MODERN DATA MAGAMENT

BI AND STREAMS REPORT

Kelly Boukouvala
Anna Roumpelaki
Eleni Tsanoula

Athens, March 2018

Scope

The scope of this report is to demonstrate the main steps followed for the creation of a basic data warehouse scheme using a dataset of our choice and some basic visualizations of it . The tools used for this report have been Microsoft SQL Server and Power BI.

Dataset

The source of our dataset has been the kaggle.com site (https://www.kaggle.com/kiva/data-science-for-good-kiva-crowdfunding/data) and we have decided to work on data provided by the Kiva.org i crowdfunding platform.

Kiva.org is an online crowdfunding platform to extend financial services to poor and financially excluded people around the world. Kiva lenders have provided over $1 billion dollars in loans to over 2 million people. In order to set investment priorities, help inform lenders, and understand their target communities, knowing the level of poverty of each borrower is critical.

Kiva has provided a dataset of loans issued over the last two years, and participants are invited to use this data as well as source external public datasets to help Kiva build models for assessing borrower welfare levels.

Metadata

Through the kaggle.com the following basic tables have been provided:

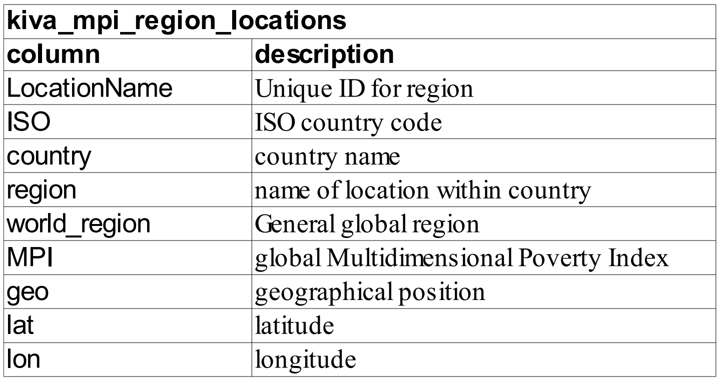
-Kiva_loans

-Kiva_mpi_region_locations

-Loan_theme_ids

-Loan_themes_by_region

Below you will find the layout of the 4 tables and a short description of their fields.



kiva_mpi_region_locations	
column	**description**
LocationName	Unique ID for region
ISO	ISO country code
country	country name
region	name of location within country
world_region	General global region
MPI	global Multidimensional Poverty Index
geo	geographical position
lat	latitude
lon	longitude

kiva_loans

column	**description**
id	Unique ID for loan
funded_amount	Dollar value of loan funded on Kiva.org
loan_amount	Total dollar amount of loan
activity	Loan activity type
sector	Sector of loan activity as shown to lenders
use	text description of how loan will be used
country_code	country ISO code
country	country name
region	name of location within country
currency	currency in which loan is disbursed
partner_id	Unique ID for field partners
posted_time	date and time when loan was posted on kiva.org
disbursed_time	date and time that the borrower received the loan
funded_time	date and time at which loan was fully funded on kiva.org
term_in_months	number of months over which loan was scheduled to be paid back
lender_count	number of lenders contributing to loan
tags	tags visible to lenders describing loan type
borrower_genders	gender of borrower(s)
repayment_interval	frequency at which lenders are scheduled to receive installments
date	date on which loan was posted

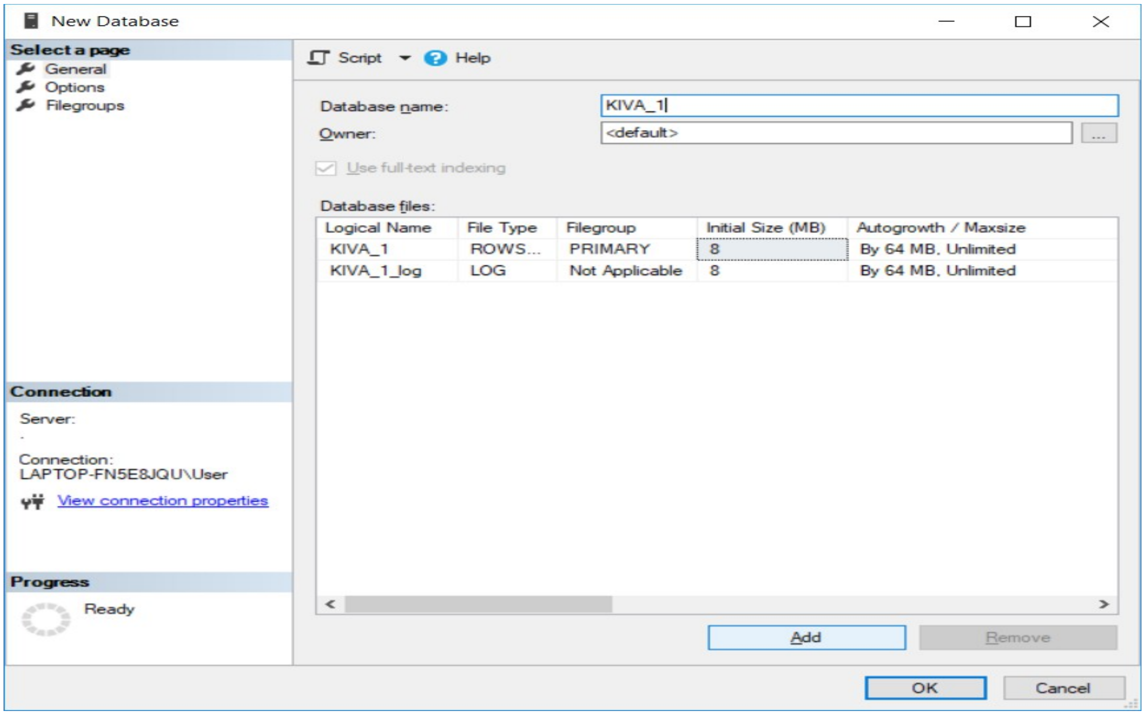
loan_theme_ids

column	**description**
id	Unique ID for loan (Loan ID)
Loan Theme ID	Unique ID for loan theme
Loan Theme Type	General description of the loan theme category
Partner ID	Unique ID for field partners (Partner ID)

loan_themes_by_region	
column	**description**
Partner ID	Unique ID for field partners
Field Partner Name	Name of Field Partner
sector	Sector in which a loan is placed on Kiva's main page
Loan Theme ID	Unique ID for loan theme
Loan Theme Type	General description of the loan theme category
country	country name
forkiva	Was this loan theme created specifically for Kiva?
region	Region within country
geocode_old	Kiva's old geocoding system Lots of missing values
ISO	country ISO code
number	Number of loans funded in this LocationName and this loan theme
amount	Dollar value of loans funded in this LocationName and this loan theme
LocationName	country and region name
geocode	geographical position
names	All placenames that the Gmaps API associates with LocationName
geo	geographical position
lat	latitude
lon	longitude
mpi_region	MPI Region where we think this loan theme is located
mpi_geo	mpi geographical position
rural_pct	The percentage of this field partners' borrowers that are in rural areas

Microsoft SQL Server : Database creation and schema

Below you will find the basic steps we followed for the import of the data in a new database scheme called Kiva and the creation of the corresponding tables.



Database name: KIVA_1

Owner: <default>

☑ Use full-text indexing

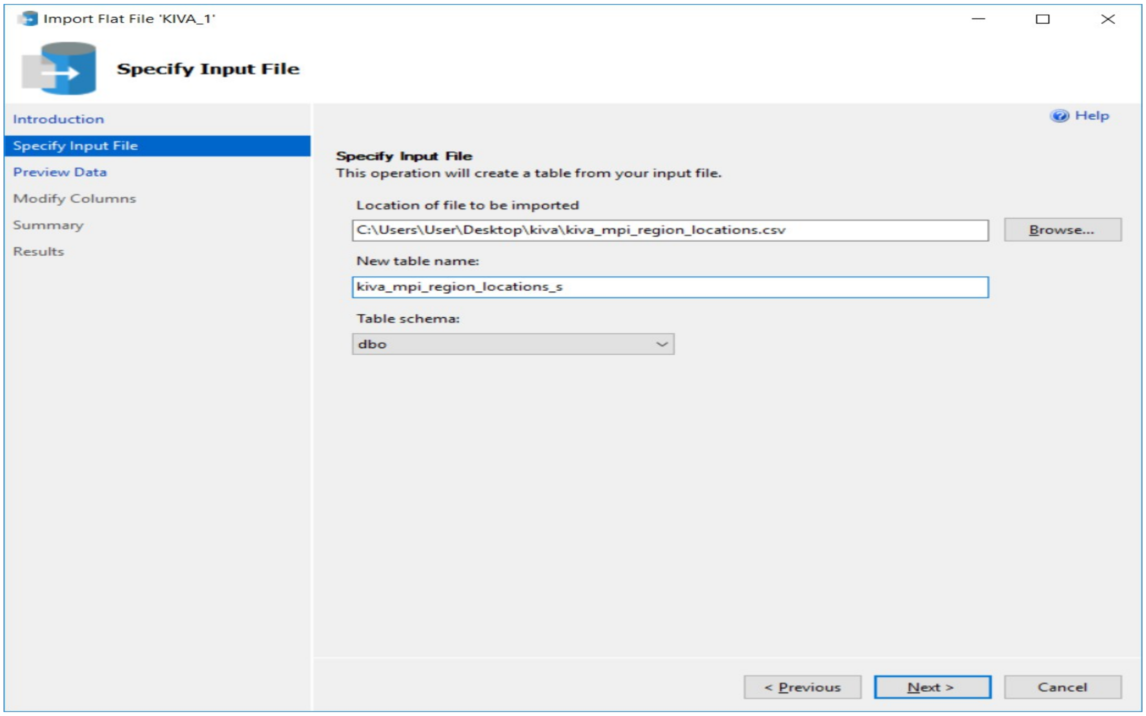
Database files:

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth / Maxsize
KIVA_1	ROWS...	PRIMARY	8	By 64 MB, Unlimited
KIVA_1_log	LOG	Not Applicable	8	By 64 MB, Unlimited

Server:
.

Connection:
LAPTOP-FN5E8JQU\User

Ready



Specify Input File

This operation will create a table from your input file.

Location of file to be imported

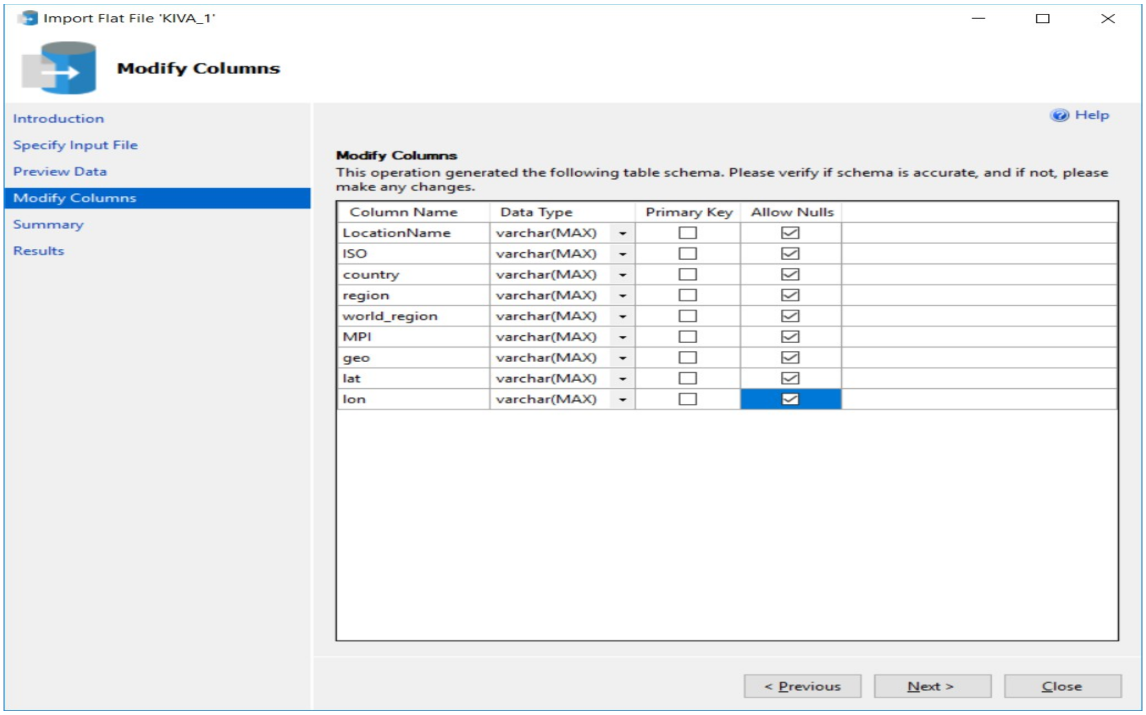
C:\Users\User\Desktop\kiva\kiva_mpi_region_locations.csv

