种格式转换，所以在Logical格式中指定这种类型的谓词值通常会导致SQLCODE错误。  
例如，SQLCODE -146 “无法将日期输入转换为有效的逻辑日期值” (IRIS假设提供的逻辑值是ODBC或Display值，并试图将其转换为逻辑值——但没有成功)。  
受影响的谓词包括=、<、>、BETWEEN和IN。
- 模式谓词不能执行格式模式转换，因为IRIS不能有意义地转换谓词值。  
因此，谓词值必须以Logical格式指定，而不管%SelectMode设置如何。  
以ODBC或Display格式指定谓词值通常会导致没有数据匹配或意外的数据匹配。  
受影响的谓词包括%INLIST、LIKE、%MATCHES、%PATTERN、%STARTSWITH、[(Contains操作符)和](Follows操作符)。

可以使用%INTERNAL、%EXTERNAL或%ODBCOUT格式转换函数来转换谓词操作的字段。

这允许以另一种格式指定谓词值。

例如，WHERE %ODBCOut(DOB) %STARTSWITH '1955-'。

但是，在匹配字段上指定格式转换函数将阻止对该字段使用索引。

这可能会对性能产生显著的负面影响。

在以下动态SQL示例中，BETWEEN谓词(相等谓词)必须以%SelectMode=1 (ODBC)格式指定日期：

```
ClassMethod Predicates3()  
{  
    s q1 = "SELECT Name,DOB FROM Sample.Person "  
    s q2 = "WHERE DOB BETWEEN '1950-01-01' AND '1960-01-01' "  
    s myquery = q1 _ q2  
    s tStatement = ##class(%SQL.Statement).%New()  
    s tStatement.%SelectMode = 1  
    s qStatus = tStatement.%Prepare(myquery)  
    if qStatus '= 1 {
```

## 第二章 SQL谓词的概述 (二)

Published on InterSystems Developer Community (<https://community.intersystems.com>)

---

```
w "%Prepare failed:"
d $System.Status.DisplayError(qStatus)
q
}
s rset = tStatement.%Execute()
d rset.%Display()
w !,"End of data"
}
```

```
DHC-APP>d ##class(PHA.TEST.SQLCommand).Predicates3()
```

Name	DOB
Houseman,Martin D.	1955-09-25
Ingrahm,Yan S.	1954-06-15
Smith,Elvis Y.	1955-06-29
Gore,Alfred M.	1958-09-15
Yoders,Liza U.	1959-06-05
Ng,Liza Z.	1955-10-05
Yeats,Debby G.	1951-12-06

在以下动态SQL示例中，%STARTSWITH谓词(模式谓词)不能执行格式模式转换。  
第一个示例尝试以%SelectMode=ODBC格式为20世纪50年代的日期指定%STARTSWITH。  
但是，由于该表不包含以\$HOROLOG 195开始的出生日期(日期在1894年)，所以没有选择行:

```
ClassMethod Predicates4()
{
    s q1 = "SELECT Name,DOB FROM Sample.Person "
    s q2 = "WHERE DOB %STARTSWITH '195'"
    s myquery = q1 _ q2
    s tStatement = ##class(%SQL.Statement).%New()
    s tStatement.%SelectMode = 1
    s qStatus = tStatement.%Prepare(myquery)
    if qStatus '= 1 {
        w "%Prepare failed:"
        d $System.Status.DisplayError(qStatus)
        q
    }
    s rset = tStatement.%Execute()
    q rset.%Display()
    w !,"End of data"
}
```

下面的示例在匹配的DOB字段上使用%ODBCOut格式转换函数，以便%STARTSWITH可用ODBC格式选择20世纪50年代的年份。  
但是，请注意，这种用法会阻止在DOB字段上使用索引。

```
ClassMethod Predicates5()
{
    s q1 = "SELECT Name,DOB FROM Sample.Person "
    s q2 = "WHERE %ODBCOut(DOB) %STARTSWITH '195'"
    s myquery = q1_q2
    s tStatement = ##class(%SQL.Statement).%New()
    s tStatement.%SelectMode=1
    s qStatus = tStatement.%Prepare(myquery)
```

## 第二章 SQL谓词的概述 (二)

Published on InterSystems Developer Community (<https://community.intersystems.com>)

