IN4086-14 Data Visualization - InfoVis Project Group 3

Jos Smalbil	1362933	p.j.smalbil@student.tudelft.nl
Olivier	4223209	o.d.f.dikken@student.tudelft.nl
Lex		

December 19, 2018

1 Introduction

This information visualization project is on the topic of refugee migration. During our project we used the methodologies as described in Munzner [4], specifically we followed the four steps of validation as shown in Figure 1.

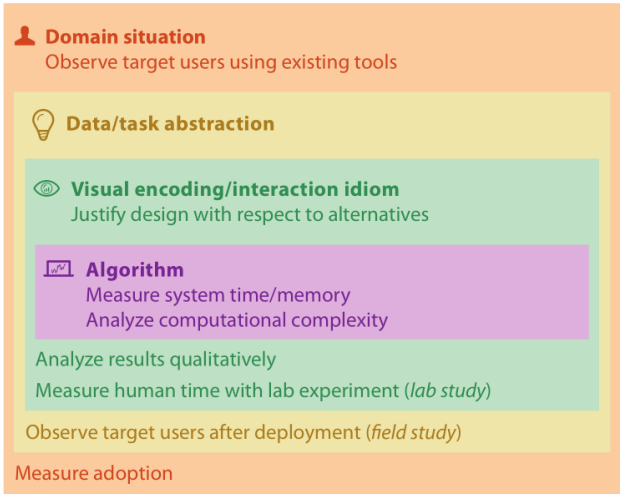


Figure 1: The four nested levels of validation for visualization design

The structure of our report is similar to these validation steps. In Section 2 we introduce the domain situation, in Section 3 we show our data selection and preprocessing steps, in Section 4 we list our task requirements, in Section 5 we go through the visualization and design choices and in Section 6 we mention our results.

2 Domain situation

We have chosen to do a design study about the domain of refugee migration. This is a very heavily debated topic these days, and also a topic often discussed within politics. We have the feeling that there is a lot of misinformation/fake news about refugee migration. Therefore, we created an interactive application using D3 [3] that enables users to validate their ideas about refugee migration and several related indicators. The data we used are data sets *Persons of concern* from UNHCR [1] and 'World Development Indicators' from the World Bank [2].

To get a taste of what kind of questions we expect our application to answer are:

1. How many refugees reside withing a country per year (1951 till 2017)?
2. How do two countries compare based on total refugees?
3. How are refugees originating from country X distributed over the world in year Y?
4. How do indicators like *GDP per capita* or *land area* influence the amount of refugees?
5. Between the Netherlands and Germany, which country took in most refugees from Syria in the last two years? And how do these statistics look when also taking other variables into account like *total population* and *urban land area*?

3 Data selection and processing

We used two data sets *persons of concern* and *world development indicators* stored as csv files. We chose to merge these two data sets into one data set.
The column structure of *persons of concern* is as follows: Year, Country / Residence, Origin, Refugees, Asylum-seeker, Returned refugees, Internally displaced persons, Returned internally displaced persons, stateless person, others of concern, Total population. A lot of these columns had a tremendous amount of null values. Therefore, the main columns we used are Year, Country / Residence, Origin, Refugees.
The *world development indicators* data set has the columns Country Name, Country Code, Series Name, Series Code, 1960 [YR1961],....,2017 [YR2017]. Here we dropped the Country Name and Series Code since we use ISO3 country codes throughout the application and the *Series Code* values were not readable. This format was used as main inspiration for the merged data set.
Due to the large amount of columns (one per year) we generated SQL scripts in JavaScript (create table, bulkinsert data, convert table to new format for the *persons of concern* table so the resulting table has a column per year, compute total refugees per country sum, remove rows where every year value is null). This allows us to easily add new data if the online datasets are updated e.g. with values for 2018, we regenerate the SQL scripts with the new *maximum_year* value.

After the merge the column structure of our new csv files looks as follows:

Country, cell value is the corresponding country three letter ISO code which is an international standard. Country code is more reliable for joins of different data sets, then just using a country name. For instance: one data set could have "Democratic Republic of the Congo" and a different one "Congo (the Democratic Republic of the)" while the ISO letter code has only one unambiguous form as "COD". This attribute is categorical.

