COM-480:Data visualization
Data visualization of marine plastic pollution

Ana Stanojevic *(268578)*
Sergii Shynkaruk *(269469)*
Mohammad Aquil *(273107)*
École polytechnique fédérale de Lausanne, Switzerland

December 22, 2017

Contents

1 Introduction

As the world's population continues to grow, so does the amount of garbage that people produce. On-the-go lifestyles require easily disposable products, such as soda cans or bottles of water, but the accumulation of these products has led to increasing amounts of plastic pollution around the world. Plastic is a polymeric material. Natural polymers such as rubber and silk exist in abundance, but nature's "plastics" have not been implicated in environmental pollution, because they do not persist in the environment. [1] However, since synthetic plastics are largely nonbiodegradable, they tend to persist in natural environments for a long time. Plastic pollution involves the accumulation of plastic products in the environment that adversely affects plants, wildlife and even human population.

Since the ocean is downstream from nearly every terrestrial location, it is the receiving body for much of the plastic waste generated on land. Between 4.8 million and 12.7 million tonnes (between 5.3 million and 14 million tons) of debris end up in the world's oceans every year, and much of it is improperly discarded plastic litter. While solving the problem of plastic pollution may seem as easy as just implementing recycling or cleaning up empty bottles, the truth is that the plastic, causing the pollution, can range in size from big to microscopic. The major contributors to this problem today include everything from "plain old trash" such as milk cartons and water bottles, to cosmetics which produces mainly microplastic, to fishing nets which spend long times submerged in water (leaking toxins).

It seems rather obvious that this amount of material that is not meant to break down can wreak havoc on natural environments, leading to long-term issues for plants, animals, and people. Because it comes in sizes large and small, polluting plastics even affect the world's tiniest organisms such as plankton. When these organisms become poisoned, due to plastic ingestion, this causes problems for the larger animals that depend on them for food. This can cause a whole slew of problems, each step further along the food chain. Hence, plastic are present in the fish that many people eat everyday.

Plastic also pollutes without being littered—specifically, through the release of compounds used in its manufacture. Indeed, pollution of the environment by chemicals leached from plastics into air and water is an emerging area of concern. [2] It has also been shown that even use of and exposure to plastics increases number of health concerns affecting people. Different dangerous compounds have been detected in humans, disrupting the endocrine system. [1] Phthalates act against male hormones and are therefore known as anti-androgens; BPA mimics the natural female hormone estrogen; and PBDE has been shown to disrupt thyroid hormones in addition to being an anti-androgen. The people most vulnerable to such hormone-disrupting chemicals are children and women of reproductive age. Moreover, the world's water is in great danger because of leaking plastics and waste. Groundwater and reservoirs are susceptible to leaking environmental toxins.

Given seriousness of negative effects and also tendency to further growth of plastic pollution, there is a huge need to talk about this topic, from different perspectives. Plastic pollution topic attracts more and more enlightened people everyday, willing to give their contribution. Related work to this topic is mainly connected with different type of organizations and foundations, trying to raise awareness as well as undertake concrete actions.

Among other research, scientist are also being payed to collect, as much as possible data, on plastic pollution in oceans, and make accurate approximation on actual amounts. The first oceanographic study to examine the amount of near-surface plastic debris in the world's oceans was published in 2014. It estimated that at least 5.25 trillion individual plastic particles, weighing roughly 244000 tonnes, were floating on or near the surface. Floating plastic waste has been shown to accumulate in five subtropical gyres that cover 40 percent of the world's oceans.

Main goal of this investments and researches is the hope their findings will help to to raise awareness and encourage people to work on solutions to solve the global plastic problem. Goal of

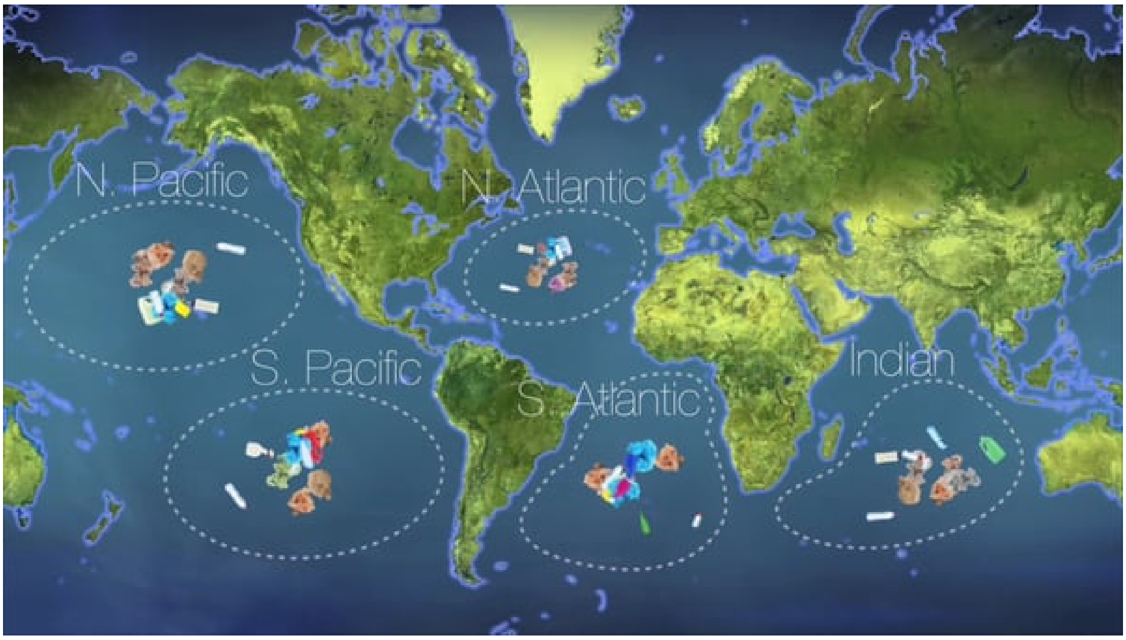

Figure 1: Five subtropical gyres that cover 40 percent of the world's oceans.

this project is according with that aim. By visualizing two different data sets, we want to make previously stated facts closer to large audience. Our target group are people throughout the world who are not yet "convinced" about the effects that plastic pollution brings. There is small emphasize to younger generations, because they will need to deal with it in the futures, as the problem won't go away so quickly. By simplifying matter and data through visualization, we want to strip naked important and significant facts, with interactive and attractive interface. Moreover, we wish to highlight non-obvious facts.

Finally, ultimate goal of the visualization, is to make people feel responsible for their actions, and become an active part of solution.

2 Dataset

For our visualization we used two separate data sets, both containing data on marine plastic.

First data set represents measurements of expeditions, which were happening across different oceans, in period from year 2008. to year 2013.

Other data set we used, is referring to directions of garbage spreading (throughout oceans), after it has been thrown into the water at the specific point on earth.

