

# Example of Value Constraints Used in Queries

## Value Constraint by Number

If the fact with a numerical value didn't have the normalized numerical value with a single *UNIT\_CD* for a particular concept, then the user can tell the service to do the unit conversion of the *NVAL\_NUM* column before applying the value constraints in the query. The unit conversion of *NVAL\_NUM* is calculated using the concept's metadata xml defined in the Ontology cell (<ConvertingUnits/>, <MultiplyingFactor/>). To enable the unit conversion, set the following project parameter in the Project Management cell.  
 CRC\_ENABLE\_UNITCD\_CONVERSION = ON | OFF

Value	Description
ON	Unit conversion is enabled
OFF	Unit conversion is not enabled

- *This unit conversion option will slow down the query. For better query performance load, the numerical fact values in the in the normalized units and do not enable this option.*

<b>Greater than operator</b>	
<b>Query Numeric Value Constraint:</b>	<pre>&lt;constrain_by_value&gt; &lt;value_operator&gt;GT&lt;/value_operator&gt; &lt;value_constraint&gt;99.9&lt;/value_constraint&gt; &lt;value_type&gt;NUMBER&lt;/value_type&gt; &lt;/constrain_by_value&gt;</pre>
<b>Numeric Constraint SQL:</b>	<pre>(valtype_cd = 'N' AND nval_num &gt; 99.9 AND tval_char IN ('GE','E')) OR (valtype_cd = 'N' AND nval_num &gt;= 99.9 AND tval_char = 'G')</pre> <p><b><i>Unit Conversion Enabled</i></b></p> <pre>(valtype_cd = 'N' AND case when unit_cd = 'mg/5ml' then nval_num &gt; 99.9 * 5 when unit_cd = 'mg/15ml' then nval_num &gt; 99.9 * 15 when unit_cd = 'mg/0.5ml' then nval_num &gt; 99.9 * 0.5 AND tval_char IN ('GE','E')) OR (valtype_cd = 'N' AND case when unit_cd = 'mg/5ml' then nval_num &gt; 99.9 * 5 when unit_cd = 'mg/15ml' then nval_num &gt; 99.9 * 15 when unit_cd = 'mg/0.5ml' then nval_num &gt; 99.9 * 0.5 AND tval_char = 'G')</pre>
<b>Less than operator</b>	
<b>Query Numeric Value Constraint:</b>	<pre>&lt;constrain_by_value&gt; &lt;value_operator&gt;LT&lt;/value_operator&gt; &lt;value_constraint&gt;99.9&lt;/value_constraint&gt; &lt;value_type&gt;NUMBER&lt;/value_type&gt; &lt;/constrain_by_value&gt;</pre>
<b>Numeric Constraint SQL:</b>	<pre>(valtype_cd = 'N' AND nval_num &lt; 99.9 AND tval_char IN ('LE','E')) OR (valtype_cd = 'N' AND nval_num &lt;= 99.9 AND tval_char = 'L')</pre>
<b>Between operator</b>	
<b>Query Numeric Value Constraint:</b>	<pre>&lt;constrain_by_value&gt; &lt;value_operator&gt;BETWEEN&lt;/value_operator&gt; &lt;value_constraint&gt;1 and 100&lt;/value_constraint&gt; &lt;value_type&gt;NUMBER&lt;/value_type&gt; &lt;/constrain_by_value&gt;</pre>
<b>Numeric Constraint SQL:</b>	<pre>(valtype_cd = 'N' AND nval_num BETWEEN 1 and 100 AND tval_char = 'E')</pre>
<b>Equal to operator</b>	