Indicator, here the cell values are Series name - renamed as indicator - from the "world development indicator" data set like GDP per capita, but we also made new indicators based on "persons of concern" data set. For instance: the format "Refugees_CountryOfOrigin" would be a new indicator, where "Refugees_NLD" is a indicator which shows how many refugees originating from the Netherlands are at the country of residence. This choice gives us the option to visualize where the refugees per country of origin. This attribute is quantitatively ordered and mostly sequential, but some are diverging such as the Population growth in percentages.

Years, each year is a column just as the world development indicator data set. Start year is still 1951 - the start year of *persons of concern* - and the last year is 2017. The cell values are values from our indicators for the year of the column name. Querying years is easier now as where clauses are not needed anymore compared to the alternative of storing all years in one column. This attribute is ordinal and sequential.

The last preprocessing step was to convert the merge csv file to a JSON format. The most important step was to group by Country (object) and make each Indicator a property. The value of a property is an

array of length (last year - start year = 2017 - 1951 = 66), where at index zero we get the value of the property for year 1951, and so forth. The JSON table format makes the data very easy to access with the d3 visualization tools, also the direct accessing of JSON[Country][Indicator][selected_year - 1951] makes it fast and straight forward without the need of queries. We added the option to select the total refugees per country per year (without selecting a country of origin) by summing over all refugees_[country of origin] indicator values per country/(country of origin) combination per year. Selecting *country of origin: All* is also set to the default option in our application.

For drawing the world map we used a TopoJSON file which contains spatial data. TopoJSON is an extension of GeoJSON that encodes topology and it is more compact than GeoJSON files because of less redundancy. The objects in our TopoJSON file are of type Polygon or MultiPolygon and they contain the attributes arcs, the full name and the three letter ISO code for each country.

In our case, both the indicator dataset and the TopoJSON can be considered as static data. When new yearly data becomes available it is easy to rerun our preprocessing code to update the application.

3.1 Chosen indicators

The main indicator is the amount of refugees per country. When this indicator is selected a second drop down appears in which the user can select the country of origin (e.g. to track Syrian refugees).

This is the list of indicators available in our application:

1. Refugees_Total
2. Land area (sq. km)
3. Individuals using the Internet (% of population)
4. Population growth (annual %)
5. International migrant stock—total
6. Labor force—total
7. Wage and salaried workers—total (% of total employment) (modeled ILO estimate)
8. Urban land area (sq. km)
9. GDP growth (annual %)
10. GDP (current US$)
11. Population—total
12. GDP per capita (current US$)
13. GNI (current US$)
14. Electric power consumption (kWh per capita)

The user can normalize the primary indicator data with the secondary indicator in the bar chart. *Land area*, *Migrant stock*, *labor force*, *total workers*, *gdp* and *total population* are indicators that can help compare the data of two countries with different characteristics. The other indicators were added for the user to explore the data from different perspective. In the results section (see 8, 8a and 8b) The Netherlands and Germany total refugees are first compared without normalizing using the secondary indicator, and then normalizing using total population and then normalizing using land area. All three charts give a different perspective about the data.

4 Task descriptions

A first action of the user is to consume. Without too much effort the users should be able to discover global aspects of the refugee migration data. The user should be able to track the development of refugee migration throughout the years for it's selected countries and see an overview for all countries simultaneously. An important aspect of the discovery action is to let the user test or explore its own beliefs (hypothesis) about refugee migration. If the user has no specific hypothesis the performed actions can be exploratory. Example questions of this category are:

1. Which country holds the largest amount of refugees in 2010?
2. Which country has the largest population in 2017?

The user supposedly also has some hypothesis about migration in his own country or certain other countries. If the geographical location of a country is known and if the hypothesis is known the search action will be a simple lookup. If the user has no hypothesis the search action will be a browse action. For example, after selecting a certain country the user can browse through the list of secondary indicators. Example questions of this category are:

1. What is the number of refugees over time residing in The Netherlands?
2. How does the number of refugees residing in the Netherlands in 2000 compare to other countries in the world? Is it high or low?
3. What are popular countries of destination for refugees from Afghanistan in 2017?