2.1 Expeditions data

The data set was collected by an oceanographer named Marcus Eriksen. He spent six years going on expeditions, around the world's, to study the amount of plastic pollution. In his paper 'Plastic pollution in the world's ocean'([3]), he used obtained measurements, to created a model which approximates total number of plastic waste in each ocean. According to his results, at least 5.25 trillion plastic particles (weighing close to 268,940 tons), are currently floating into the sea.

[4] The data set represents 1571 locations, each of them belonging to an expedition (over 30 expeditions in total). At each of this points, several measures were taken. Eriksen categorized all plastic garbage into 4 groups, according to the size of it (0.33-1mm, 1-4.75mm, 4.76-200mm and >200mm). Then, for every group, he measured total weight of garbage and total number of garbage pieces, found at the location.

For expeditions visualization, it was important to distinguish two core notions:

- Expedition Point: complete set of measured data; see Table 1
- Expedition: finite set of Expedition Points

Measured Parameter	Type	Info	Units of measure	Details
Date	dd/mm/yyyy	Expedition Date		
Latitude	Float	Expedition's Point Geo-Location		
Longitude	Float	Expedition's Point Geo-Location		
CD1	Float	Particles Number	$\#/km^2$	.335-.999 mm
CD2	Float	Particles Number	$\#/km^2$	1.00-4.75 mm
CD3	Float	Particles Number	$\#/km^2$	4.75-200 mm
CD4	Float	Particles Number	$\#/km^2$	>200 mm
WD1	Float	Weight	g/km^2	.335-.999 mm
WD2	Float	Weight	g/km^2	1.00-4.75 mm
WD3	Float	Weight	g/km^2	4.75-200 mm
WD4	Float	Weight	g/km^2	>200 mm
Sea State	Float	Beaufort Scale		
Source	String	Data Provider		
Info	String	Expedition Date Identifier		

Table 1: Measured parameters at a single Expedition Point; First column contains names of all parameters. CD[1-4] and WD[1-4] are abbreviations for *count debris* and *weight debris.*

2.1.1 Pre-Processing

In our visualization we used only some of measured parameters. Some other e.g. 'Source' and 'Sea State' are weakly relevant, hence we have not included them to the data visualization. Moreover, data preprocessing phase included separation of the list of 1571 locations, into independent expeditions. To do this, we noticed that every data-point contains *Info* property, which uniquely identifies expedition it belongs to, as well as provides an order of this precise point within it. By grouping the locations using this property, we partitioned the overall data into separate blocks of points. Furthermore, we combined expeditions based on their geolocation, so that each group represents the data for some bounded region, in specific - an ocean.

We have previously mentioned that Eriksen used collected data to create a relatively complex model, which approximates the total amount of plastic waste in each ocean. In our visualization, we also wanted to represent total estimated amount of garbage in each ocean. Even though there is a paper describing this precise model, because of the purpose of this project (which is mainly on data visualization), we decided to use much simpler approximation.

To accomplish this, we observed groups of Expeditions belonging to the same ocean. For each such group (ocean), we observed all Expedition Points contained inside it, with measured values of parameters, which we described before (e.g. value of CD1 could be for instance - 58102 pieces). For each parameter, our model calculates its approximate value for the ocean, by averaging values over all Expedition Points belonging to that ocean. This is a fairly modest algorithm and one should be aware that it can be pretty inaccurate, given that points in which the measures are taken are not taken randomly (and also there is not enough of them for the results to be completely reliable). However, this gave us a decent approximate values, and for reader who wants to find out more, there is always additional material which focuses on more advanced estimations.

Furthermore, it is worth mentioning that there was some amount of missing data among Expedition Points. The non-informative data was removed, however by following the same principle from above, we did not spend a lot of time figuring out what is the best way to deal with such data.

For the reasons stated in previous two paragraphs, we invite a user to enjoy our visualization, but to take exact numbers with a grain of salt.

2.2 Spreading data

Scientific methods used to calculate the paths of floating debris through the oceans, for 10 years after release, are quite complicated. Driven by the sun and the wind, the oceans develop mighty currents and eddies, some of which can take centuries to loop through all of our planet's ocean basins. These currents also move through three dimensions. Many rise from the deep ocean near coastlines while other currents descend to the deepest parts of the ocean. However, it is important to notice here that almost all plastic materials are lighter than water and therefore, they stay on the surface. Starting from 1982, World Climate Research Program put forward the idea of a standardized global array of drifting buoys. These buoys float with the currents just like plastics except - they send a short message to scientists every six hours about where they are and the conditions in that location. This information enabled team lead by Dr Erik van Sebille, to create a statistical model of the surface pathways of our oceans [5]. This model is then used to generate an animation of the likely path and destination of floating debris over a ten year period into the future. The data we use is downloaded from their official website ([6]). There, it is possible to obtain spreading predictions for each point in the ocean. For our project, we chose 11 strategically important locations and gathered the relevant data, in a .csv format.

The obtained data-set contains five columns (features): year, month, latitude, longitude and the probability. Each row in the file, corresponds to one point on the map. The sum of all probabilities, for a given year and month (there could be multiple lines with the same year and month) equals to 1. With exception of locations, where the weight is smaller than 0.00025 (they are not listed). One row in the file describes probability that thrash, which is thrown at certain point in the ocean, ends up after x years and y months, at point with latitude lat and longitude lng. In the Table 2 there is a detailed description of each of these fields.

Parameter name	Type	Info
Year	Integer $\in [0, 10]$	Year offset
Month	Integer $\in \{0, 2, 4, 6, 8, 10\}$	Month offset
Longitude	Float $\in [-90, 90]$	Spreading Point Geo-Location
Latitude	Float $\in [0, 359]$	Spreading Point Geo-Location
Probability	Float $\in [0, 1]$	Probability of reaching Geo-Location

Table 2: Parameters measured at a certain Geo-Location; First column contains names of all features.

2.2.1 Pre-Processing

In the design process, we decided to present spreading of thrash as a tree-structure, with a root being a point where the garbage was initially thrown. Furthermore, locations which are given in the original data, would to be nodes of the tree. Turns out, that there is a whole range of problems which need to be overcome in order to create such a structure. Some of them we describe below. Once the tree structure is created, we save it in a .csv file, and use it directly for the visualization.

As a beginning of the pre-processing step, we decided to remove all the points with probability less than a certain threshold. As a result, we significantly reduced the number of points, hence the number of nodes in the final tree. We wanted to avoid overloaded visualization, and focus only on points with significant probabilities. Moreover, in order to work with longitudes as they are defined in the literature, we mapped longitudes from the data which belong to range [181, 359], to the interval [-179, -1]. This made it possible to use libraries `geopy`, which are specified for geographical maps.

Since our goal was to define a tree structure, it was required to know which nodes belong to the same level and which node is a parent of the chosen node. We decided that all nodes that belong to a certain time interval should be at the same level. According to this, we chose that time difference between each two levels, should be 2 months. This means, that all nodes on the first level, would be the points on map where there is a probability that garbage appears after only 2 months. Further, all nodes on second level are made of points on a map, for which there is a probability that garbage arrives there after 4 months, and so forth (This makes a logical choice for levels, as they follow our goal, which is to show possible spreading of garbage over time, starting from a root node.)