New table name:

kiva_mpi_region_locations_s

Table schema:

dbo

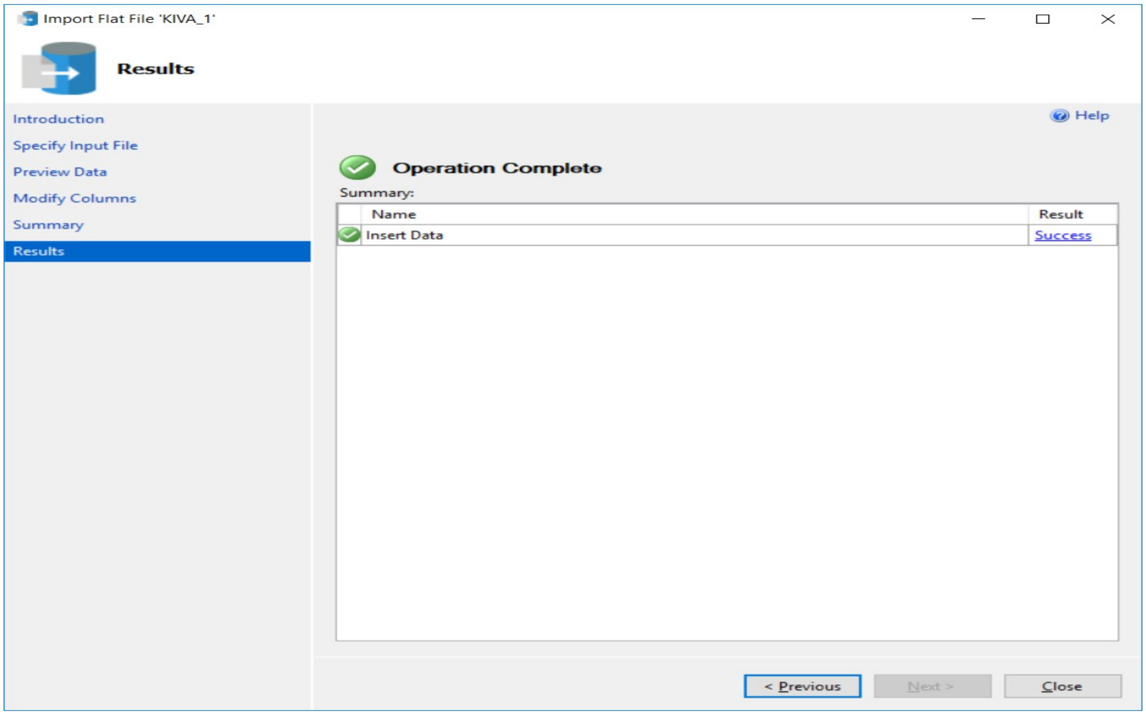


Modify Columns

Modify Columns

This operation generated the following table schema. Please verify if schema is accurate, and if not, please make any changes.

Column Name	Data Type	Primary Key	Allow Nulls
LocationName	varchar(MAX)	☐	☑
ISO	varchar(MAX)	☐	☑
country	varchar(MAX)	☐	☑
region	varchar(MAX)	☐	☑
world_region	varchar(MAX)	☐	☑
MPI	varchar(MAX)	☐	☑
geo	varchar(MAX)	☐	☑
lat	varchar(MAX)	☐	☑
lon	varchar(MAX)	☐	☑

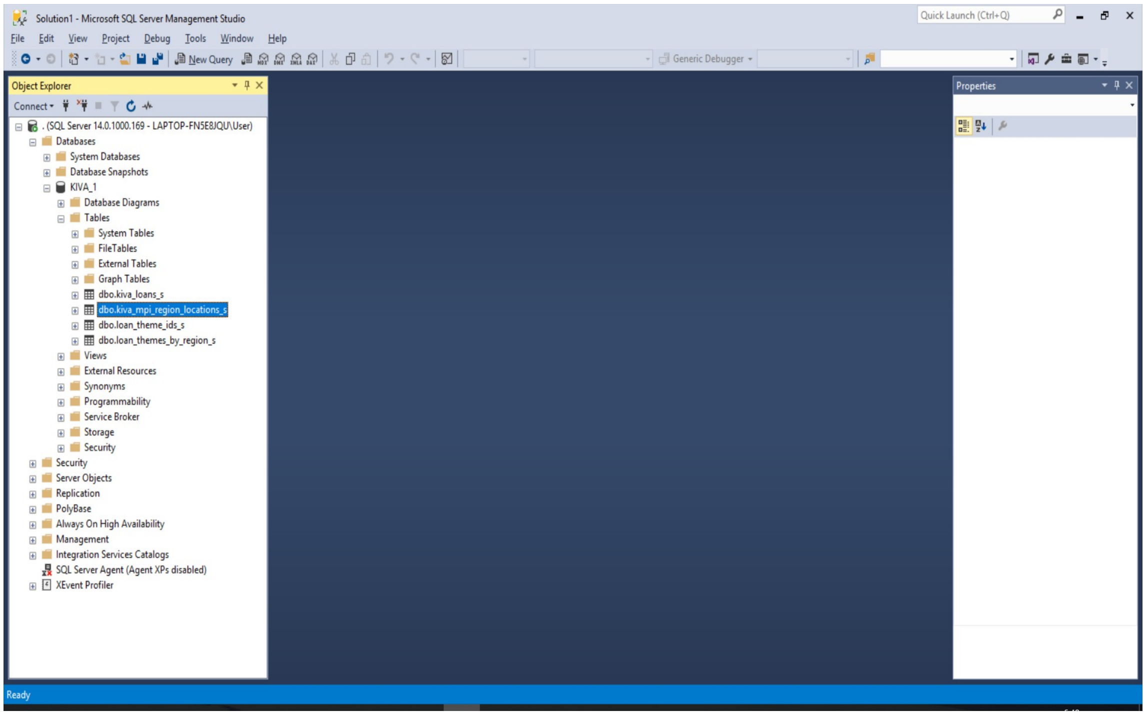


Results

Operation Complete

Summary:

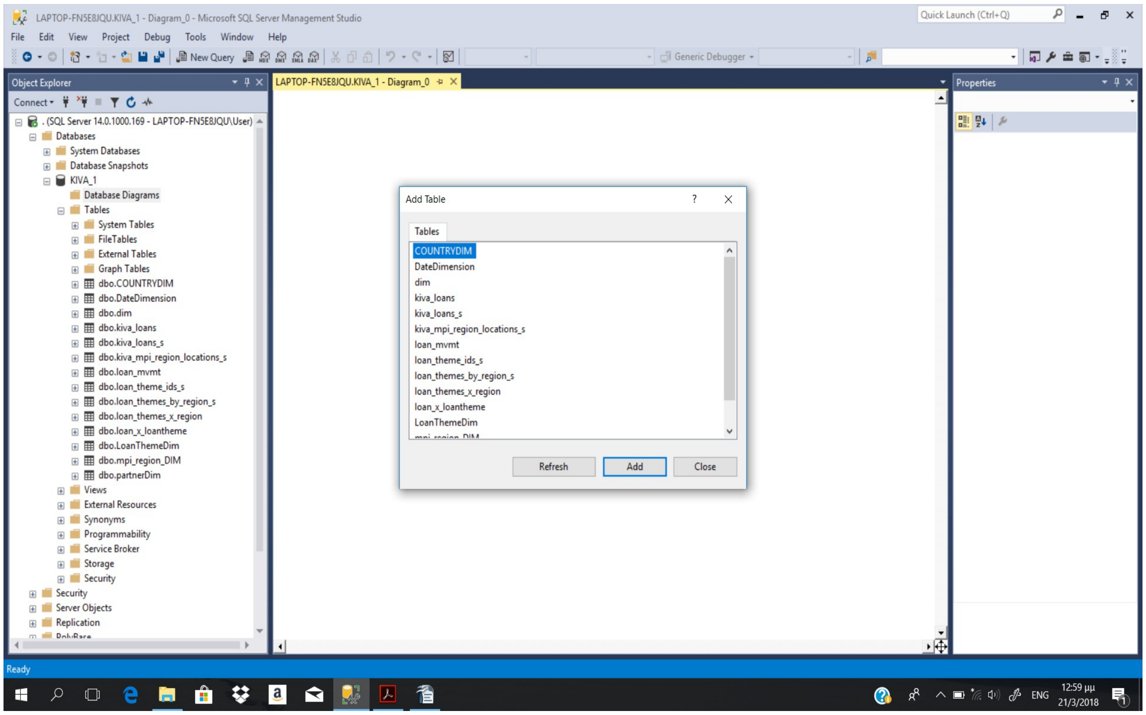
Name	Result
Insert Data	Success

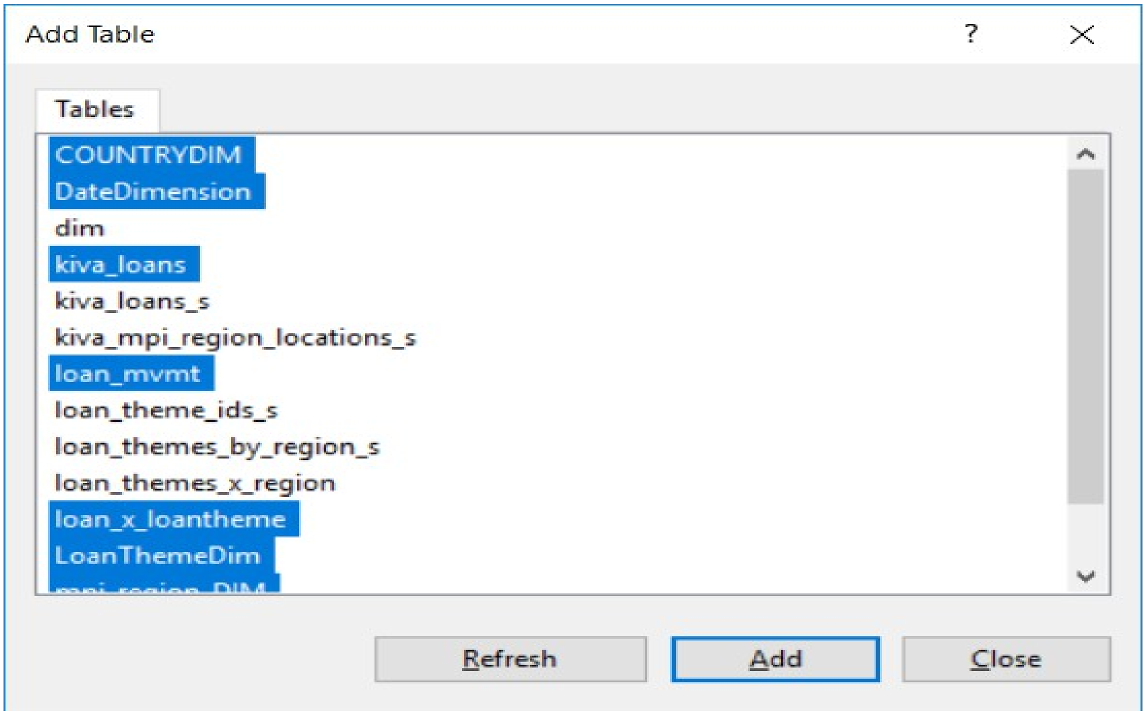


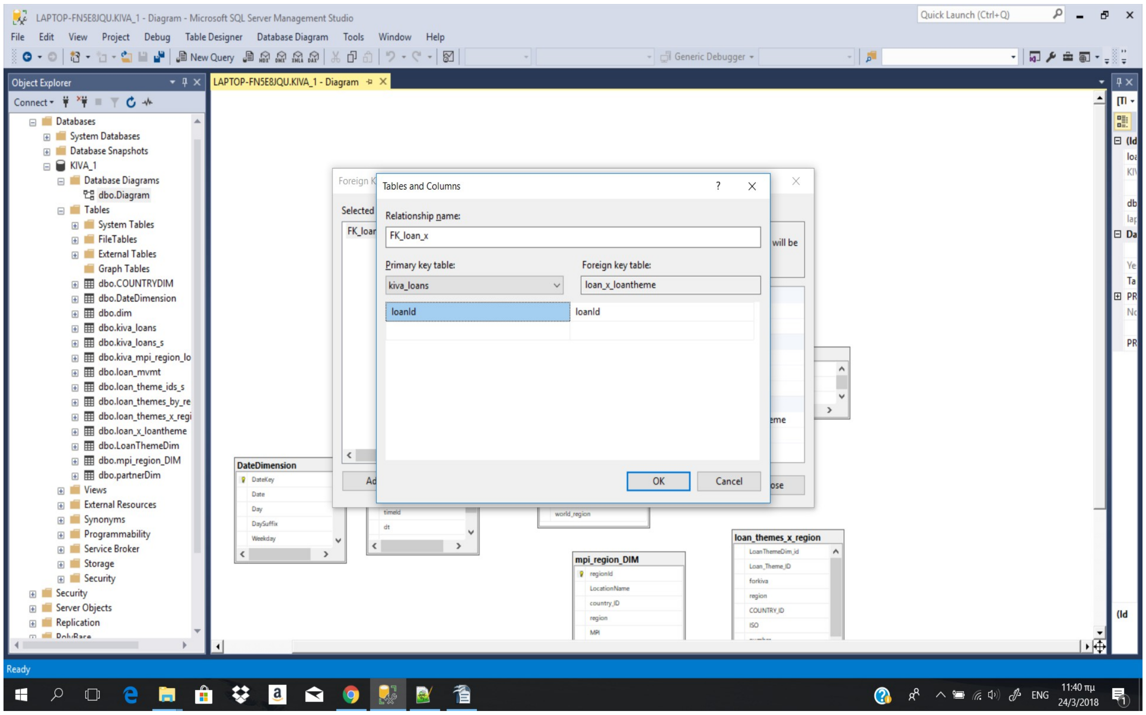
Based on the above mentioned implementation the following **dimensions** came out:
-Kiva_loans (loan dimension)
-Countrydim (country dimension)
-Mpi_region_DIM (region dimension based on the Global Multidimensional Poverty Index)
-PartnerDim (partner dimension)
-LoanThemeDim (loan theme dimension)

In order to facilitate our workflow we have also created a time/date **dimension** , the DateDimension.
The **fact table of our schema** is a new table we have created , the loan_mvmt table which includes the amounts and movements related to the loans requested and funded.
The bridge tables of our snowflake schema are the below:
-loan_x_loantheme (correlation between loans and loan themes)
-loan_themes_x_region (correlation between loan themes and MPI of each region).

The steps we have followed in order to create the fact and bridge tables are described through the following screenshots.


LAPTOP-FN5E8JQU.KIVA_1 - Diagram_0 - Microsoft SQL Server Management Studio
Quick Launch (Ctrl+Q)
File Edit View Project Debug Tools Window Help
New Query
Generic Debugger
Object Explorer
Connect
. (SQL Server 14.0.1000.169 - LAPTOP-FN5E8JQU\User)
Databases
System Databases
Database Snapshots
KIVA_1
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.COUNTRYDIM
dbo.DateDimension
dbo.dim
dbo.kiva_loans
dbo.kiva_loans_s
dbo.kiva_mpi_region_locations_s
dbo.loan_mvmt
dbo.loan_theme_ids_s
dbo.loan_themes_by_region_s
dbo.loan_themes_x_region
dbo.loan_x_loantheme
dbo.LoanThemeDim
dbo.mpi_region_DIM
dbo.partnerDim
Views
External Resources
Synonyms
Programmability
Service Broker
Storage
Security
Security
Server Objects
Replication
PolyBase
LAPTOP-FN5E8JQU.KIVA_1 - Diagram_0
Properties
Add Table
Tables
COUNTRYDIM
DateDimension
dim
kiva_loans
kiva_loans_s
kiva_mpi_region_locations_s
loan_mvmt
loan_theme_ids_s
loan_themes_by_region_s
loan_themes_x_region
loan_x_loantheme
LoanThemeDim
mpi_region_DIM
Refresh
Add
Close
Ready
ENG
12:59 μμ
21/3/2018


Add Table
?
Tables
COUNTRYDIM
DateDimension
dim
kiva_loans
kiva_loans_s
kiva_mpi_region_locations_s
loan_mvmt
loan_theme_ids_s
loan_themes_by_region_s
loan_themes_x_region
loan_x_loantheme
LoanThemeDim
mpi_region_DIM
Refresh
Add
Close



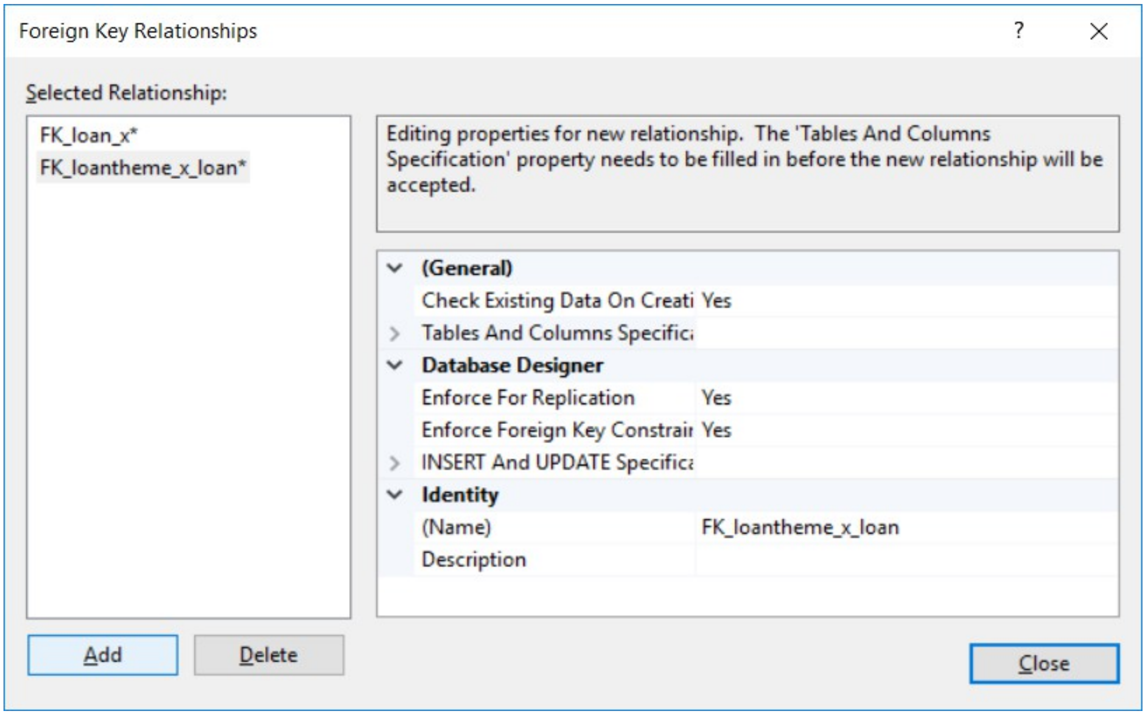
Tables and Columns

Relationship name:

FK_loan_x

Primary key table:	Foreign key table:
kiva_loans	loan_x_loantheme
loanId	loanId

OK Cancel



Foreign Key Relationships

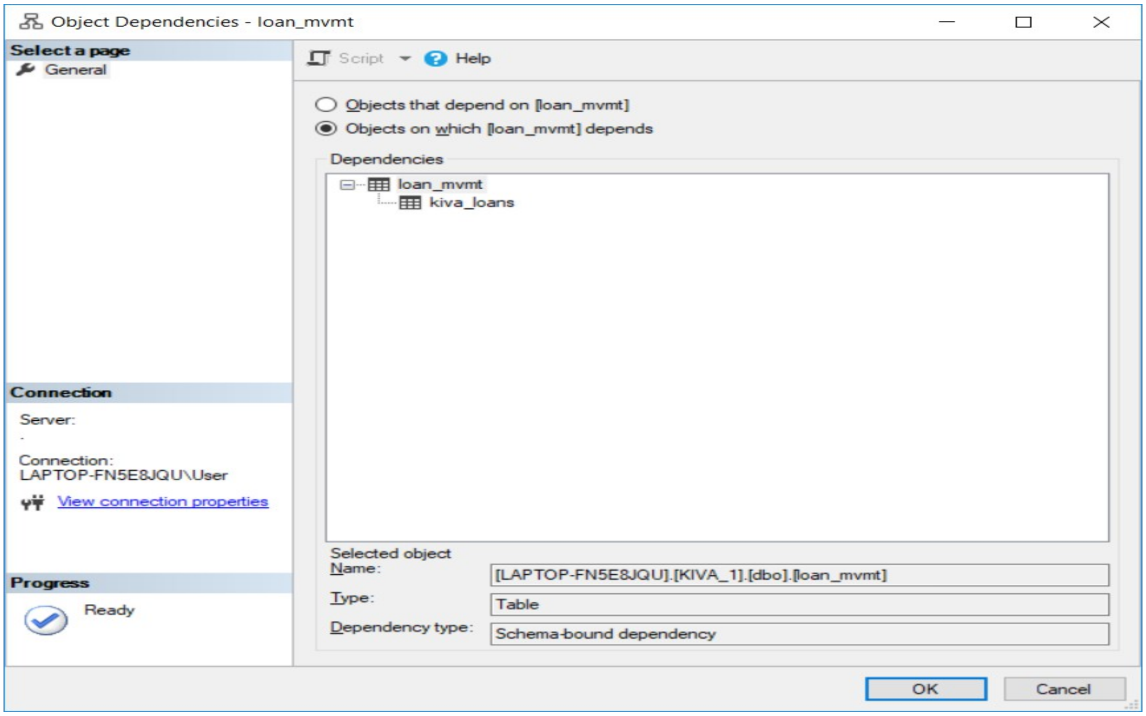
Selected Relationship:

FK_loan_x*
FK_loantheme_x_loan*

Editing properties for new relationship. The 'Tables And Columns Specification' property needs to be filled in before the new relationship will be accepted.