Because data of a single country is hard to interpret we also want an action task that enables country comparison. It could be interesting to see how your own country stacks up against a neighbouring country. Additionally, there might be explanatory variables which influence the total migration of a country. This action task enables a user to browse and lookup and answer these kind of questions:

1. How does the number of refugees residing in The Netherlands and Belgium compare over time?
2. Does The Netherlands or Belgium host the most refugees from Afghanistan in 2017?
3. Does the secondary variable total land area influence the amount of refugees?
4. Which countries have similar amounts of refugees as my country of interest for a specific year? How do these values compare when looking at other years? How do these values compare when dividing by the total land area or total population?

5 Visualization and design choices

After defining our task descriptions we did a brainstorm on visualization techniques that are best suitable for our tasks and domain. Besides a world map we chose to use a dual bar chart and a scatter plot. The bar chart combines the important task functionality to compare two countries and the functionality to observe the data over time. The scatter plot is suitable for the tasks to explore and discover outliers and correlation in the data.

5.1 World map

One of the first design choices we made is to construct a world map. A world map synergies well with the strong geometric and spatial perception of humans. It links the name of a country to the geometrical shape of the country and its position on the map relative to other countries. It represents all the different countries better in it's view than alternatives like drop down menus where each country is an option. The world map is used as a chloropleth - which can be seen at Figure 3 - for the amount of refugees for each country, this technique excels at presenting data differences for one variable between

countries world wide which is exactly what we want. We have chosen for a sequential single hue (green) for the legend of the chloropleth; we have already used red and blue in our dual chart, therefore green seemed as a logical choice. We explicitly wanted to avoid a coloring scheme with too many colors, because these are harder to interpret and because we already use other colors for the selection of countries. Because almost all countries have a fairly small amount of refugees and only a few have large amounts (see Figure 2) a linear color scale gives strange results. We decided to base the color scale on the quantiles of the underlying data distribution for an optimal visual representation. The chloropleth makes for a nice global overview of the refugees distribution world wide, besides that it opens up for different interaction possibilities. The user can select two countries at the world map to further investigate specific information about these countries. Selected countries are recognized by their border color hue. We decided to increase their border size to make the channel more visible. By using this technique instead of fill coloring we made sure that very small countries are still easily recognized as selected. To select small countries such as The Netherlands, Belgium and Luxembourg easily we decided to add a zoom function to the world map. The selection of the two countries is linked to the dual bar chart and the scatter plot which we will introduce now.

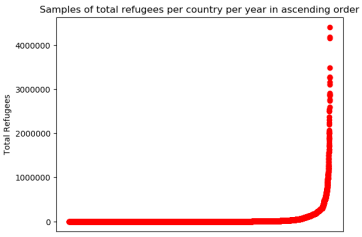


Figure 2: Distribution of the total refugees for each country and each year data in ascending order