Hence, in our implementation we iterate over the list of locations (nodes), sorted by the time. For each of them, we had to find a parent node, among nodes from a previous level. To do this, we calculated geographical distances, and chose the closest one as ancestor.

In particular, it is interesting to describe some special cases, which we encountered. Namely, some of the eleven root points, which chose for our visualization, are very close to the "boarder" of the map. E.g. point close to Chine coast, has longitude value, around 160 degrees. This creates a problem, which might not be obvious immediately. While spreading, there is a big probability that some of the nodes will have longitude bigger than 180, i.e. they should appear on the other side of the map. For such nodes our algorithm creates edges which go over whole screen, and that was certainly not what we wanted to achieve. Therefore, for the previously described case, we created a new, auxiliary node, with longitude -179 and with the same latitude as the original node. Consequently, our original node was able to recognize it as a parent node. In this case, auxiliary node keeps no parent, so it could serve as a root of a new tree. Therefore, in this special case we deviated from single tree structure, and created multiple trees, which can be connected between them. This solution allowed to create an expected and intuitive visualization.

Once we created a graph, we also wanted to color the edges in the way that they reflect probability of a garbage to arrive at a certain point. We sorted all nodes according to probability value given in the original data. Edges which go towards nodes with smallest probability (lowest 90% of sorted list), we color in gray. Edges that go to the nodes with higher probability (i.e. belonging to top 10% highest), we color in red.

Final output which we use for visualization is given in Table 3

Parameter name	Type	Info
Depth	Integer $\in [0, x] (x \sim 59)$	Level in the tree structure
Child Longitude	Float $\in [-90, 90]$	Spreading Point Geo-Location
Child Latitude	Float $\in [0, 359]$	Spreading Point Geo-Location
Parent Longitude	Float $\in [-90, 90]$	Spreading Point Geo-Location
Parent Latitude	Float $\in [0, 359]$	Spreading Point Geo-Location
Color	Float $\in \{red, grey\}$	Color of edge

Table 3: Parameters measured at a certain Geo-Location; First column contains names of all parameters.

3 Design

In the beginning phase of our project we were thinking of the best ways to visualize the given data sets. During this period, in order to better understand the data, we used `python` language together with `pandas` package. With `describe()` function of a `DataFrame` we observed basic descriptive statistics. We also plotted some informative figures, which helped us to understand the data distribution. E.g. for spreading, we observed probabilities distribution, by using simple histograms.

Taking into account that both data sets that we work with, contain latitude and longitude values, we agreed to pursue the option of using the Geo-map for our demonstration. Indeed, map charts are good for giving numbers a geographical context. They make it possible to quickly visualize and spot best/worst performing areas.

From exploration of Expedition data set, we agreed that there are two types of information which essentially need to be shown. Firstly, we should familiarize a viewer with the details of expeditions alone. Where are they led, how long they were, what are the Expedition points where the biggest amounts of garbage were found, etc. It is important to give a user closer information on how exactly the measurements were done. Secondly, we should represent results of their research, in the best possible way. It is needed to aggregate information and offer wider perspective, analysis and comparison.

On the other hand, spreading data contains much smaller number of parameters. Therefore, options were much narrower, considering what we can present. We decided to show the main information which it brings, i.e. to emulate probable trajectories of plastic waste motion with regard to the Geo-location of pollution source. Even though there could be other interesting aspects of this data set, this is in according with our goal to make the viewer feel the possible negative impact of her/his actions. User can identify and imagine, how small actions lead to long-term effect in the environment.

With this three aspects in mind, we created visualization which consists of three separate web pages. In the bottom left corner of each of these web pages there are three buttons, which make it possible from any of them to switch to other two (Figure 2). Buttons are colored such that, they make easier for user to use them intuitively. Button which is currently pressed is colored in "darker" color which makes a perception, that it is pressed. Also, while hovering over pressed button, no action happens. In contrast to this, other two buttons are colored in lighter colors, and while hovering over them there is a change in color, in such way so that user understands intuitively that the button can be pressed. First two buttons, denoted *Overview* and *Expeditions*, are displaying information obtained from expeditions data set. The third button, denoted *Spreading*, displays information obtained from other data set (spreading data set).

Our main goal for this project was to create interactive, relatively simple and intuitive visualization, which displays enough information to tickle the imagination of the viewer and make one start asking questions. Throughout the process, we payed attention to follow good practises, as well as use the perceptual and design principles.

3.1 Overview

As previously mentioned, overview web page was meant to display an aggregated information of results of many expeditions done in a period of 6 years. At the bottom of the page there is a button *Oceans' statistics* (which satisfies all "rules" mentioned for all other buttons). With a mouse click on this button we open the chart, with statistics summarizing expeditions for each ocean.

What may come as a bit unexpected on this page, is that we have an "empty" map, while a button opens "true" visualization. This phenomenon is worth explaining. Namely, while we were

Figure 2: Buttons, which are use to switch among different web pages of visualization.

discussing our design methods, we were considering to represent aggregated oceans' statistics on a map as well. E.g. one of the design ideas was to color each ocean, depending on the total amount of garbage in it. However, this would have some challenges in the implementation. Another idea, was to put names of the oceans on the map, and make them "clickable". Hence, when the user clicks, it would show information about the garbage in that ocean. This idea was also eventually rejected, because we wanted to make some clear comparison among the oceans, and it was not obvious how to show that with such an interface. Finally, we decided to leave the map behind the statistics, so that user can have nice reference while observing the chart. Also, it is a nice groundwork for some possible future improvement.

[7] Design principles which we applied, (in terms of intended functionality and actions), are given in Table 4.

Analysis:	Discover	Present	Enjoy	Annotate	Record	Derive
	✓	✓	✓			
Search:	LookUp	Locate	Browse	Explore		
		✓	✓	✓		
Query:	Identify	Compare	Summarize			
	✓	✓	✓			

Table 4: Design principles for overview web page.

[7] For visual encoding of principles stated in Table 4, we decided to use Sankey diagram ([8]). A Sankey diagram is a visualization used to depict a flow from one set of values to another. The things being connected are called nodes and the connections are called links. Sankeys are best used when the goal is to represent a many-to-many relationship (mapping) between two domains. Since we have mapping from set of oceans to multiple garbage categories, we recognized this type of diagram as a perfect tool to use. Moreover, the width of the arrows is shown proportionally to the flow quantity. This is very useful, since the length is recognized as one of the most reliable channels.