(General)	
Check Existing Data On Creati	Yes
Tables And Columns Specifica	
Database Designer	
Enforce For Replication	Yes
Enforce Foreign Key Constrair	Yes
INSERT And UPDATE Specifica	
Identity	
(Name)	FK_loantheme_x_loan
Description	

Add Delete Close



Object Dependencies - loan_mvmt

○ Objects that depend on [loan_mvmt]
● Objects on which [loan_mvmt] depends

Dependencies

- loan_mvmt
  - kiva_loans

Selected object

Name: [LAPTOP-FN5E8JQU].[KIVA_1].[dbo].[loan_mvmt]

Type: Table

Dependency type: Schema-bound dependency

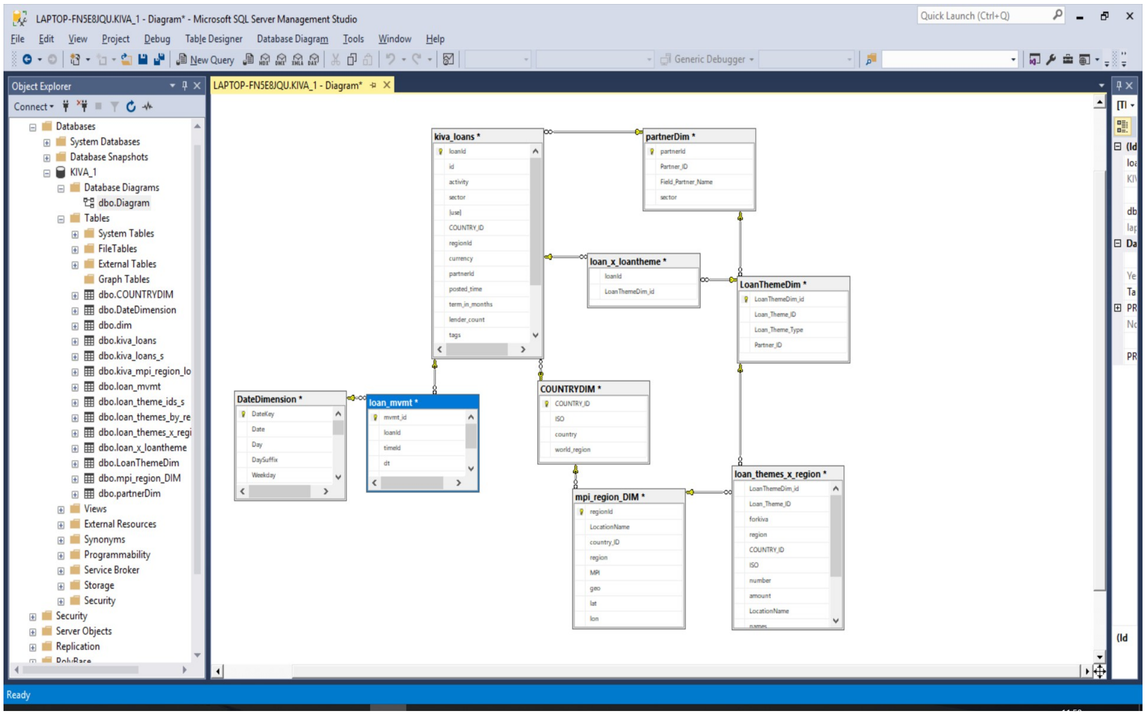
Connection

Server:
.

Connection:
LAPTOP-FN5E8JQU\User

Progress

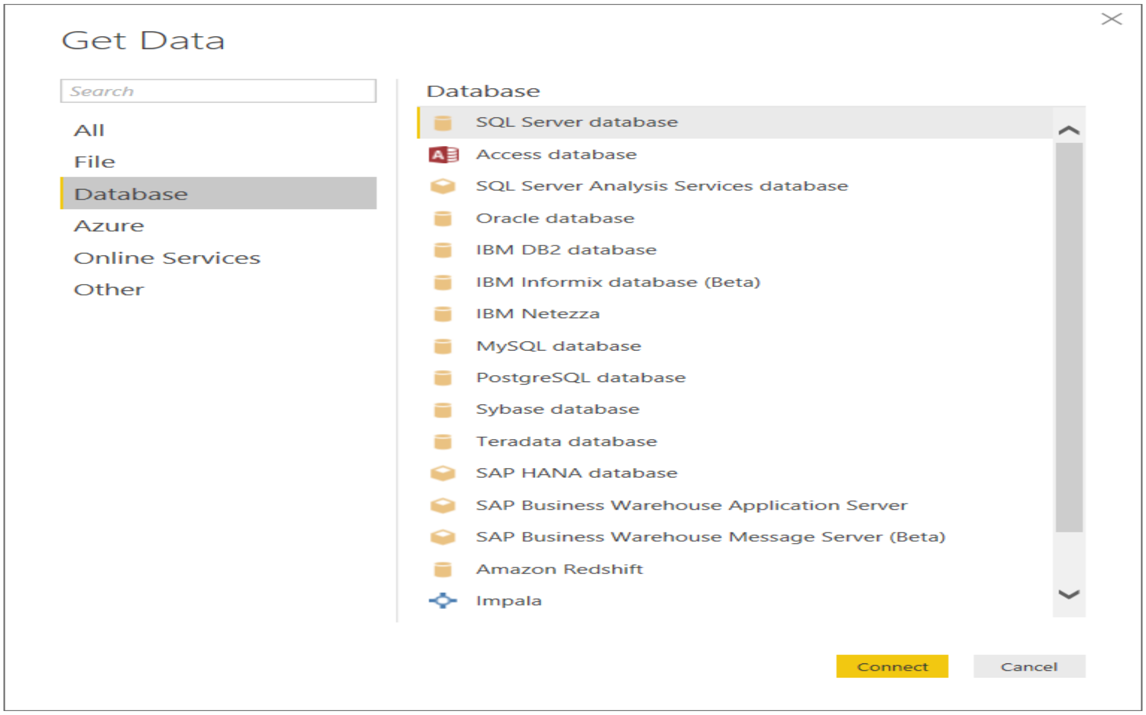
Ready

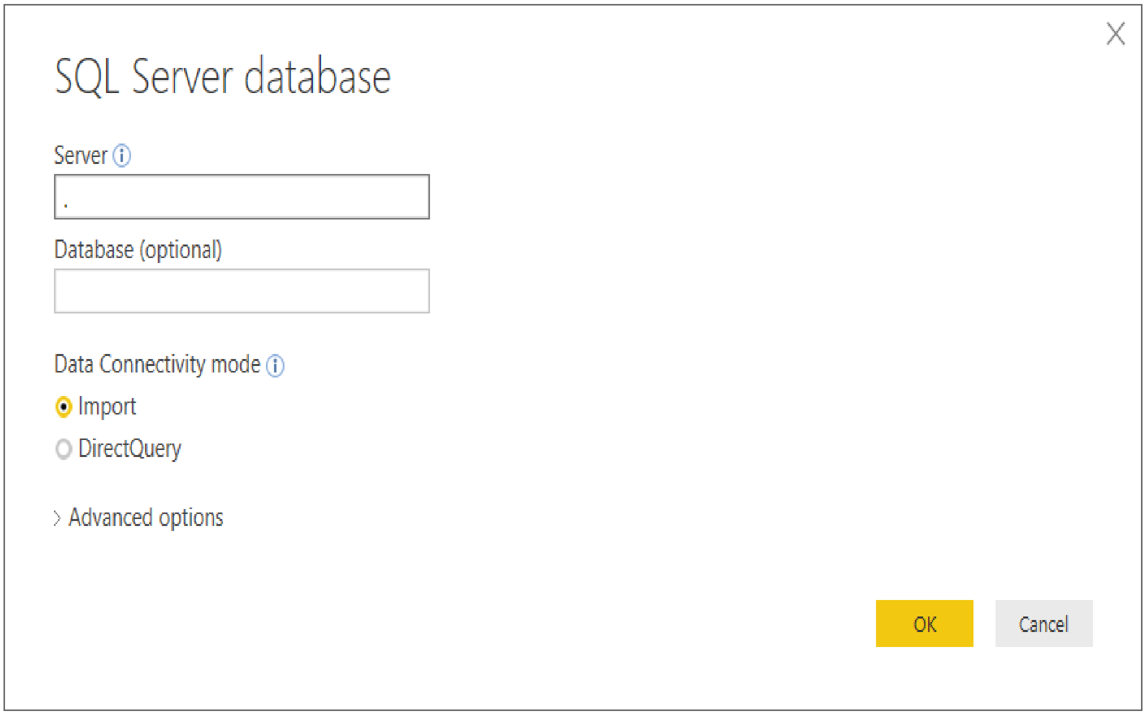


LAPTOP-FN5E8JQU.KIVA_1 - Diagram* - Microsoft SQL Server Management Studio

Power BI: Visualizations

In order to import our data from the SQL Server database to the Power BI application we have followed the steps below:




SQL Server database

Server ⓘ

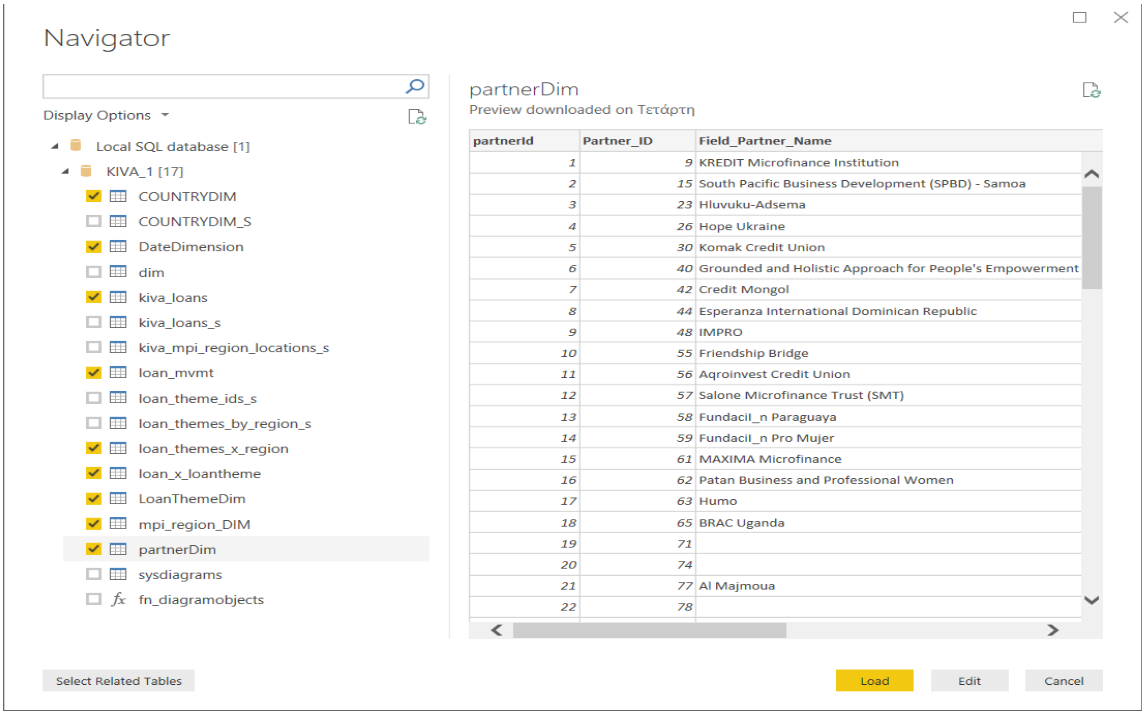
.