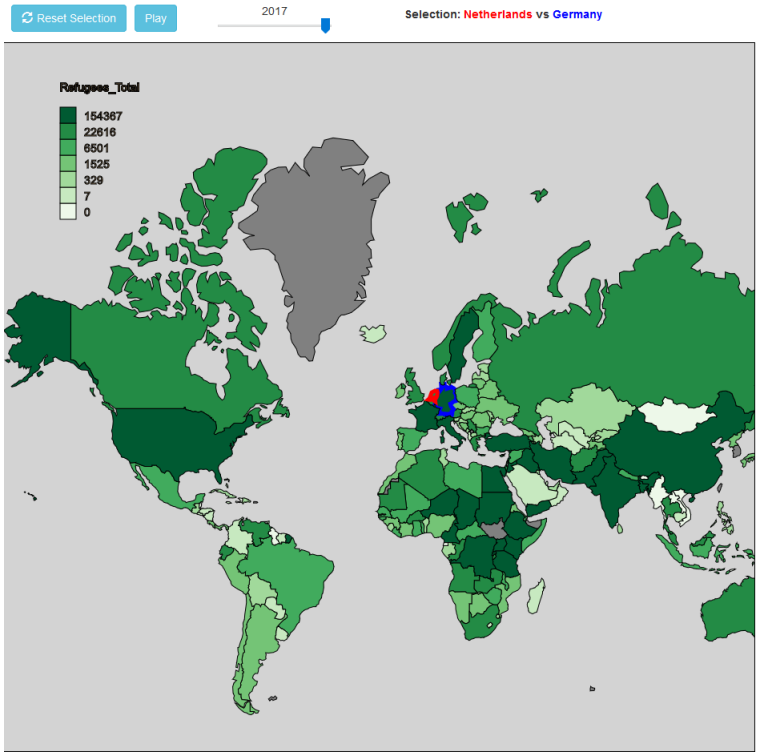


Figure 3: World map Netherlands vs Germany 2017

5.2 Dual bar chart

Figure 4 shows the dual bar chart where the comparison is between Netherlands (red) and Germany (blue). This bar chart shows the progression of time at the x axis and at the y axis is the selected primary indicator variable. The primary indicator variable can also have negative values for diverging attributes, thus the bars start at y axis value 0 and can go up or down. This view is important for the comparison between the two selected countries. The dual bar chart shows the progression of the selected primary indicator variable through time to the user. We have chosen for a dual bar chart, because we want to do a comparison between two countries for all years. The length of a bar gives the user a good interpretation for the different values that occur within the variables. Line charts would also have been a viable option. However, we have discrete data and the changes for the variables are sometimes very large and the bar charts visualize these changes better. The dual chart is also very handy to compare two countries compared to vertical difference between the y axis of two lines.

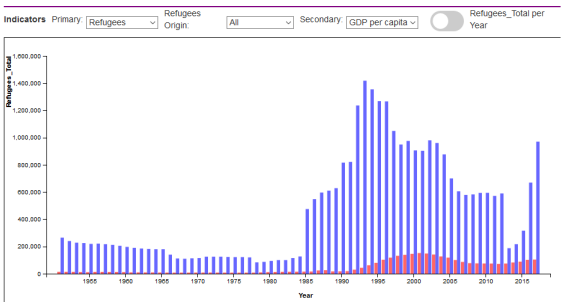


Figure 4: Bar chart Netherlands vs Germany

5.3 Scatter plot

Figure 5 shows the scatter plot for the year 2017 with Germany and Netherlands selected, and variables Refugees_Total and GDP per capita were chosen as example; at the y axis the primary indicator (Refugees_Total) is shown and at the x axis the secondary indicator (GDP per capita) is shown, and each dot represent a country. This is a great plot to observe outliers in the data and correlations between variables among countries. To discover correlations we could also have used a parallel coordinates plot, but this plot is unable to detect outliers which is an important task. Countries can be selected - at figure 4 red is Netherlands and blue is Germany - by clicking on their scatter plot dot. We don't want the user to be only reliant on the world map to make selections and above that numerical considerations are also possible now.