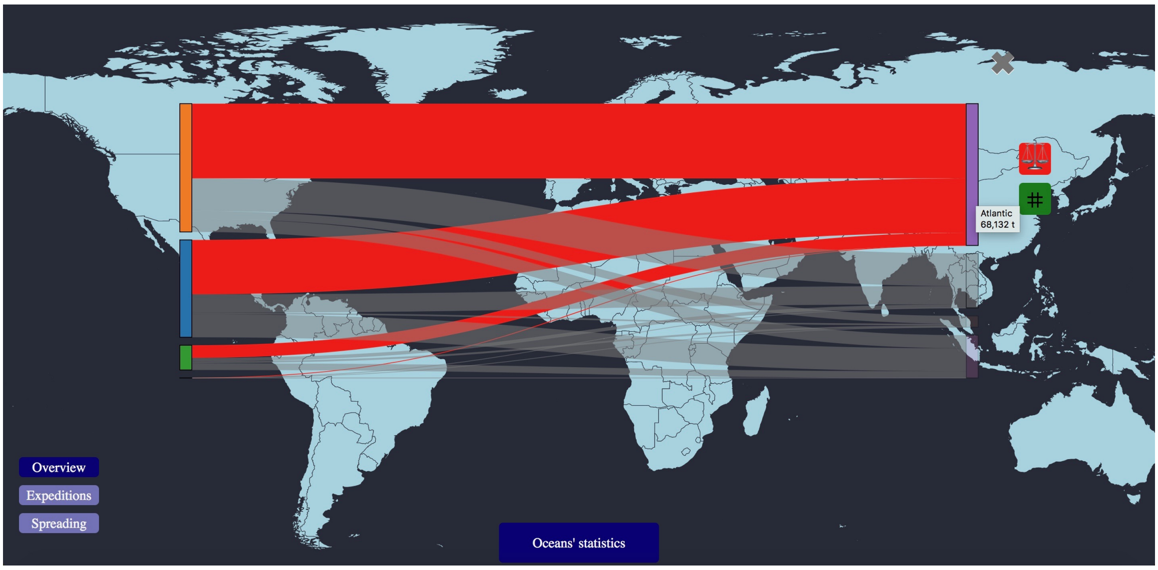

Figure 3: Shows functionality, where by hovering over rectangles we can explore statistics for each ocean and each garbage category.

Once the diagram is opened, we observe the transparent panel, i.e. the map can be seen behind it. We implemented this in order to make a direct connection and reference from statistics chart to real oceans. One can also observe that the chart has pretty low opacity. Moreover, we did not put ocean names directly on the chart. Our intention was to make user take a closer look and explore chart, by trying different actions (such as for example hovering). Main idea for the user is to "play around" with the chart and to discover detailed information and functionality.

First layer of interaction, is possibility to hover over the chart rectangles, and semantically zoom on each ocean and on each type of garbage. By applying wide range of perception methods (color, similarity, connectedness and continuity), we make information about that particular object "pop out" on the chart. What we effectively do, on the already existing Sankey diagram, is increasing opacity and changing color (while the rest is implied from the diagram). Here, we can explore which ocean contains what amount of each category of garbage, how does it compare to other oceans, what is the total amount of garbage in specific ocean, and many more. This can be observed on Figure 3.

Furthermore, it is also possible to hover a mouse over the links and get information. When a mouse is over an edge, that link "pops out", together with the specific ocean and type of garbage to which it is connected. Information which can be explored here is e.g. which amount of '>200mm' garbage type is contained in e.g. Atlantic. This can be observed on Figure 4.

From the previously stated it is clear that when we put mouse on rectangles and links, this objects "pop out". However, it may be not-intuitive the way we obtain exact numbers on oceans and garbage. This is actually another action implemented. Namely, when the mouse is hovered over some element (either node or edges), the small box with this numbers appears, informing us about selected group of element. This can be also observed on Figure 3 and Figure 4.

Lastly, on the right side of the chart we can observe two buttons (very descriptive). By switching to one or the other button, we show two distinct charts (both of them working with the similar principles though). One of them shows the data expressed in weight of garbage (denoted with scale

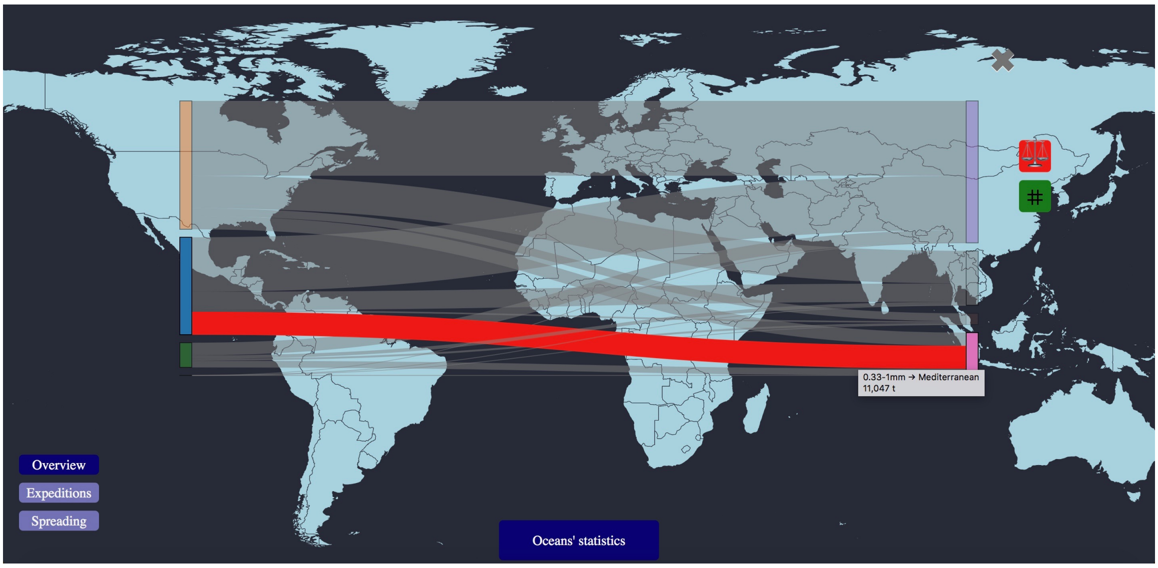

Figure 4: Shows functionality, where by hovering over links we can explore amount of certain garbage type in chosen ocean.

picture), in a certain ocean, with *tones(t)* as a measurement unit. Other one is expressing number of pieces found in a certain ocean (denoted using '' sign), with *billions* as a measuring unit.

Finally, it would be good to mention that Snakey diagram, was not our first idea for the chart. Namely, shape of the first chart (which we implemented partially) contained two axis, denoting oceans and types of garbage respectively. In the intersection point of a certain ocean and thrash type, there could be found a rectangle of a certain size (proportional to the amount of thrash in the ocean). This provided nice way to compare amount of garbage in oceans. However, this chart looked a bit simple and not too pleasant to interact with, therefore we decided to change it for a more complex, but more interesting version.

3.2 Expeditions

First part of visualization for expedition web page, was related to projecting Expeditions and Expedition Points, to our map. Moreover, we provided a functionality for a certain Expedition and certain Expedition Point to be selected, when the mouse is hovered over. This can be seen at Figure 5.

