Creation of New Fiscal Dimensions (Fiscal Calendar) - The Hitchhiker's Guide

Motivation

GoodData supports date dimensions according to many standard definitions. However, from time to time customers request the creation of custom calendars. This has been quite painful process as almost every customer has a set of specific requirements. To reduce the complexity of the process of implementing fiscal calendars, below you will find a description of the process for implementing fiscal calendars (assuming the customer can generate a properly formatted input file).

Process in a nutshell

- First, check to make sure that the same fiscal calendar has not already been created (URN for our project templates and share dimensions).
- If not, ask the customer to generate an input data with the fiscal calendar requirements (according to this structure: fiscal_calendar.csv). **The important part is: the format of file must be exactly the same.** Otherwise the ETL will fail and your timeline for implementation will likely be delayed.
- Run ETL - https://github.com/gooddata/ms_projects/tree/master/projects/FiscalDimension and send the results to ROLAP (and ask Michael Stencl for the earliest possible release date).

Structure of an input file (to pass ETL expectation)

There is an ETL which prepare objects according to GoodData's date dimension requirements. Usually a customer is supposed to generate the file in this structure. The current version of ETL (https://github.com/gooddata/ms_projects/tree/master/projects/FiscalDimension) is able to read the input file fiscal_calendar.csv.

Field	Description	Format	Example value
dt	A date needed to remap for a fiscal date. The most important field.	yyyy-MM-dd	2000-01-30
day_name	A day name of date in **dt** field.		Sunday
day_of_week	A day of week of date in **dt** field.		1
week_id	An id of week for date in **dt** field.		20005
week_start_date	A date where date in **dt** field starts.	yyyy-MM-dd	2000-01-30

week_end_date	A date where date in **dt** field ends.	yyyy-MM-dd	2000-02-05
week_of_month	A record which tracks a week number in a month for **dt** field.		1
week_of_month_name	An extended record which tracks a week number in a month for **dt** field.		wk1 February 2000
week_of_year	A record which tracks a week number in a year for **dt** field.		1
week_of_year_name	An extended record which tracks a year number in a month for **dt** f ield.		wk1 2000
month_id	An id of month for date in **dt** field .		20002
month_name	Month name + year of date in **dt** field.		February 2000
month_short_name	Month name of date in **dt** field.		February
month_start_date	A date where month in **dt** field starts.	yyyy-MM-dd	2000-02-06
month_end_date	A date where month in **dt** field ends.	yyyy-MM-dd	2000-03-04
weeks_in_month	A number of weeks in a months for date in **dt** field.		4
fiscal_month_of_year	A month number in a year for date in **dt** field.		1
quarter_of_year	A quarter of a year of a date in **d t** field.		1
quarter_name	A quarter name and year of a year of a date in **dt** field.		Q1 2000
quarter_id	A quarter identifier of a year of a date in **dt** field.		20001
quarter_key	A year of a date in **dt** field.	yyyy	2000
year_name	A year of a date in **dt** field.		FY2000
day_of_year	A day of year of a date in **dt** field .		1
day_of_quarter	A day of quarter of a date in **dt** fi eld.		1
day_of_month	A day of month of a date in **dt** fie ld.		1

Structure of GoodData date dimension

Every GoodData date dimension template has a standardized structure (with few exceptions). There are 18 tables that represent each specific date granularity (year, month, week, day, ...) and the objects are connected together. For a reference, please check the following description.

Year (1)

Description: Values for available years.

Field	Type	Format	Example

year_id	integer		2000
descr_default	string		FY2000

Week_in_year (2) / EUWeek_in_year (3)

Description: Values for available weeks in a year. A standard year has usually 52-53 weeks. At the moment values for weeks and euweeks are the same.

Field	Type	Format	Example
id	integer		5
descr_default	string		W5

Week_in_quarter (4) / EUWeek_in_quarter (5)

Description: Values for available weeks in quarters. A standard quarter has usually 12 weeks (3 months * 4 weeks). At the moment values for weeks and euweeks are the same.