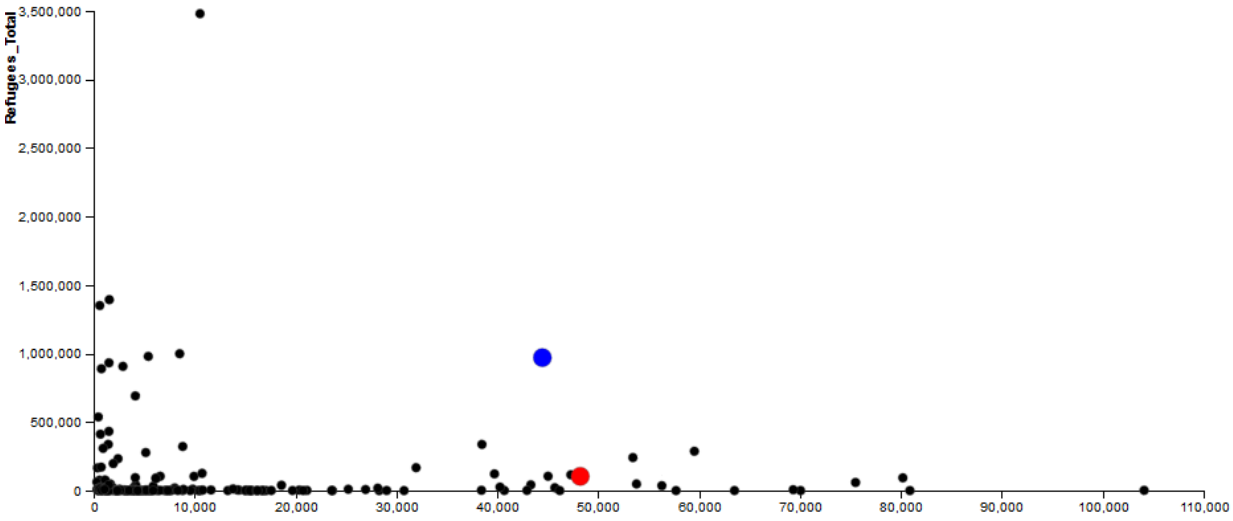


Figure 5: Scatter plot Netherlands and Germany selected 2017

5.4 Linking the visualizations

Our multiform design choice using juxtaposed views enables the support for more tasks and faster switching between them. We note that all our three views have different visual encodings. However, the views do share (some of) the data. For example, all countries are visible in the world map geographically but also in the scatter plot as dots. Also, for the two selected countries, the bar chart and the scatter plot show the same data for the primary indicator for the selected year. Because of this overlap it is possible to link the views and thereby also improve the performance of some of our tasks as described in Section 4.

There are three main variables linked between all three views; the current selected year, the two selected countries and the primary indicator which is the total number of refugees by default. The selected year of all three views can be updated in two ways; by clicking or moving the time slider at the top or by selecting a bar of another year in the bar chart. The primary indicator can be updated by using the drop down list. The selection of two comparison countries can also be done in two ways; one either selects a country by clicking on the geographic location on the world map or by clicking a dot on the scatter plot.

The geographic countries and the scatter plot dots are linked through linked highlighting; for an on hover action of the country the corresponding scatter plot dot becomes yellow and for an on hover action of a country dot the corresponding border of the geographic country is highlighted yellow. Additionally, the first selected country will color red in all three views and the second country blue. Only when two countries are selected their primary indicator values will be shown in the dual bar chart.

5.5 Play button animation

The play button increments the selected year variable at fixed time intervals. The scatter plot and the world map visualizations depend on the selected year and they are updated accordingly. This button enables the user to recognize refugees patterns over time for more than two countries. The button should be used an exploratory tool which can lead to new questions. Note that there are a lot of updates going on at the same time, thus the user should experiment with the transition speed and easing type in order to be able to track the changes over time. Note that transitions work slightly differently during the play animation: transition time is decreased but more importantly we removed delaying events (see 5.6 for more details) so that the scatter plot transitions look smooth.

5.6 Transitions

When changing the selected year value or when selecting a different second country for the comparison the data interpolates between the previous and new values over time using of the D3 transition functions. Our application only shows transitions when it is relevant: when changing indicator(s) there are no transitions because the new indicator data values have no relation to the previously selected indicator data values (i.e. when switching primary indicator from *total refugees* to *land area* we do not want the bar chart bars' heights representing the total refugees to transition to the bar chart bars' heights representing the land area because this data is not linked and could disorient the user).
The dual bar chart bars' heights and y axes transition when the second selected country changes. Selecting countries in our application works by selecting the first country if no countries are selected, and selects/replaces the second country if one or two countries have already been selected. The behaviour of replacing the second country (the 1st country stays selected) is done so that the user can compare multiple countries one by one to the first selected country. When changing the second selected country the dual bar chart bars' heights and y axes transition to the new values. This allows the user to get a feeling of how several countries compare to the first selected country.
The scatter plot points transition when the selected year changes. The selected year can change through direct user input (moving the year slider or clicking on a dual bar chart bar) or during the play animation. When the year changes due to direct user input then the scatter plot points that are matched with the new data transition normally, the removed points fade out, but the new point fade in with a delay. This needed to be done so that the user can track the points that are transitioning without new points fading in which is visually confusing since we can have more than 200 points clustered closely together in the plot in some scenarios. Fading in the new points with a delay allows the user to first observe the movement of the

transitioning points and then see the new points (for which there was no data for the previously selected year). This delay is disabled when the year changes due to the play animation.