Considering further design, we realized that this data could be represented as hierarchies, where some nodes are nested within some other nodes and could form the multiple levels nesting. (I.e. we have set of Expedition Points belonging to an Expedition, and set of Expeditions belonging to an ocean.) To demonstrate the relative indicators of expedition data ocean-wise, we created a three-leveled hierarchies. Its top level (root-node) stands for an ocean, its descendants are expeditions and the leaf nodes represent expedition points. The size of every parent node is determined by the total set of its descendants.

Here we also implemented linking between Expeditions and circle hierarchy. Namely, when the mouse is hovered over the certain circle in the hierarchy, corresponding object is selected on the map. This can be observed on Figure 6.

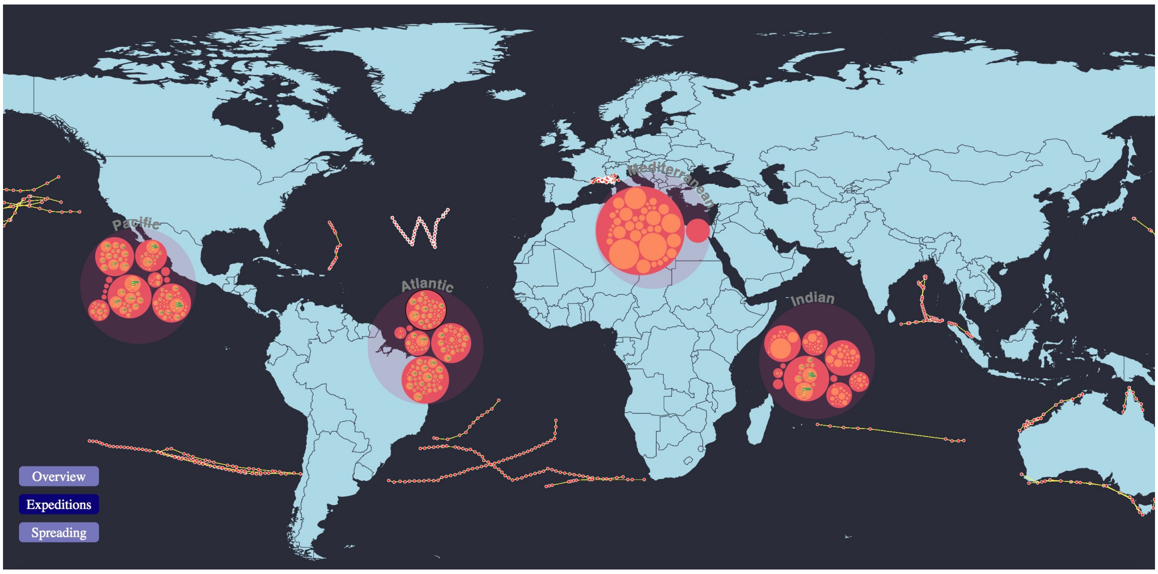

Figure 5: Showing projection of expedition on a map, with possibility of doing selections.

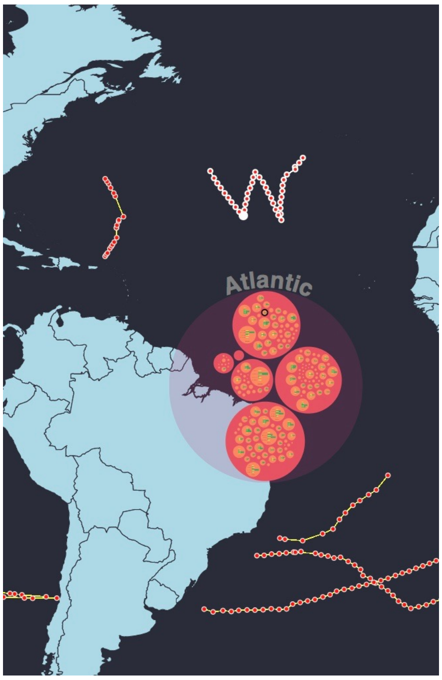

Figure 6: Showing functionality, where by hovering over the certain circle in the hierarchy, corresponding object on the map is selected.

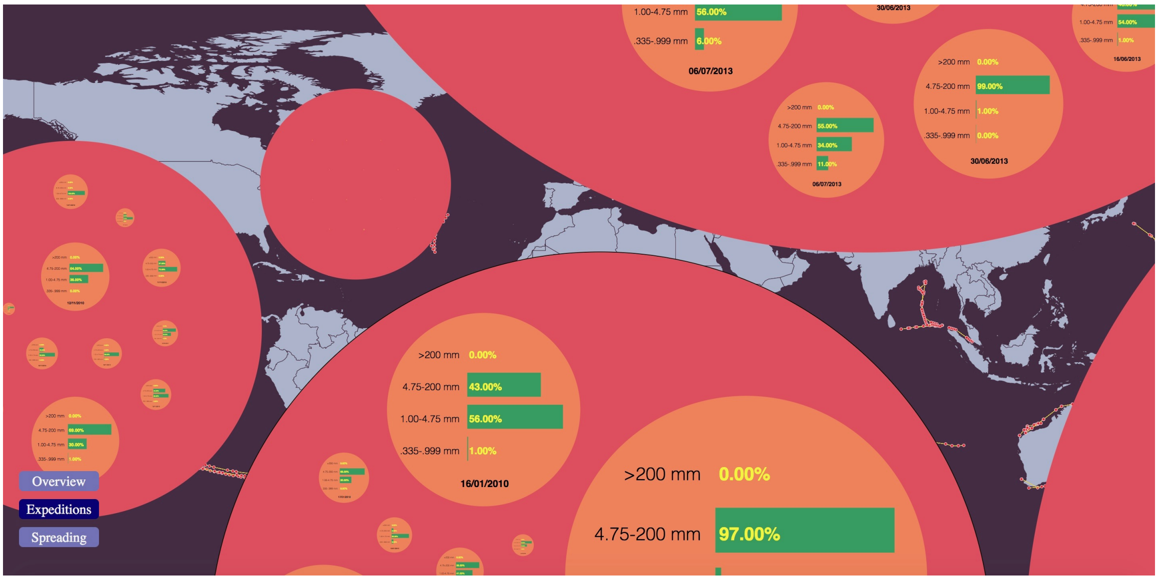

Figure 7: Showing horizontal bar charts, which contain details and the comparative data.

To visualize this data organization, we applied circle packing method (specific layout which is provided by d3), with ability to tightly-organize any representation of groupings and part-to-whole relationships [9]. In order to represent details and the comparative data, horizontal bar charts were integrated into the shapes of leaf-nodes. This can be seen on Figure 7. Detailed use cases diagram can be found on Figure 8.

Initially four layouts are displayed on the screen. As it was mentioned every layout represents the overall information about all available data ocean-wise. Host-circle - is the biggest circle (semantically means and ocean), it contains two levels of nested nodes - expeditions and expedition points. We integrated a bar-chart diagram in order to display the date and the data (garbage weight with respect to garbage size type) sampled at this point during an expedition. Nothing could be seen inside, if the data is missing.