Database (optional)

Data Connectivity mode ⓘ

● Import

○ DirectQuery



Navigator

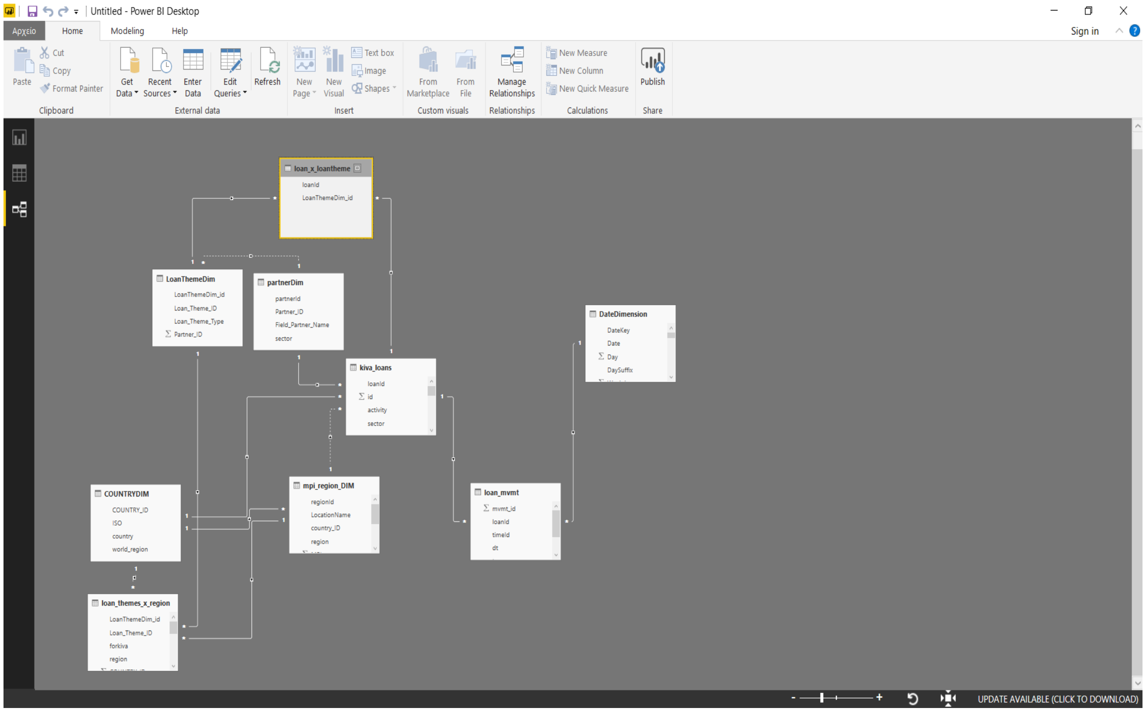
Display Options

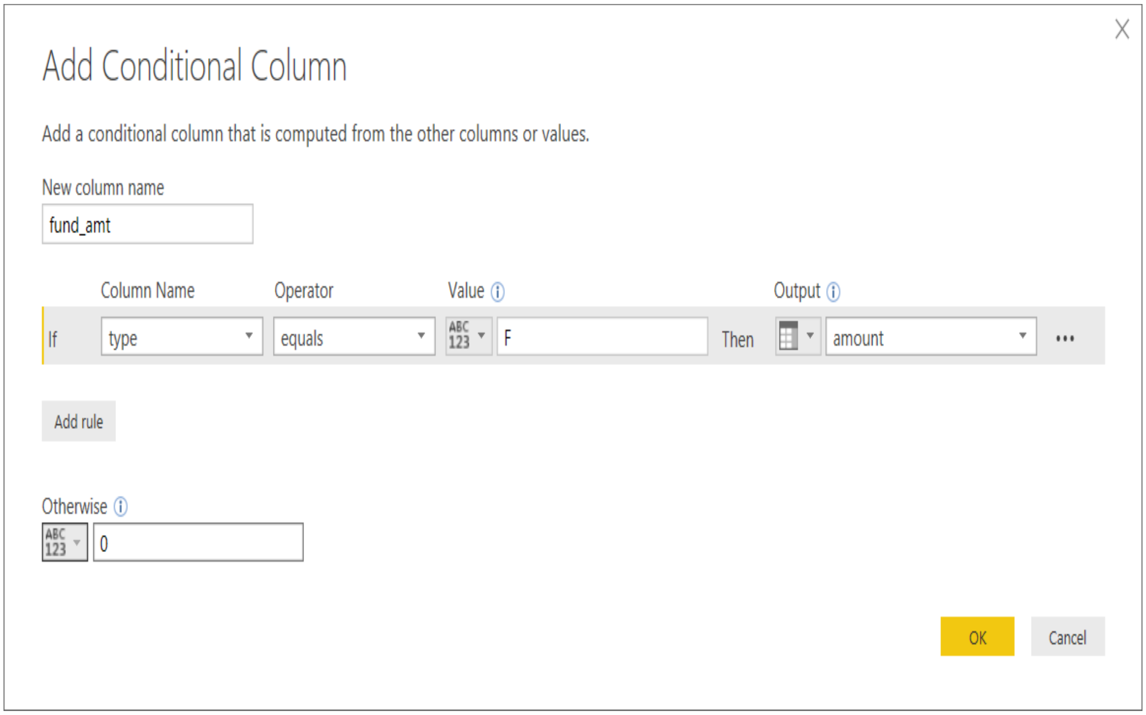
- Local SQL database [1]
  - KIVA_1 [17]
    - ☑ COUNTRYDIM
    - □ COUNTRYDIM_S
    - ☑ DateDimension
    - □ dim
    - ☑ kiva_loans
    - □ kiva_loans_s
    - □ kiva_mpi_region_locations_s
    - ☑ loan_mvmt
    - □ loan_theme_ids_s
    - □ loan_themes_by_region_s
    - ☑ loan_themes_x_region
    - ☑ loan_x_loantheme
    - ☑ LoanThemeDim
    - ☑ mpi_region_DIM
    - ☑ partnerDim
    - □ sysdiagrams
    - □ fn_diagramobjects

partnerDim

Preview downloaded on Τετάρτη

partnerId	Partner_ID	Field_Partner_Name
1	9	KREDIT Microfinance Institution
2	15	South Pacific Business Development (SPBD) - Samoa
3	23	Hluvuku-Adsema
4	26	Hope Ukraine
5	30	Komak Credit Union
6	40	Grounded and Holistic Approach for People's Empowerment
7	42	Credit Mongol
8	44	Esperanza International Dominican Republic
9	48	IMPRO
10	55	Friendship Bridge
11	56	Aqroinvest Credit Union
12	57	Salone Microfinance Trust (SMT)
13	58	Fundacil_n Paraguaya
14	59	Fundacil_n Pro Mujer
15	61	MAXIMA Microfinance
16	62	Patan Business and Professional Women
17	63	Humo
18	65	BRAC Uganda
19	71	
20	74	
21	77	Al Majmoua
22	78	





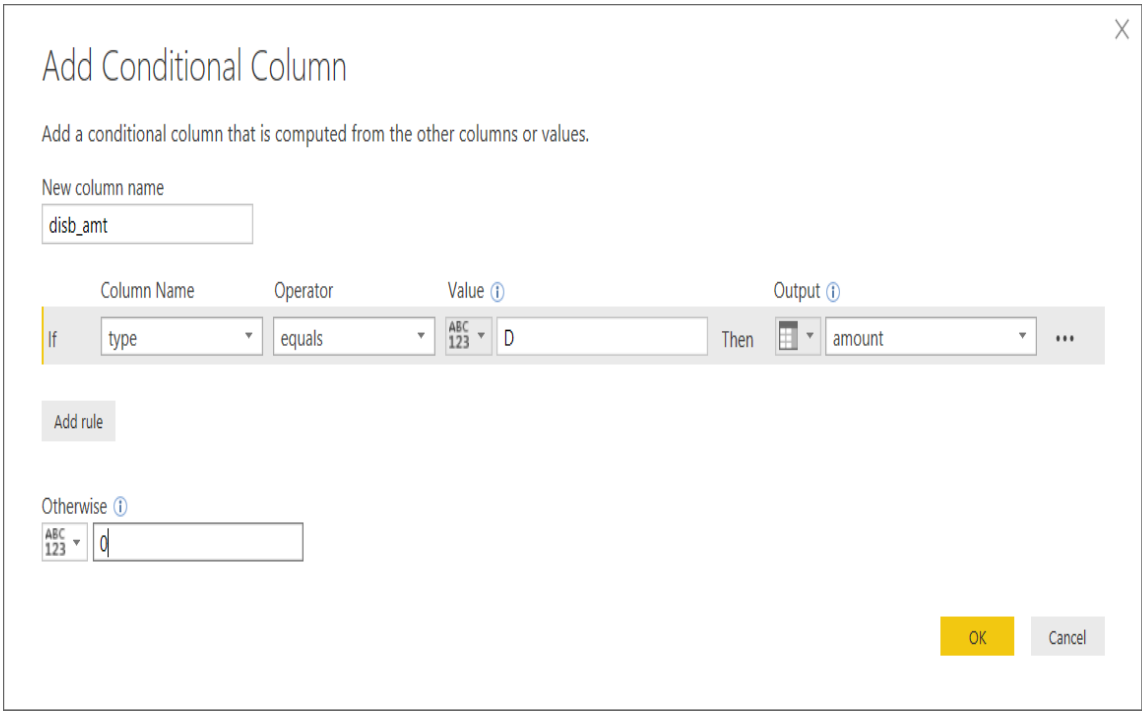
Add Conditional Column

Add a conditional column that is computed from the other columns or values.

New column name: fund_amt

	Column Name	Operator	Value		Output
If	type	equals	F	Then	amount

Otherwise: 0



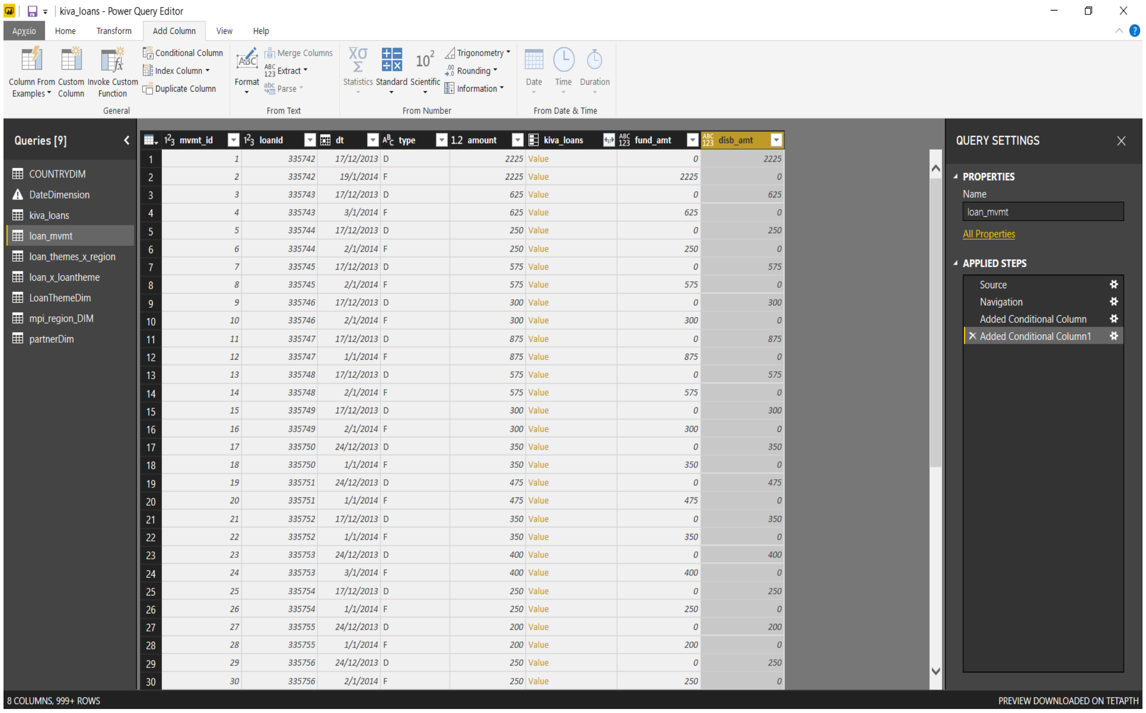
Add Conditional Column

Add a conditional column that is computed from the other columns or values.