<b>Query Numeric Value Constraint:</b>	<constrain_by_value> <value_operator>EQ</value_operator> <value_constraint>99.9</value_constraint> <value_type>NUMBER</value_type> </constrain_by_value>
<b>Numeric Constraint SQL:</b>	(valtype_cd = 'N' AND nval_num = 99.9 AND tval_char = 'E')
<b>Less than and Equal to operator</b>	
<b>Query Numeric Value Constraint:</b>	<constrain_by_value> <value_operator>LE</value_operator> <value_constraint>99.9</value_constraint> <value_type>NUMBER</value_type> </constrain_by_value>
<b>Numeric Constraint SQL:</b>	(valtype_cd = 'N' AND nval_num <= 99.9 AND tval_char IN ('L','E','LE'))
<b>Greater than and Equal to operator</b>	
<b>Query Numeric Value Constraint:</b>	<constrain_by_value> <value_operator>GE</value_operator> <value_constraint>99.9</value_constraint> <value_type>NUMBER</value_type> </constrain_by_value>
<b>Numeric Constraint SQL:</b>	(valtype_cd = 'N' AND nval_num >= 99.9 AND tval_char IN ('G','E','GE'))
<b>Not Equal operator</b>	
<b>Query Numeric Value Constraint:</b>	<constrain_by_value> <value_operator>NE</value_operator> <value_constraint>99.9</value_constraint> <value_type>NUMBER</value_type> </constrain_by_value>
<b>Numeric Constraint SQL:</b>	(valtype_cd = 'N' AND nval_num <> 99.9 AND tval_char <> 'NE') OR (valtype_cd = 'N' AND nval_num = 99.9 AND tval_char = 'NE')

## Value Constraint by Text

<b>Equals operator</b>	
<b>Query Text Value Constraint:</b>	<constrain_by_value> <value_operator>EQ</value_operator> <value_constraint>H</value_constraint> <value_type>TEXT</value_type> </constrain_by_value>
<b>Text Value Constraint SQL:</b>	valtype_cd = 'T' AND tval_char = 'H'
<b>Not equals operator</b>	
<b>Query Text Value Constraint:</b>	<constrain_by_value> <value_operator>NE</value_operator> <value_constraint>L</value_constraint> <value_type>TEXT</value_type> </constrain_by_value>
<b>Text Value Constraint SQL:</b>	valtype_cd = 'T' AND tval_char <> 'L'
<b>Like operator</b>	
<b>Query Text Value Constraint:</b>	<constrain_by_value> <value_operator>LIKE</value_operator> <value_constraint>L</value_constraint> <value_type>TEXT</value_type> </constrain_by_value>
<b>Text Value Constraint SQL:</b>	valtype_cd = 'T' AND tval_char LIKE 'L%')

<b>In operator</b>	
<b>Query Numeric Value Constraint:</b>	<constrain_by_value> <value_operator>IN</value_operator> <value_constraint>'A','B'</value_constraint> <value_type>TEXT</value_type> </constrain_by_value>
<b>Text Value Constraint SQL:</b>	valtype_cd = 'T' AND tval_char = ('A','B')
<b>Between operator</b>	
<b>Query Text Value Constraint:</b>	<constrain_by_value> <value_operator>BETWEEN</value_operator> <value_constraint>'A' and 'B'</value_constraint> <value_type>TEXT</value_type> </constrain_by_value>
<b>Text Value Constraint SQL:</b>	valtype_cd = 'T' tval_char BETWEEN 'A' AND 'B'

## Value Constraint by Flag

<b>Equals operator</b>	
<b>Query Flag Value Constraint:</b>	<constrain_by_value> <value_operator>EQ</value_operator> <value_constraint>H</value_constraint> <value_type>FLAG</value_type> </constrain_by_value>
<b>Flag Value Constraint SQL:</b>	valueflag_cd = 'H'
<b>Not equals operator</b>	
<b>Query Flag Value Constraint:</b>	<constrain_by_value> <value_operator>NE</value_operator> <value_constraint>L</value_constraint> <value_type>FLAG</value_type> </constrain_by_value>
<b>Flag Value Constraint SQL:</b>	valueflag_cd <> 'H'
<b>In operator</b>	
<b>Query Flag Value Constraint:</b>	<constrain_by_value> <value_operator>IN</value_operator> <value_constraint>'A','B'</value_constraint> <value_type>FLAG</value_type> </constrain_by_value>
<b>Flag Value Constraint SQL:</b>	valueflag_cd IN ('A', 'B')