First and second level circles are responsive to user via mouse manipulations. In specific, performing mouveover on first-level nested circles highlights corresponding expedition path; same on the most nested circles highlights relative expedition point on the map. It worth to mention that it works in reverse order for expedition points only. In order to observe the graphical representation of an expedition point a user can mouseclick onto one of the most nested circles. This initiated zoom-in transformation of its geometry, and makes it possible for the user to inspect the date and the diagram. Also we provide the user with ability to move from one point to another just by mouse clicking on its sibling circle. It is important to say that only one circle packing layout is available for interaction during this. We provided two possibilities to get back to expedition visualization initial state. The first one is to mouseclick on previously zoomed circle, the second one is to mouse click on the biggest (root) circle. After this user can chose one of available layouts to interact again. Indeed, all layouts manage their zoom behavior independently.

Its also good to mention that we added label for every packing layout in a form of arc. We expect from the user to be quite familiar with elementary geography, however we decided to add this label just in order to be more explicit.

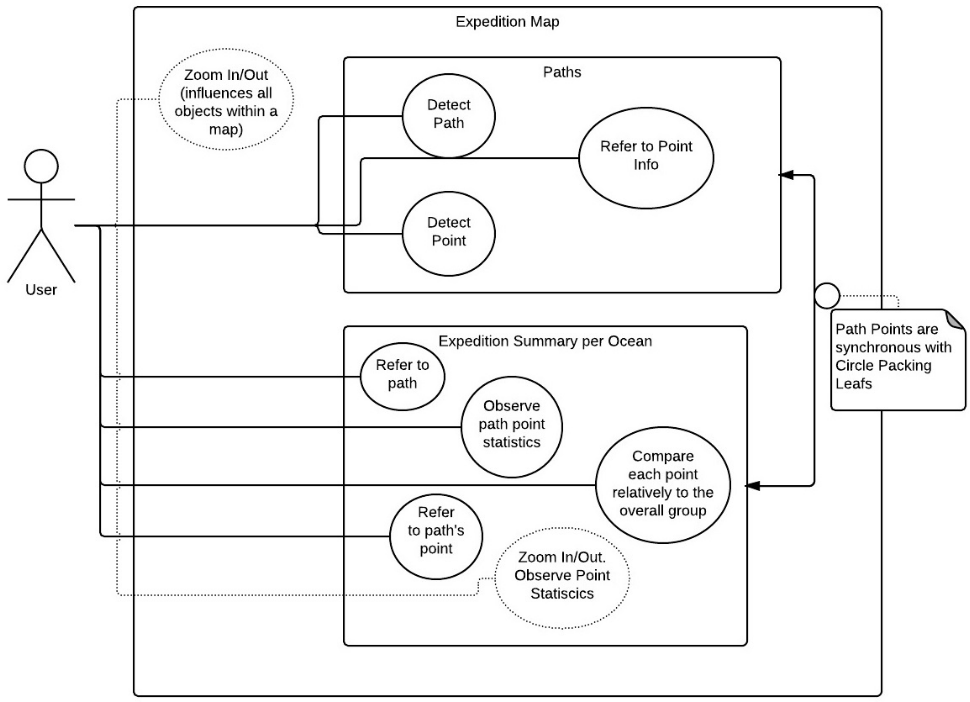

Figure 8: This use case diagram describes a set of actions (use cases) that a user can interactively perform.

3.2.1 Visualization Primitives

- **Path**
  - **Creation:**
    D3 line path data generator is used to generate a polyline. It completes linear interpolation between each point that is included into a path.
  - **Actions:**
    1. **'mouseover'**
       Highlights path and all its points, so that the user can see a trajectory. In some locations there is heavy overlapping, due to intensive expedition data sampling. Indeed, it is important to distinguish one expedition from another when points are placed too densely.
       We performed highlighting by changing stroke and stroke-width styles.
    2. **'mouseout'**
       Cancels highlighting of a path.
  - **Initial Styles:**
    Are represented as css class .line
    * fill: none;

* stroke: yellow;
    * stroke-width: 1.5;
    * shape-rendering: crispEdges;
    * stroke-linejoin: round;
    * stroke-linecap: round;

- **Circle**
  - **Creation:**
    Initial Latitude and Longitude data is projected onto map
  - **Actions:**
    1. **'mouseover'**
       Repeats behavior of Path's 'mouseover' action, in addition it changes a radius of the circle and fills it with white color. Also this synchronizes the responsive circle with one within a packing layout ([10]) and highlights it accordingly.
    2. **'mouseout'**
       Cancels highlighting of a path, a circle and corresponding packing layout node.
  - **Initial Styles:**
    Are represented as css class .circle
    * fill:red;
    * stroke: white;
    * stroke-width:1;

- **Circle Pack Layout**
  - **Creation:**
    The most complex visualization primitive in expedition visualization part. The layout is basically an enclosure nesting diagram for hierarchy representation. Every such layout should be initialized with a hierarchy object. In order to build the hierarchy, objects described in Figure 1, should be combined and then converted to JSON:

```
{
  "name": "atlantic",
  "children": [
    {
      "name": "NAG10",
      "children": [
        {
          "name": "st_point_0_NAG10",
          "total_cd": 81589.02,
          "wd1": 4.45,
          "wd2": 26.83,
          "wd3": 4.23,
          "wd4": 0,
          "date": "09/01/2010"
        },
        {...}
      ],
```

```
            "total_wd1": 47.62,
            "total_wd2": 2482.5899999999997,
            "total_wd3": 34362.61,
            "total_wd4": 0
        },
        {...}
    ]
}
```

Listing 1: Real data example of ocean 1-* expeditions 1-* expedition point relationship (1-* states for one-to-many) in JSON format. Name 'atlantic' represents an ocean name, sequential children block is an array of expeditions; name NAG10 is an expedition id; sequential children block contains an array of expedition points with data; every expedition point contains listed properties: total_cd = cd1+cd2+cd3+cd4, this is the total number of plastic particles detected in this geolocation. total_cd determines the size of a nodes within hierarchy; wd1..wd4 represent weight of every plastic category detected at this geolocation. We use them to compute the relative amount of debris at particular expedition point; date represents expedition date

3.3 Spreading

Through the design of spreading web page, we wanted to demonstrate to the user, possible negative impact of a small action.

[7] By design, this part of the visualization is not intended for an intensive interaction. In the Table 5 it can be seen, which user actions are predicted.

Analysis:	Discover	Present	Enjoy	Annotate	Record	Derive
	✓	✓	✓			
Search:	LookUp	Locate	Browse	Explore		
	✓					
Query:	Identify	Compare	Summarize			
	✓	✓				

Table 5: Design principles for spreading web page.

[7] For visual encoding of design principles, we discussed several ideas on how to represent them. Concerning starting points of thrash spreading, we used circle buttons, to engage a user for interaction. On the other hand, the representation form of spreading dynamics over the time was not evident. Initially, we had an intention to directly color the ocean, so that it could reflect probability for trash to reach a certain point. However, after further discussion, we made a decision that it would be more effective to use some specific graph structure (or more precisely, tree structure). We already have an obvious root node, and we defined its children in a way we described in the section 2. Furthermore, we decided to use *Edge attribute encoding*, to present the probability of garbage reaching the certain point.