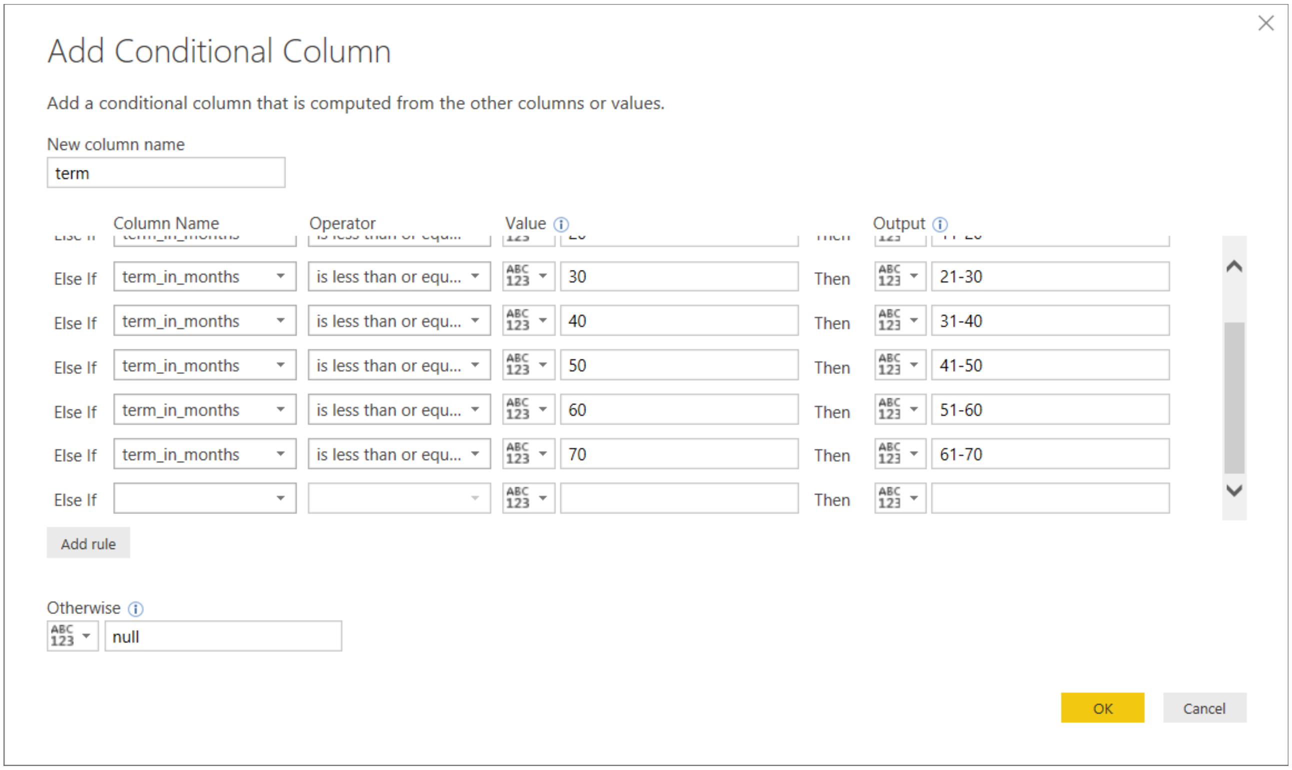
New column name: disb_amt

	Column Name	Operator	Value		Output
If	type	equals	D	Then	amount

Otherwise: 0



	mvmt_id	loanId	dt	type	amount	kiva_loans	fund_amt	disb_amt
1	1	335742	17/12/2013	D	2225	Value	0	2225
2	2	335742	19/1/2014	F	2225	Value	2225	0
3	3	335743	17/12/2013	D	625	Value	0	625
4	4	335743	3/1/2014	F	625	Value	625	0
5	5	335744	17/12/2013	D	250	Value	0	250
6	6	335744	2/1/2014	F	250	Value	250	0
7	7	335745	17/12/2013	D	575	Value	0	575
8	8	335745	2/1/2014	F	575	Value	575	0
9	9	335746	17/12/2013	D	300	Value	0	300
10	10	335746	2/1/2014	F	300	Value	300	0
11	11	335747	17/12/2013	D	875	Value	0	875
12	12	335747	1/1/2014	F	875	Value	875	0
13	13	335748	17/12/2013	D	575	Value	0	575
14	14	335748	2/1/2014	F	575	Value	575	0
15	15	335749	17/12/2013	D	300	Value	0	300
16	16	335749	2/1/2014	F	300	Value	300	0
17	17	335750	24/12/2013	D	350	Value	0	350
18	18	335750	1/1/2014	F	350	Value	350	0
19	19	335751	24/12/2013	D	475	Value	0	475
20	20	335751	1/1/2014	F	475	Value	475	0
21	21	335752	17/12/2013	D	350	Value	0	350
22	22	335752	1/1/2014	F	350	Value	350	0
23	23	335753	24/12/2013	D	400	Value	0	400
24	24	335753	3/1/2014	F	400	Value	400	0
25	25	335754	17/12/2013	D	250	Value	0	250
26	26	335754	1/1/2014	F	250	Value	250	0
27	27	335755	24/12/2013	D	200	Value	0	200
28	28	335755	1/1/2014	F	200	Value	200	0
29	29	335756	24/12/2013	D	250	Value	0	250
30	30	335756	2/1/2014	F	250	Value	250	0



Add Conditional Column

Add a conditional column that is computed from the other columns or values.

New column name: term

	Column Name	Operator	Value		Output
Else If	term_in_months	is less than or equ...	30	Then	21-30
Else If	term_in_months	is less than or equ...	40	Then	31-40
Else If	term_in_months	is less than or equ...	50	Then	41-50
Else If	term_in_months	is less than or equ...	60	Then	51-60
Else If	term_in_months	is less than or equ...	70	Then	61-70
Else If				Then	

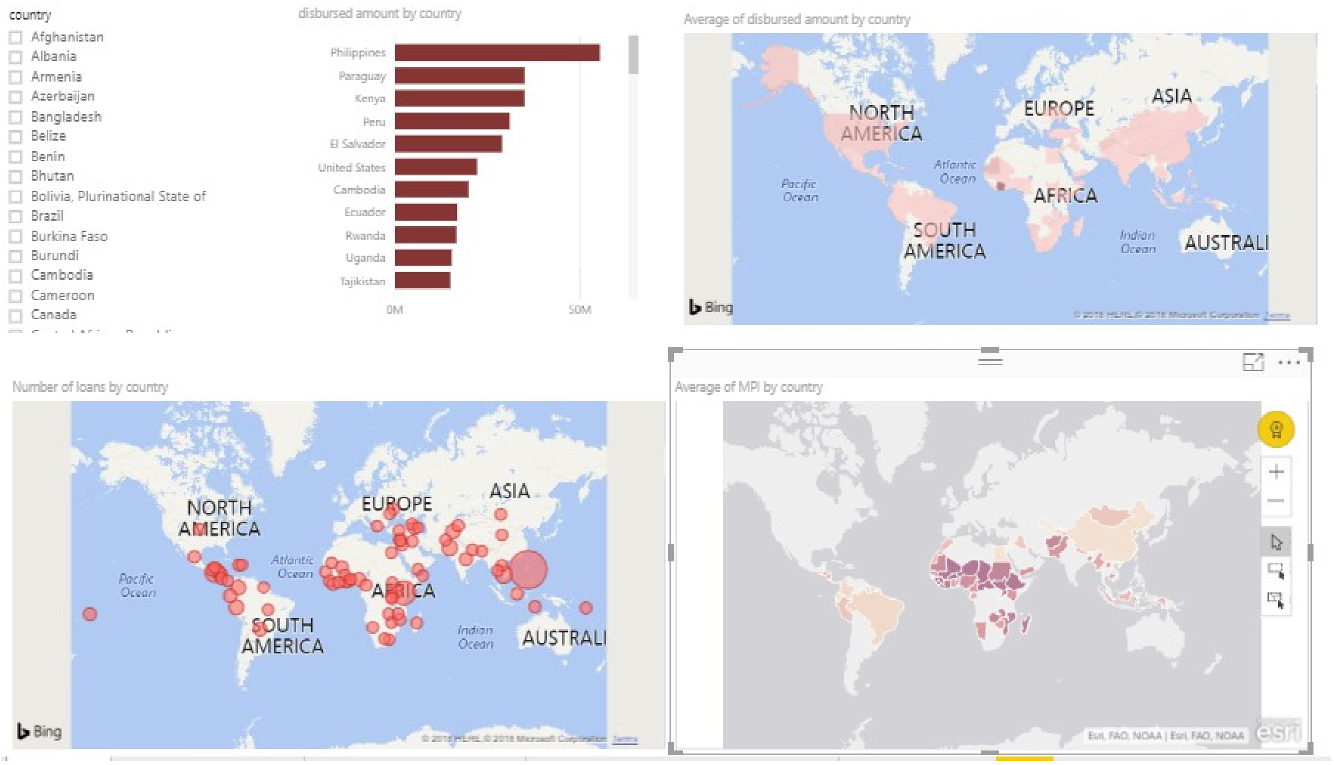
Otherwise: null

After processing the data and creating conditional columns, our final outcome is depicted below, through the following visualizations.

We created four basic categories of visualizations, and focused on the dimension of countries, by creating a slicer that is synchronized among all categories:

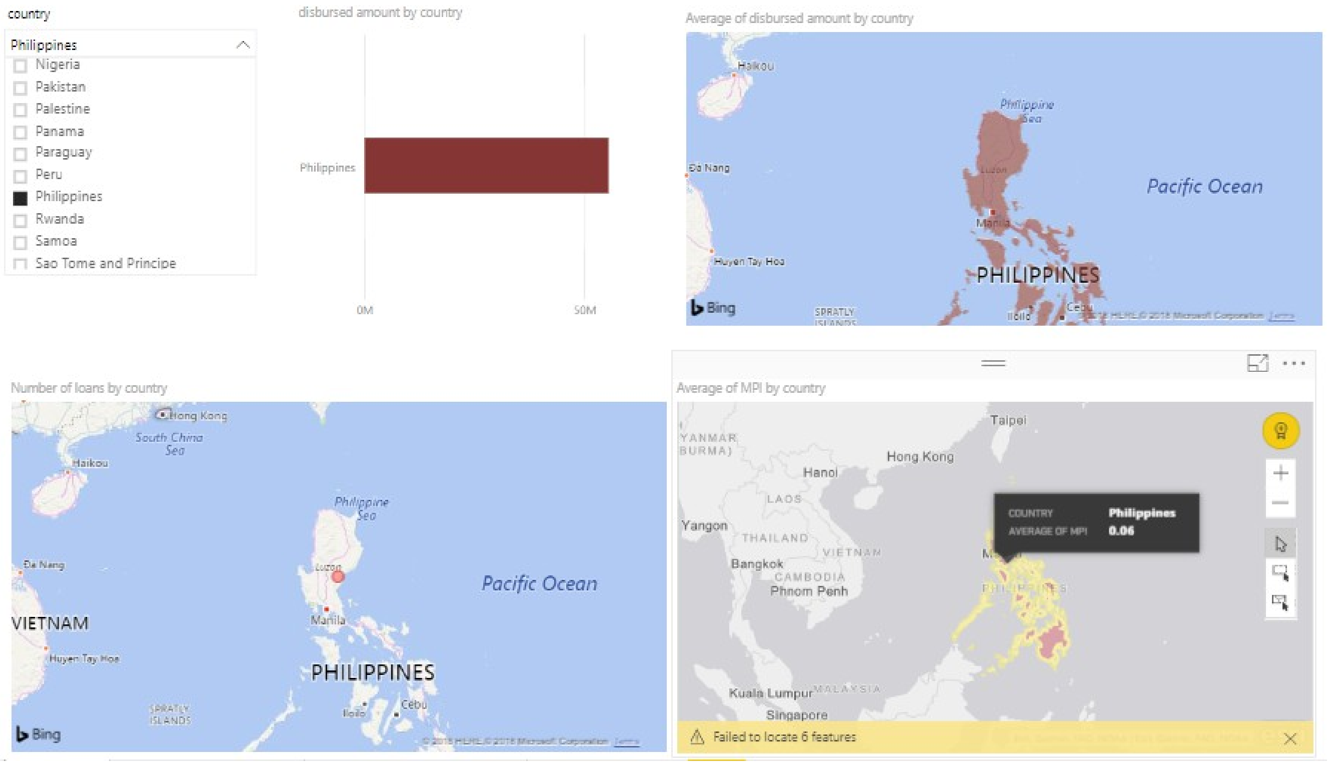
- Regions and Maps
- Loan distribution
- Loan amount distribution
- Loan themes

Regions and Maps:



The purpose of the grouping was to be able to see at first glance what is the connection between the sum of disbursed amount by country, the average loan amount per country, the quantity of loans funded and the Multidimensional Poverty Index (MPI).
We see that Philippines is the country that was given the largest amount, thus we decided to show page filtering by using this country as an example.

