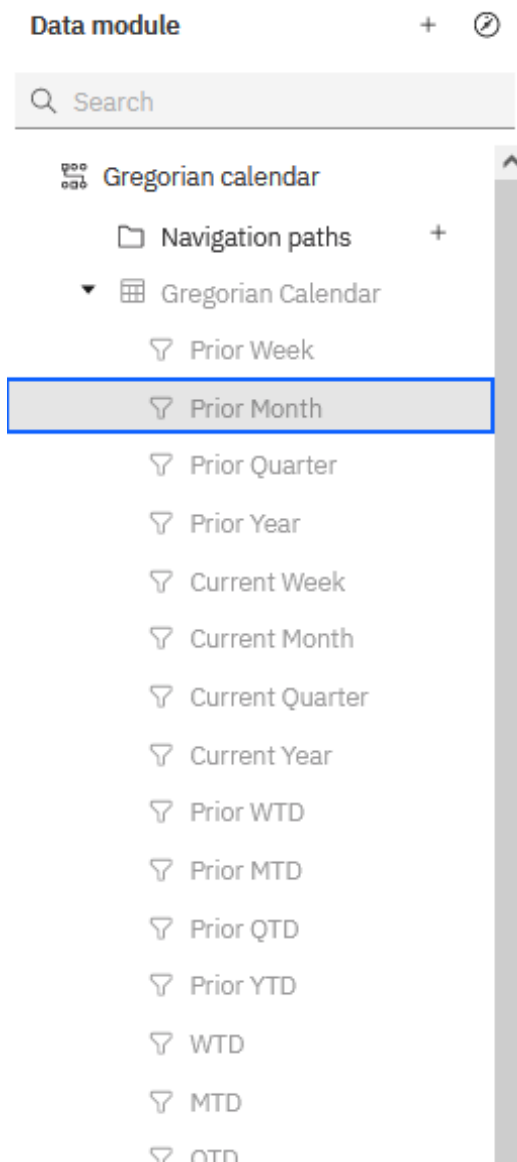


Gregorian Calendar with Dynamic As Of Date

How does it work?

The revised *Gregorian Calendar* data module has been updated to include the **optional** specification of the *Time Perspective* global parameter. Specifically, all the underlying filters (e.g. Prior Week, Prior Month, etc) that are specified in the table *Gregorian Calendar* have been modified:



For example, the **original** *Prior Month* filter expression was:

```
// This is a template expression that is used by the column property 'Lookup
reference'.
// To pass validation, the line below must remain as a comment. Do not remove the
forward slashes.
// validate: 1 = 1
#$_this.parent.idForExpression# >=
  #queryValue($_this.parent.split.ref + '.dMonth',
    $_this.parent.split.ref + '.TheDate = ' +
    queryValue($_this.parent.split.ref + '.PM_TheDate',
      $_this.parent.split.ref + '.TheDate = ' + $_as_of_date)
  )#
AND
#$_this.parent.idForExpression# <
  #queryValue($_this.parent.split.ref + '.dMonth',
    $_this.parent.split.ref + '.TheDate = ' + $_as_of_date)#
```

The **new** *Prior Month* filter expression is:

```
// This is a template expression that is used by the column property 'Lookup
reference'.
// To pass validation, the line below must remain as a comment. Do not remove the
forward slashes.
// validate: 1 = 1
#
let calTable = $_this.parent.split.ref;
let today = substr($current_timestamp, 0, 10);
let refDate =
  case tolower($[Time Perspective] )
  when 'use yesterday' then _add_days (today, -1)
  when 'use last sunday' then _add_days (today, '-' +
    queryValue(calTable + '.DayOfWeek', calTable + '
.TheDate = ' + today))
  else $_as_of_date
  end;
let parentId = $_this.parent.idForExpression;
let low = queryValue(calTable + '.dMonth',
  calTable + '.TheDate = ' +
  queryValue(calTable + '.PM_TheDate',
    calTable + '.TheDate = ' + refDate)
  );
let high = queryValue(calTable + '.dMonth',
  calTable + '.TheDate = ' + refDate);
parentId + ' >= ' + low + ' AND ' + parentId + ' < ' + high
#
```

The interesting part is in the computation of the macro variable *refDate*:

```
let refDate =
  case tolower($[Time Perspective] )
  when 'use yesterday' then _add_days (today, -1)
  when 'use last sunday' then _add_days (today, '-' +
    queryValue(calTable + '.DayOfWeek', calTable + '.TheDate = ' + today))
  else $_as_of_date
end;
```

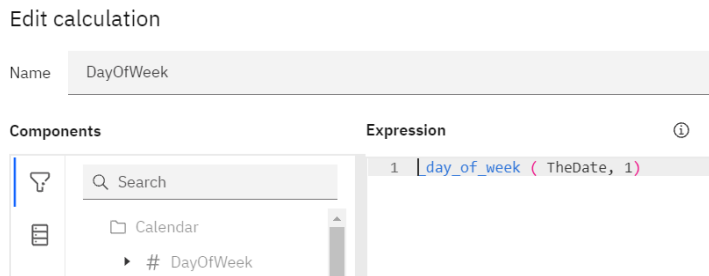
Note 1: The implementation of this feature in Cognos Analytics will first check for “use yesterday”, then for “use last Sunday”. If the parameter *Time Perspective* is not set, or if *Time Perspective* is set to anything but the two values “use yesterday” or “use last Sunday”, it will default to use the current setting of the *_as_of_date* parameter (which in turn defaults to today if it has not been set). If the user checks all three checkboxes, only the first will be used.

Time Perspective:

- use yesterday
- use last Sunday
- use *_as_of_date*

Note 2: If the global parameter *Time Perspective* is not created in your Cognos Analytics environment, the first two WHEN clauses above will be false, and the ELSE clause will be used.

Note 3: The computation for ‘use last sunday’ uses a call to *queryValue*. This call references the column *DayOfWeek*, which was added to the *Calendar* table with the expression:



This was needed because there is no macro function that computes the *_day_of_week*. However, there is a Business Time Function that can be used. This function is defined with the tip:

***_day_of_week* (date_expression, integer)**

Returns the day of week (1 to 7), where 1 is the first day of the week as indicated by the second parameter (1 to 7, 1 being Monday and 7 being Sunday). Note that in ISO 8601 standard, a week begins with Monday being day 1.

Example: *_day_of_week* (2003-01-01 , 1)

Result: 3