

# TARGET Period Table Task

TARGET Period Table Task is a custom component for SQL Server Integration Services. The component assists in creating and maintaining complex period tables.

## Main task solved

The Period Table task helps create and maintain period tables with a wide range of datetime based columns. When the task runs, the period table defined is dropped and re-created.

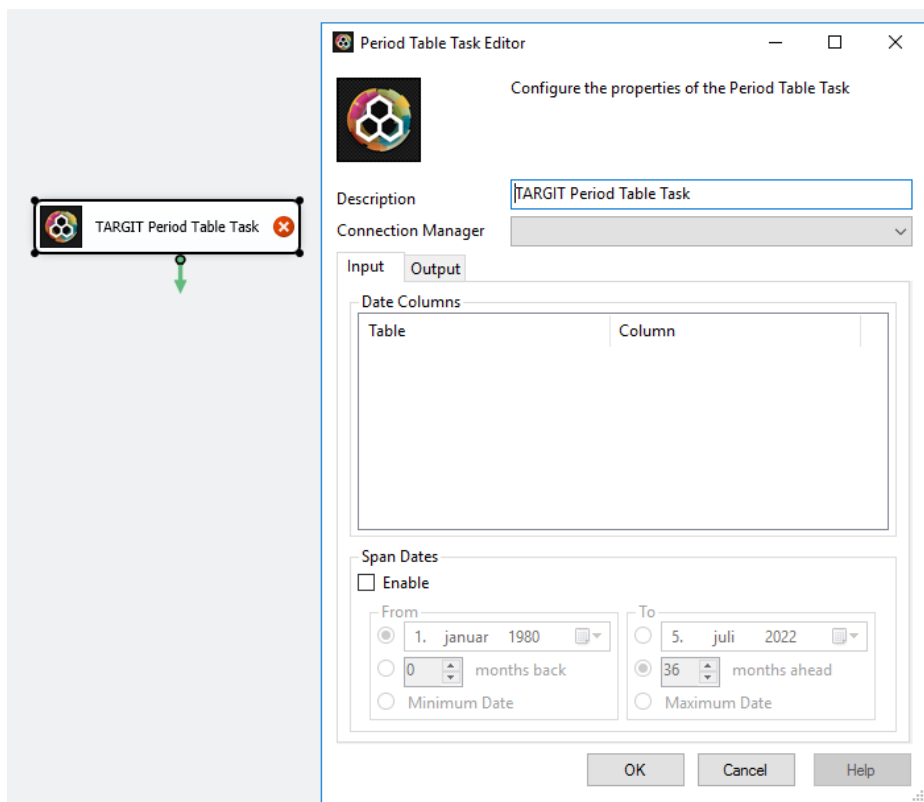
## Installation

The TARGET Period Table component is included in the "TARGET Custom SSIS Tasks" installation package found in the [TARGET App & Gauge Store](#) or the Download Center on the TARGET Portal. After installation, a restart of SQL Server Data Tools and the SSIS project is required for the task(s) to appear in the SSIS toolbox.

## How to use

### Getting started

- In the SSIS package, create an ADO.NET connection manager for the SQL Server database if not already available
- Drag the Period Table task from the SSIS Toolbox to the package Control Flow
- Double click the task and select the ADO.NET connection created.



### Input

In the Input tab, the unique dates to be used for creating the period table are selected. If the table is to contain only unique dates from one or more database tables, add the tables and date columns to be included. Dates will be selected as unique and the timestamp part will be included, so be careful with large tables with unique dates as the period table can be unnecessarily large. Consider truncating the time part from dates in the input tables if time is not relevant.

**Period Table Task Editor**

Configure the properties of the Period Table Task

Description: TARGIT Period Table Task

Connection Manager: 365 BC

Input Output

Date Columns

Table	Column
fSalesInvoiceLine	ShipmentDate
fGLEntry	PostingDate

Span Dates

☐ Enable

From: ☐ 1. januar 1980 ☐ 0 months back ☐ Minimum Date

To: ☐ 5. juli 2022 ☒ 36 months ahead ☐ Maximum Date

OK Cancel Help

The **Span Dates** function will inject any missing dates into the resulting period table in a selected interval if it is necessary to have all dates represented, e.g., for use with Balance at Date calculations. It can also be used for creating a period table without table inputs. This will result in a period table with all dates (without the time part) in the selected date range.