The image below shows the effect of applying the filter on Philippines.



country

Philippines

- □ Nigeria
- □ Pakistan
- □ Palestine
- □ Panama
- □ Paraguay
- □ Peru
- ■ Philippines
- □ Rwanda
- □ Samoa
- □ Sao Tome and Principe

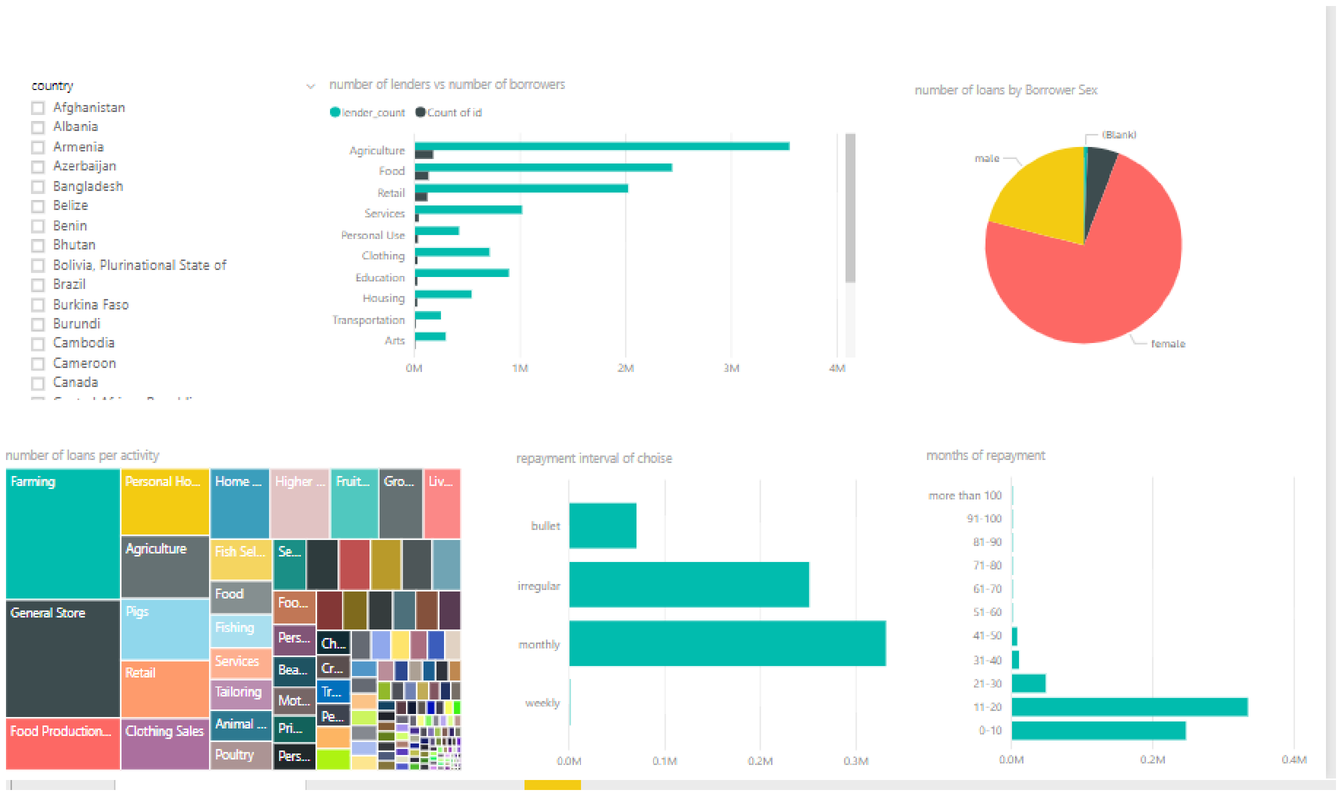
COUNTRY Philippines

AVERAGE OF MPI 0.06

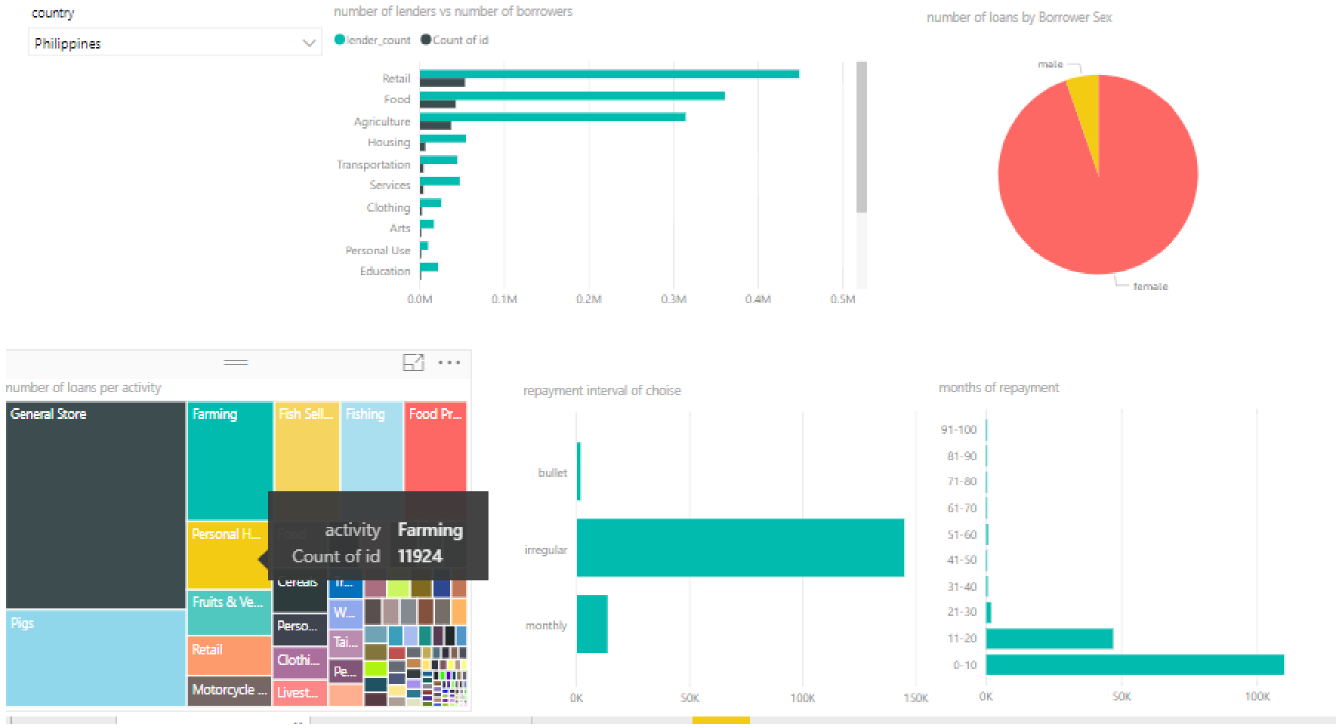
⚠ Failed to locate 6 features

Loan Distribution:

The purpose of this grouping is to depict the main loan characteristics, i.e.activities for the loan application, the gender of applicants, the repayment type and duration distributions, and finally, the big difference between the number of lenders and number of borrowers.

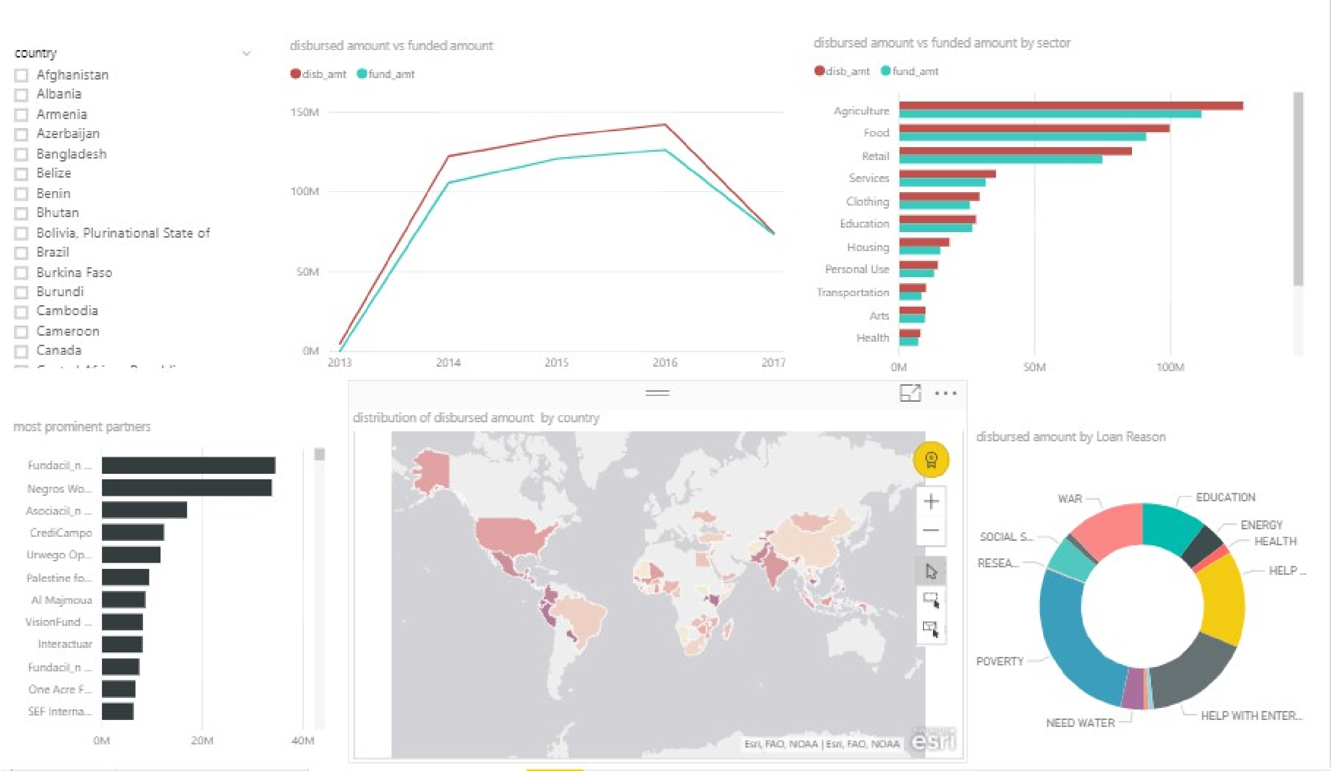


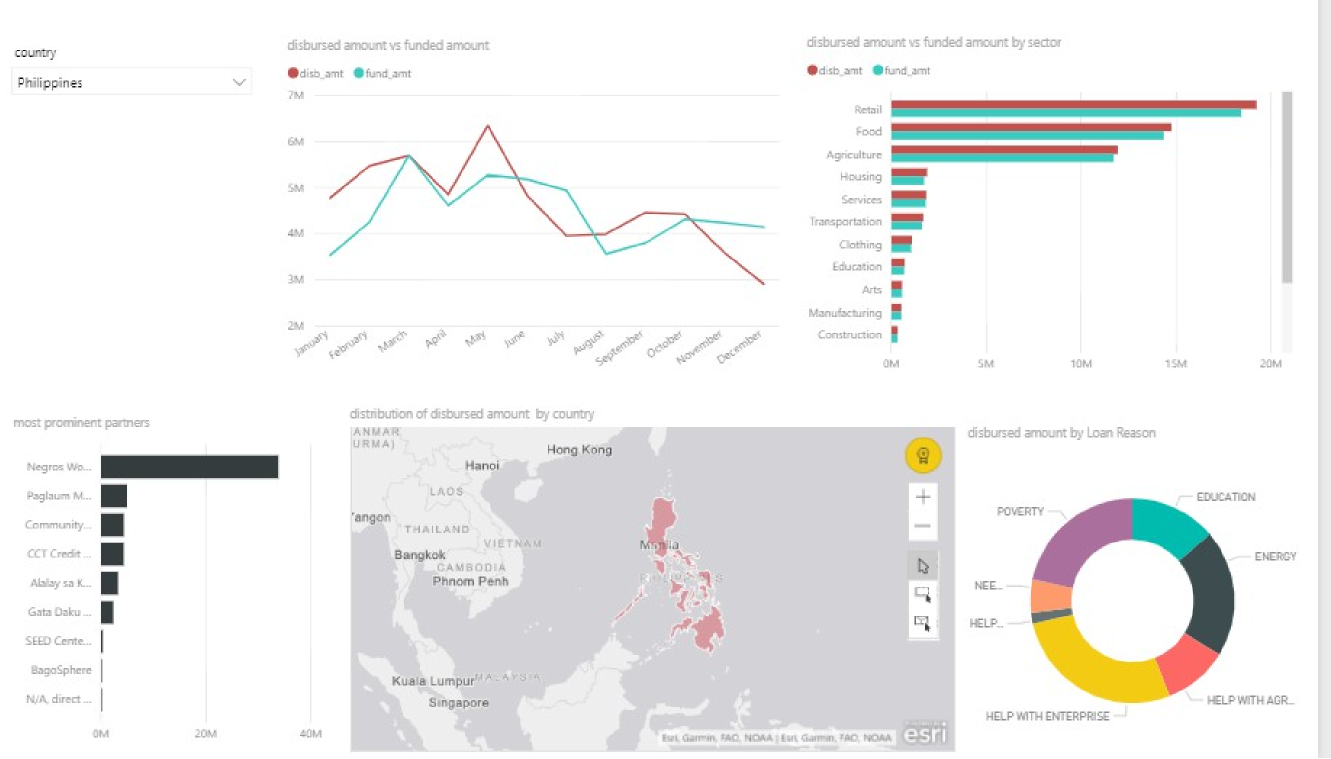
Once again, we applied the Philippines country filter.
We can see that the results concerning distinct countries, can be dramatically different from the average results.