The transition settings (can be accessed by clicking on the gear icon in the top right corner) to change transition speed and the easing function are meant for the user to fine tune the application to its needs. When the user wants to closely follow the play animation then it is suggested to lower the animation speed and change the easing function to *linear*. When the user does not want to be distracted by animations or only want the visual feedback without paying attention to the data it is suggested to use a high transition speed setting and select the *cubic* easing function.

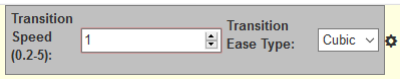


Figure 6: Transition settings

6 Results

After performing the tasks as defined in Section 4 one of our findings is that the number of refugees residing in The Netherlands is lower than in Belgium until 1991. Since that year the number of refugees in the Netherlands have increased significantly. We easily found this observation by selection the two countries on the world map using the zoom functionality and then by looking at the dual bar chart. The bars of the first country are colored red and the bars of the second country blue. Thus, our choice for linked highlighting by coloring the borders of the selected country was beneficial for this observation. Also the zoom functionality for the map is beneficial to select a small country such as Belgium.

Another finding was when comparing The Netherlands with Germany (Figure 7). Germany has a significant larger refugee population than The Netherlands but after normalizing this value with the secondary indicator total land area variable on the bar chart we see that The Netherlands and Germany perform actually quite similar. This observation was found by first selecting The Netherlands and Germany using the world map and consecutively executing the normalization function for the dual bar chart (Figure 8b).

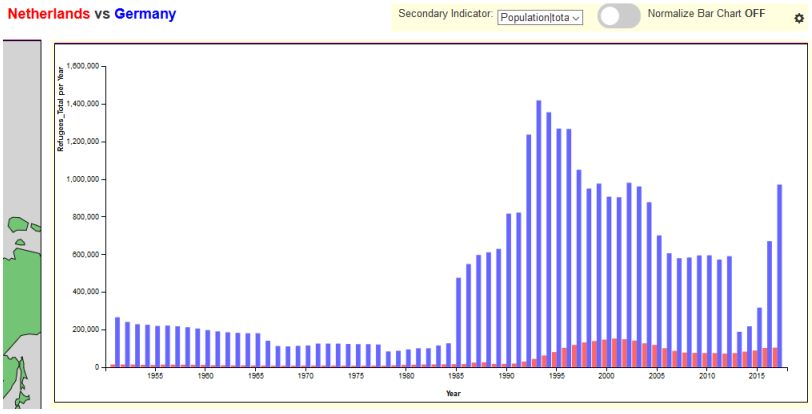
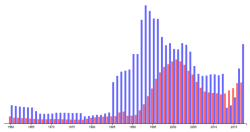


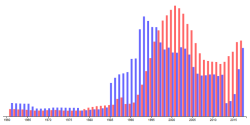
Figure 7: Total Refugees Netherlands vs Germany

Our application can answer all of the tasks as previously defined. A somewhat disappointing result is that we did not find any secondary indicators with a high correlation with the total number of refugees of each country.

We note that when performing the tasks all visualization changes due to on hover and on click events are performed instantly. This is due to our design choice to load both JSON files on page load in memory and also to reduce the necessary calculations as much as possible.



(a) Total Refugees / Total population



(b) Total Refugees / Land Area

Figure 8: Netherlands vs Germany

References

[1] Unhcr population statistics - data - overview, 2018. Retrieved 19 December 2018 from `https://popstats.unhcr.org/en/overview`.

[2] World development indicators - databank, 2018. Retrieved 19 December 2018 from `https://databank.worldbank.org/data/source/world-development-indicators`.

[3] Bostock, M. D3.js - data-driven documents, 2018. Retrieved 19 December 2018 from `https://d3js.org/`.

[4] Munzner, T. *Visualization Analysis and Design*. AK Peters Visualization Series. CRC Press, 2015.