---

```
if qStatus '= 1 {
    w "%Prepare failed:"
    d $System.Status.DisplayError(qStatus)
    q
}
s rset = tStatement.%Execute()
d rset.%Display()
w !,"End of data"
}
```

```
DHC-APP>d ##class(PHA.TEST.SQLCommand).Predicates5()
```

Name	DOB
Houseman,Martin D.	1955-09-25
Ingrahm,Yan S.	1954-06-15
Smith,Elvis Y.	1955-06-29
Gore,Alfred M.	1958-09-15
Yoders,Liza U.	1959-06-05
Ng,Liza Z.	1955-10-05
Yeats,Debby G.	1951-12-06
Zweifelhofer,Zelda J.	1954-02-19

在下面的示例中，%STARTSWITH谓词为逻辑(内部)格式的日期指定了%STARTSWITH。  
选择DOB逻辑值以41开始的行(日期从1953年4月4日(\$HOROLOG 41000)到1955年12月28日(\$HOROLOG 41999))。

使用DOB字段索引:

```
lassMethod Predicates6()
{
    s q1 = "SELECT Name,DOB FROM Sample.Person "
    s q2 = "WHERE DOB %STARTSWITH '41'"
    s myquery = q1 _ q2
    s tStatement = ##class(%SQL.Statement).%New()
    s tStatement.%SelectMode = 1
    s qStatus = tStatement.%Prepare(myquery)
    if qStatus '= 1 {
        w "%Prepare failed:"
        d $System.Status.DisplayError(qStatus)
        q
    }
    s rset = tStatement.%Execute()
    d rset.%Display()
    w !,"End of data"
}
```

```
DHC-APP>d ##class(PHA.TEST.SQLCommand).Predicates6()
```

Name	DOB
Houseman,Martin D.	1955-09-25
Ingrahm,Yan S.	1954-06-15
Smith,Elvis Y.	1955-06-29
Ng,Liza Z.	1955-10-05
Zweifelhofer,Zelda J.	1954-02-19
Zampitello,Josephine Q.	1953-08-14
Hertz,Uma C.	1954-07-25

Davis,Jane E. 1953-07-28

Vanzetti,Alexandra O. 1953-12-29

9 Rows(s) Affected

## 谓词和PosixTime、时间戳和日期

相等谓词比较自动在这些不同的date和datetime表示之间执行转换。

这个转换与%SelectMode无关。

因此，下面是所有有意义的比较谓词:

```
WHERE MyPosixField = MyTimestampField
WHERE MyPosixField < CURRENT_TIMESTAMP
WHERE MyPosixField BETWEEN DATEADD('month',-1,CURRENT_TIMESTAMP) AND $HOROLOG
WHERE MyPosixField BETWEEN DATEADD('day',-1,CURRENT_DATE) AND LAST_DAY(CURRENT_DATE)
```

模式谓词比较，如%STARTSWITH，不执行不同日期和日期时间表示之间的转换。

对实际存储数据值的操作。

## 取消文字替换

通过将谓词参数括在双圆括号中，可以在编译前分析期间禁止文字替换。例如，LIKE(( 'ABC%' ))。这可以通过提高总体选择性和/或下标绑定选择性来提高查询性能。但是，当使用不同的值多次调用同一查询时，应该避免这种情况，因为这将导致为每个查询调用创建一个单独的缓存查询。

## 示例

下面的示例在查询的WHERE子句中使用了各种条件:

```
SELECT PurchaseOrder FROM MyTable
  WHERE OrderTotal >= 1000
        AND ItemName %STARTSWITH :partname
        AND AnnualOrders BETWEEN 50000 AND 100000
        AND City LIKE 'Ch%'
        AND CustomerNumber IN
          (
            SELECT CustNum FROM TheTop100
            WHERE TheTop100.City='Boston'
          )
  AND :minorder > SOME
    (
      SELECT OrderTotal FROM Orders
      WHERE Orders.Customer = :cust
    )
```

[#SQL #Caché](#)

## 第二章 SQL谓词的概述（二）

Published on InterSystems Developer Community (<https://community.intersystems.com>)

---

URL:

<https://cn.community.intersystems.com/post/%E7%AC%AC%E4%BA%8C%E7%AB%A0-sql%E8%B0%93%E8%AF%8D%E7%9A%84%E6%A6%82%E8%BF%B0%EF%BC%88%E4%BA%8C%EF%BC%89>