Amount distribution:

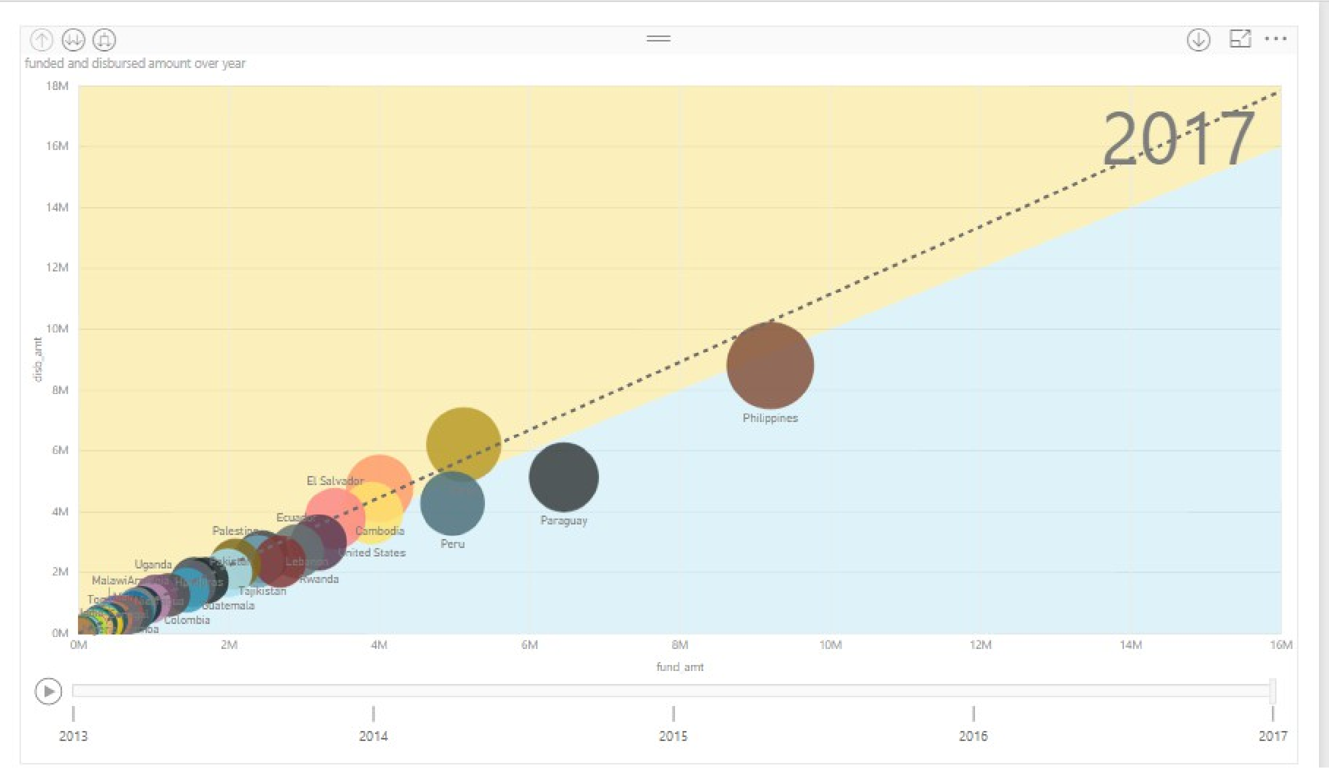
The third grouping is focused on the amount distribution. We wanted to depict the peak of the kiva process over time, as well as the main sectors and reasons the amounts were used for, as well as the most prominent partners.





Amount relations over time:

We decided to focus on the relationship between our two main measures of funded and disbursed amount. We depicted the relationships with choosing as play axes the dimension of year and the conclusion derived was that most loans were successfully disbursed.



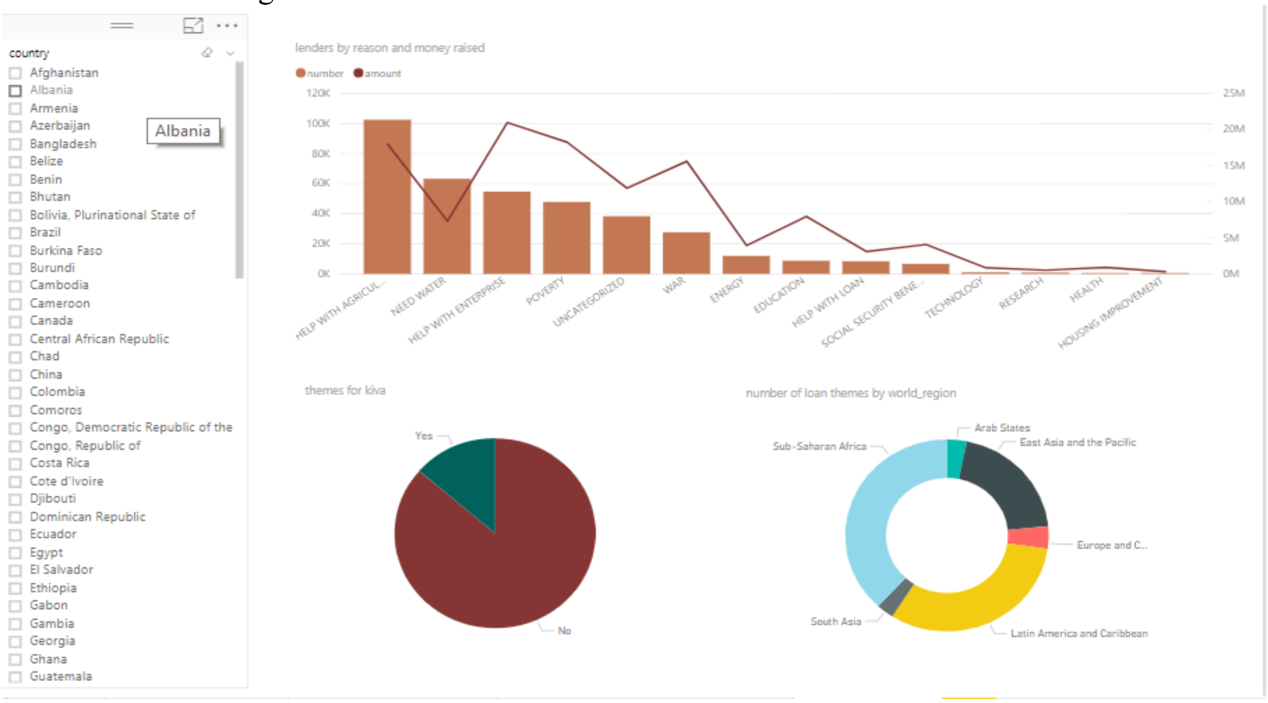
Loan themes:

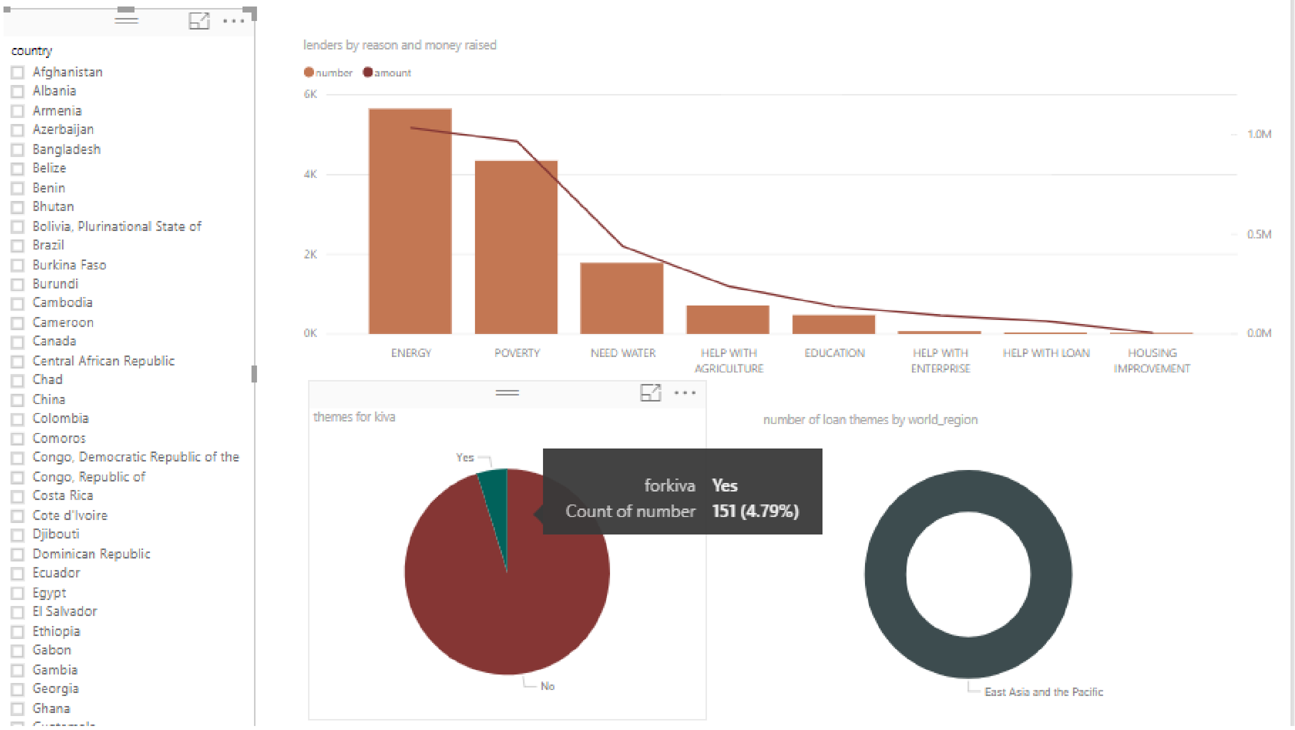
In the last section we focused on the relationship between the number of lenders, the amount of money raised and the reason of the loans.

Also, we show what percentage of themes were especially created for kiva.

Finally we show how many loan themes were created per world region.

It is interesting that, even though more money was given to Asian regions, more themes were created for the Sub-Saharan region.





Our basic conclusion are as follows:

- Most loans were provided in the agriculture sector, followed by food.
- Top loan activities: Farming & General Stores
- Most loans were given in the Philippines. Next country is Kenya.
- Approximately 80% of borrowers are female, and 20% are male.
- Most loans have been funded.
- Most loan applications were due to poverty.
- Most loans are repaid monthly, and mainly within 11-20 months.
- Loan funding was more popular between 2014-2016 and peaked on (November 2016).
- Gathered amount is not proportional to number of lenders.
- Only 13% of loan themes were made especially for kiva.
- Sub-Saharan Africa gathered more types of loan themes than other world regions.