**Span Dates**

☒ Enable

From: ☐ 1. januar 2010 ☐ 0 months back ☒ Minimum Date

To: ☐ 5. juli 2022 ☒ 36 months ahead ☐ Maximum Date

#### Span Dates - From

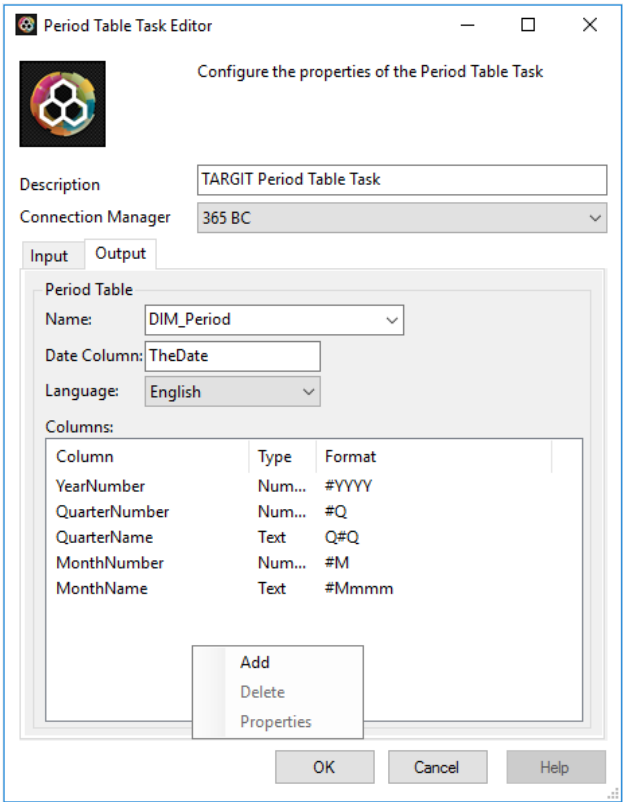
- Specific date: Adds dates from a specific start date. If input columns are selected, unique dates will be added from these, then any earlier dates added according to Span from date - if the span from date is earlier than the lowest input date.
- x Months back: Adds dates in the same way as with specific date, but sets the from date according to x months before today
- Minimum date: Spans dates starting with the earliest date in input columns

#### Span Dates - To

- Specific date: Adds dates up to a specific end date. If input columns are selected, unique dates will be added from these, then later dates added according to Span to date - if the span to date is later than the highest input date.
- x Months ahead: Adds dates in the same way as with specific date, but sets the to date according to x months after today
- Maximum date: Spans dates ending with the latest date in input columns

## Output

In the Output tab, the properties and columns of the period table to be created are defined. Add a table name and the name of the primary date (datetime) column of the completed period table. The language selection defines available naming for e.g. month names.

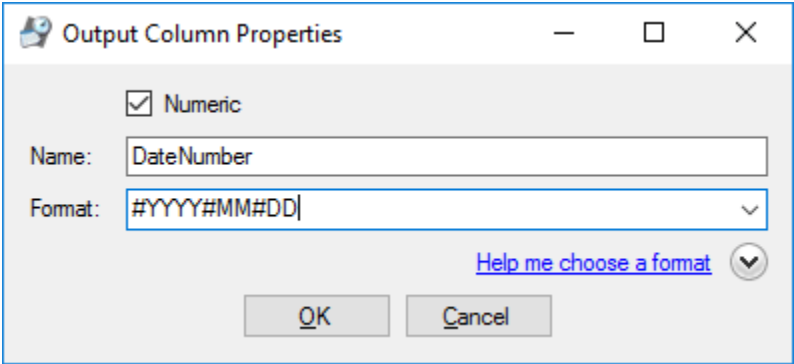


The Period Table Task Editor dialog box is shown. It has a title bar with a close button. The main area is titled "Configure the properties of the Period Table Task". It contains a "Description" field with the text "TARGIT Period Table Task". Below it is a "Connection Manager" dropdown menu showing "365 BC". There are two tabs: "Input" and "Output", with "Output" selected. The "Output" tab contains a "Period Table" section with a "Name" dropdown set to "DIM\_Period", a "Date Column" text box with "TheDate", and a "Language" dropdown set to "English". Below this is a "Columns:" table with the following data:

Column	Type	Format
YearNumber	Num...	#YYYY
QuarterNumber	Num...	#Q
QuarterName	Text	Q#Q
MonthNumber	Num...	#M
MonthName	Text	#Mmmm

Below the table is a context menu with "Add", "Delete", and "Properties" options. At the bottom are "OK", "Cancel", and "Help" buttons.

In Columns, any number of columns can be defined, all based on the list of input dates. A few simple columns are available when the task is first added, but there is no limit to the number of columns. A right click menu is available for managing the columns. Each column has a name, an output type, either Number or Text, and the formatting function describing what to calculate.



The Output Column Properties dialog box is shown. It has a title bar with a close button. It contains a "Numeric" checkbox which is checked. Below it is a "Name" text box with "DateNumber". Below that is a "Format" dropdown menu showing "#YYYY#MM#DD". To the right of the format dropdown is a link that says "Help me choose a format" and a small downward arrow icon. At the bottom are "OK" and "Cancel" buttons.

When a new column is added, a new window with the properties of the column pops up. To help define the column output format, use the format helper function. In the example, a numeric column called **DateNumber** is defined, which will return a full date of the format 20181123. In the **Output column format** box below, the list of available output column formats is shown.