When the page is open, "origins", placed all over the map, are observed, as shown on Figure 9. In order to make use of visualization more intuitive, we tried to lean on user's perception system, as well as previous experience. Therefore, all starting points are colored in green (referring to "go" on the traffic lights). In addition to this, the famous "play" sign is drawn over the button. When the user hovers over each of them, color of the button changes, in a way to indicate that interaction is possible. When the user clicks on one of the buttons, it implies that the garbage has been thrown into the ocean, and the spreading starts. At this moment, it is not possible to interact with button any more.

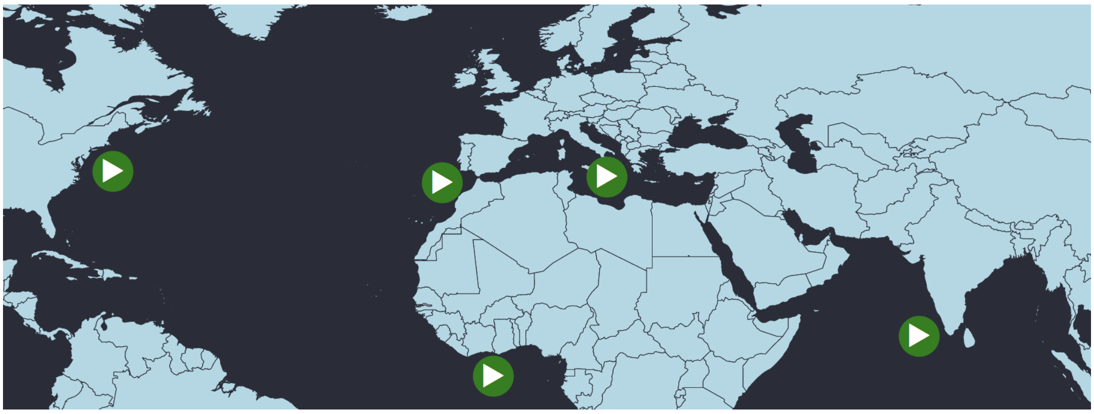

Figure 9: Shows starting positions of spreading web page, which are represented with green buttons (for user to interact by clicking on them).

As the spreading animation starts, we can notice that different groups of edges are expressing different values of attribute. We used color and edge thickness to emphasize some parts of the tree, while neglecting the others. This can be observed on Figure 10. More concretely, edges connected to points of high probability are colored red and are generally thicker. On the other hand, edges connected with low probability points, are colored in gray, generally thinner and less visible (they fit to the background, which is also dark). This part of visualization is designed to be non-interactive, so that while animation is running, user can focus on it.

Once the animation is finished, user can further inspect it, as it stays on the screen. However, at this moment the button at starting point, becomes active again. This functionality can be observed at Figure 11. In order to continue other points exploration (by clicking on the button), it is possible to remove complete tree structure from the screen. Here, again suggestively, we decided to color the button in black color. Moreover, there is a `DEL` sign written over, so that interaction of a user is facilitated.

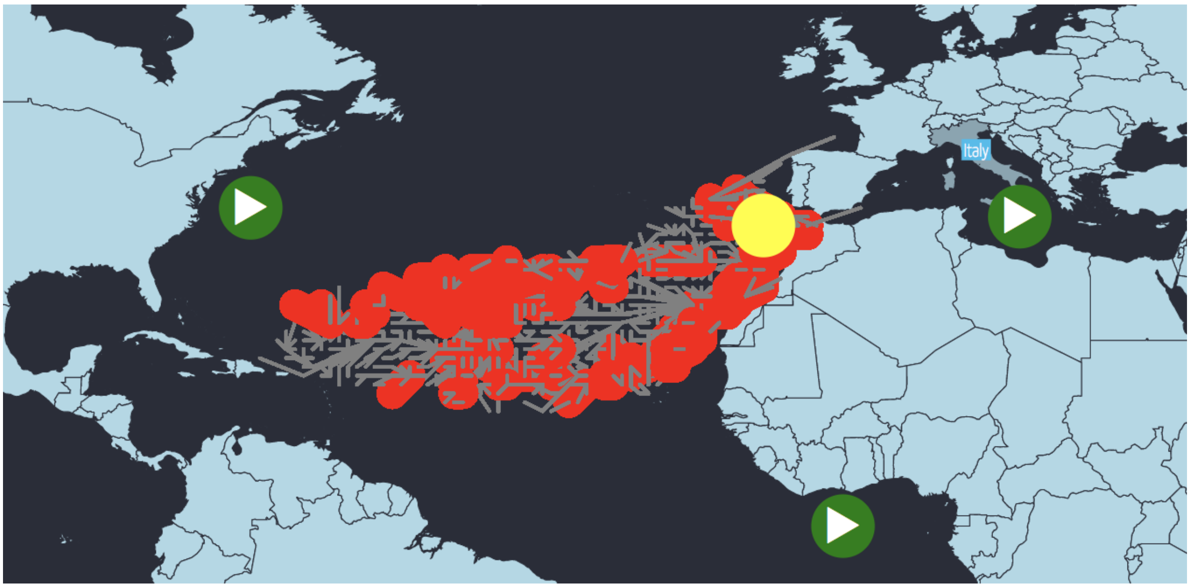

Figure 10: Shows screen shot of spreading animation, where different group of edges are expressing different values of probability attribute.

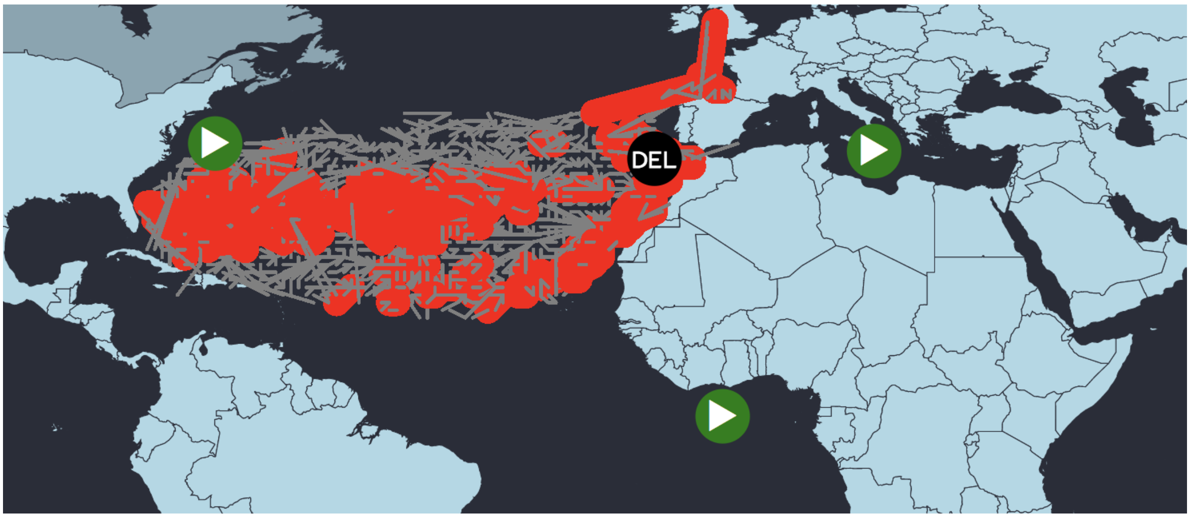

Figure 11: Shows functionality, where by clicking on the button, denoted with `DEL`, it is possible to remove complete tree structure from the screen.