Field	Type	Format	Example
id	integer		10
descr_default	string		W10

Week (6) / EUWeek (7)

Description: An object which describes weeks/eu weeks. At the moment values for weeks and euweeks are the same.

Field	Type	Format	Example
week_id	integer		1
descr_week_quarter	string		W1/Q1/2000
descr_from_to	string		Jan 30, 2000 - Feb 05, 2000
descr_default	string		Wk. of Sun 01/30/2000
descr_week_year	string		W1/2000
descr_number	string		W1/2000
descr_week_quarter_cont	string		W1/Q1/2000

Quarter_in_year (8)

Description: Quarters in a year (numbers from 1 to 4).

Field	Type	Format	Example
id	integer		1
descr_default	string		Q1

Quarter (9)

Description: Available quarters in whole fiscal dimension.

Field	Type	Format	Example
quarter_id	integer		1

year_id	integer		2000
quarter_of_year	integer		1
descr_default	string		Q1/2000

Month_in_year (10)

Description: Available months in a year + some more details relevant to each particular record.

Field	Type	Format	Example
id	integer		1
descr_default	string		Jan
desc_mq	string		M1/Q1
desc_num	string		M1
desc_us_long	string		January

Month_in_quarter (11)

Description: Available months in a quarter.

Field	Type	Format	Example
id	integer		1
descr_default	string		M1

Month (12)

Description: Available months and their particular details.

Field	Type	Format	Example
month_id	integer		1
year_id	integer		2000
month_of_year	integer		1
quarter_id	integer		1
quarter_of_year	integer		1
month_of_quarter	integer		1
desc_num	date	MMM yyyy	Jan 2000
desc_us_long	date	M/yyyy	1/2000
descr_default	date		January 2000

Day_in_year (13)

Description: Available days in a year (standard calendar contains 365 - 366 days).

Field	Type	Format	Example
id	integer		1
descr_default	string		D1

Day_in_week (14) / Day_in_euweek (15)

Description: Available days in a week. At the moment values for weeks and euweeks are the same.

Field	Type	Format	Example
id	integer		1
descr_default	string		Sun
desc_num	integer		1
desc_us_long	string		Sunday

Day_in_quarter (16)

Description: Available days in a quarter.

Field	Type	Format	Example
id	integer		1
descr_default	string		D1

Day_in_month (17)

Description: Available days in a month.

Field	Type	Format	Example
id	integer		1
descr_default	string		D1

Day (18)

Description: Available dates + all relevant details (connection with other objects).

Field	Type	Format	Example
id	integer		36554
id_day_in_euweek	integer		1
id_day_in_year	integer		1
id_quarter_in_year	integer		1
id_month_in_quarter	integer		1
id_month_in_year	integer		1
id_week	integer		1
id_euweek	integer		1
id_week_in_year	integer		1
id_day_in_week	integer		1
id_week_in_quarter	integer		20005
id_euweek_in_quarter	integer		20005
id_day_in_quarter	integer		1
id_month	integer		1

id_day_in_month	integer		1
id_year	integer		2000
id_euweek_in_year	integer		1
id_quarter	integer		1
desc_eu	date	MM/dd/yyyy	30/01/2000
descr_default	date	yyyy-MM-dd	2000-01-30
desc_us	date	dd/MM/yyyy	01/30/2000
desc_iso	date	dd-MM-yyyy	30-01-2000
desc_us_long	date	EEE, MMM dd, yyyy	Sun, Jan 30, 2000
desc_us2	date	d/M/yy	30/1/00

Further plans

This version is a simple one. Right now, the output file considers EU and US weeks the same. Radek plans to create a more automated version in future. There will be several available fields in a HTML form and the tool will generate output tables based on these parameters. He plans to discuss this more with engineering/ROLAP.

Useful links

- Latest version of the ETL: https://github.com/gooddata/ms_projects/tree/master/projects/FiscalDimension
- Required input file: fiscal_calendar.csv