The period table resulting from the example definitions with DateNumber added:

dbo.DIM_Period
Columns
TheDate (datetime, null)
YearNumber (int, null)
QuarterNumber (int, null)
QuarterName (nvarchar(50), null)
MonthNumber (int, null)
MonthName (nvarchar(50), null)
DateNumber (int, null)
Keys
Constraints
Triggers
Indexes
Statistics

	TheDate	YearNumber	QuarterNumber	QuarterName	MonthNumber	MonthName	DateNumb
1	2018-12-31 00:00:00.000	2018	4	Q4	12	December	20181231
2	2018-12-31 23:59:59.000	2018	4	Q4	12	December	20181231
3	2019-01-01 00:00:00.000	2019	1	Q1	1	January	20190101
4	2019-01-02 00:00:00.000	2019	1	Q1	1	January	20190102
5	2019-01-03 00:00:00.000	2019	1	Q1	1	January	20190103
6	2019-01-04 00:00:00.000	2019	1	Q1	1	January	20190104
7	2019-01-05 00:00:00.000	2019	1	Q1	1	January	20190105
8	2019-01-06 00:00:00.000	2019	1	Q1	1	January	20190106
9	2019-01-07 00:00:00.000	2019	1	Q1	1	January	20190107
10	2019-01-08 00:00:00.000	2019	1	Q1	1	January	20190108
11	2019-01-09 00:00:00.000	2019	1	Q1	1	January	20190109
12	2019-01-10 00:00:00.000	2019	1	Q1	1	January	20190110
13	2019-01-11 00:00:00.000	2019	1	Q1	1	January	20190111

## Output column formats

The following identifiers will be substituted during execution of the task. Examples are presented using the date 24th of august, 1970.

```
#YYYY 4 digit year (1970)
#YY 2 digit year (70)
#WYYY 4 digit week year - ISO 8601 (1970)
#WY 2 digit week year - ISO 8601 (70)
#WYYYS 4 digit week year - Simple (1970)
#WYS 2 digit week year - Simple (70)
#Q Quarter (3)
#MMMM Month name in upper case (AUGUST)
#Mmmm Month name in proper case (August)
#mmmm Month name in lower case (august)
#MMM 3 letter month in upper case (AUG)
#Mmm 3 letter month in proper case (Aug)
#mmm 3 letter month in lower case (aug)
#MM 2 digit month number (08)
#M 1/2 digit month number (8)
#WW 2 digit week number - ISO 8601 (35)
#W 1/2 digit week number - ISO 8601 (35)
#WWS 2 digit week number - Simple (35)
#WS 1/2 digit week number - Simple (35)
#DDY 3 digit day of year (236)
#DY 1/2/3 digit day of year (236)
#DDDW Day name in upper case (SATURDAY)
#Dddw Day name in proper case (Saturday)
#dddw Day name in lower case (saturday)
#DDW 3 letter day name in upper case (SAT)
#Ddw 3 letter day name in proper case (Sat)
#ddw 3 letter day name in lower case (sat)
#DW Day of week number (6)
#DD 2 digit day of month (24)
#D 1/2 digit day of month (24)
```

Any identifier can be appended with options that can be used to modify the resulting value. There are two options; date shift (days and/or months) and add/subtract.

The date shift option has the syntax {nD} or {nM}, where n can be any positive or negative integer and denotes the number of days/months that a date is adjusted before the result is calculated. D is days and M is months. If the days and months options are combined, the days option is calculated first.

### Examples:

```
#YYYY{6M} Shift date 6 months forward (1971)
#Mmmm{-4M} Shift date 4 months backward (April)
```

The add/subtract option is applied after the date calculation and is only valid for numeric identifiers, i.e. it will work for #M and #MM, but not #MMM and #MMMM.

### Examples:

```
#YYYY{-1} Subtract 1 year (1969)
#MM{6} Add 6 months (14)
```

The options can be combined in any order.

### Examples:

```
#YYYY{-9M}{1} Shift date 3 months backward and add 1 year (1970)
#YY{-1960}{6M} Shift date 6 months forward and subtract 1960 years (11)
#DD{1M}{7D}{-10} Shift date 7 days forward, then 1 month forward and subtract 10 days (20)
```



- All TARGIT custom components connect to the SQL Server database using an ADO.NET connection
- In later Data Tools (Visual Studio) versions, check the TargetServerVersion property for the SSIS project if the TARGIT components do not show up in the SSIS Toolbox
- If the TARGIT components are not in the SSIS Toolbox after installation, a trouble shooting article is available (see Related articles)
- The TARGIT custom components are available for SQL Server versions beginning at 2005
- The components are free to use, but only officially supported for active TARGIT customers

## Related articles

- [TARGIT Custom Tasks - Problem solving](#)
- [TARGIT Period Table Task](#)
- [TARGIT Quality Check Task](#)
- [TARGIT Quality Check Transformation](#)
- [TARGIT Enumeration Task](#)