4 Implementation Details

4.1 Map

We used `D3.js` and `JQuery` to plot Map with zoom/pan and rollover label. It was displayed within *map-holder* div element, to allow easier placement on the page. In order to show the map as small as possible, without leaving white space at edges, `Initialzoom` functionality is implemented. `ScaleExtent` functionality is used to ensure that map can not be zoomed out to smaller than initial value. `TranslateExtent` functionality is used to make sure map can not be moved beyond inside of div. Country name labels are placed using `Path.centroid`. Moreover, on resize of the window, SVG element gets resized as well, `scaleExtent` and `translateExtent` get recalculated and the map is zoomed and panned to preferred initial setting.

4.2 Overview Page

The trickiest part of implementing the overview web page, was creating a Sankey diagram. Because of the difference between version 3 and version 4 of `D3.js` library API, it was problematic to find working examples.

Other challenging point of implementation, was to make chart work for two different types of values (weights and pieces), which are controlled using two buttons.

4.3 Expeditions

Implementation of this part consists of two substantial units, namely - (I)expeditions paths serialization and visualization and (II)circle packing layout creation and statistics demonstration. They both require specific data preprocessing with respect to `D3` library API.

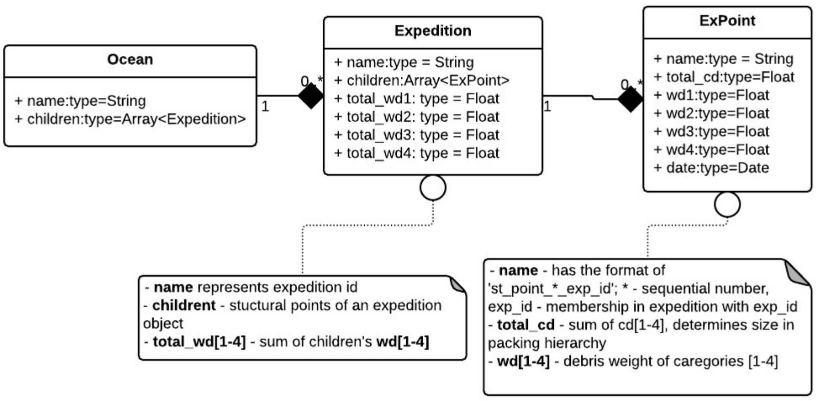

Figure 12: Basic data types and their relation

- Data read performs:
  1. Basic read and rows format of input csv files; dictionary creation of key('name of ocean') - value(array of expeditions') items.
  2. Build 'Ocean' objects (Figure 12)
  3. Compute average geolocation: Longitude and Latitude average with empirical correction on it. It is further used to position circle packing layouts.
  4. Order 'Ocean' objects - important in visualization part
- Circle packing layout setup performs:
  1. Hierarchy creation; every 'Ocean' object is transformed into JSON, and based on it a hierarchy of nodes is generated.
  2. Circle packing layout initialization implies that hierarchy exists already
  3. Layout positioning and circle element assignment at every nested level of hierarchy
  4. Individual interactive behavior setup (zooming, mouse events handling)
  5. Creation of horizontal bar-charts for all leaf nodes that contain data (no bar-chart is created when data is missing)

  In our project we build four entities of circle packing layouts for Atlantic, Pacific, Indian oceans and for Mediterranean sea.
- Path poly lines setup performs
  1. (Longitude, Latitude) coordinates on map projection. It recalculates them with respect to the map layer
  2. Data binding and application of d3's standard line generator with linear interpolation in order to obtain an expedition curve
  3. Creation of circle primitives. Each represents an expedition point keeps some statistics behind. Circles overlay paths in order to make visually an a composite expedition unit
  4. Setup of an interactive behavior for paths and points

4.4 Spreading

One challenging point in implementation, was to project all points to the map correctly.

Considering tree structure, we soon realized that we do not need a "convectional" tree structure with nodes and edges. It turns out that drawing only edges makes visualization of spreading better. Furthermore, in special cases (described in section two), we do not have traditional tree. For this reasons, we decided to implement the structure, not by using existing libraries, but generally as a set of lines with given coordinates.

4.5 Tools and Libraries

For implementing our visualization we used `JavaScript` programming language. Besides built-in functions, we used `D3.js`([11]) library for advanced visualizations. For the implementation of map, we used additionally `JQuery` package.

In a data preprocessing step we used `python` programming language with `jupyter notebook` environment. Some of the additional packages we used here are `pandas` and `numpy` as well as `geopy` (to calculate geographical distance between two points on the map, given their latitudes and longitudes).

5 Evaluation

The final product of our visualization is relatively simple, but very interactive and intuitive demonstration of two data sets.

By putting ourselves in a role of a user, we found that it is interesting to explore data this way. Moreover, we learned a lot of new insights. E.g. we were shocked to get to know the amount of marine debris over the seas. It was interesting to detect that some oceans are 'fed' with heavy weight debris, while in the same time the number of plastic pieces is relatively small. And other ones contain large amount of plastics, but mainly in small pieces. This are very interesting observations, which show how different type of data can give completely different impression. Another interesting insight can be observed on spreading visualization. Namely, we would expect that the biggest probability for garbage to be found is closer to the starting point. However, because of the sea currents, some points are more likely to be reachable than others, which are closer. This information we already mentioned in the introduction (that all garbage ends up in 5 gyres), however it is still impressive to see it on the visualization.

Our main goal was to make the problem of marine plastic pollution, closer to large audience. We tried to do that by informing them in an interesting and interactive form about current pollution and efforts. Moreover, using spreading visualization, we tried to more actively involve them (they could possibly identify with the locations closest to where they live). We can say that this demonstration, responds to the questions we posed in the beginning. However, there could be done more. E.g. considering expeditions data set, we discussed idea to implement a time-line, since for each measurement we have corresponding time. Main goal would be to make an interactive linked view. I.e. for selected time-range points on map which were measured during that time could be for example highlighted, or we can simply display statistics on measured garbage amounts in that time-range, etc. To give user more interesting results on plastic pollution, it could also be useful, in the future, to use additional data sets.

6 Conclusions

Our initial goal was to visually represent plastic pollution information detected during different expeditions, and to demonstrate the garbage spreading over the time.

Several complex and vivid data visualizations were implemented and are available for the user to play around. While working on the project, we learned in practice how to use modern visualization tools effectively, in specific `D3.js`. Also by completing the pollution map, we learned that severity of plastic pollution is worse than many people realize. From a technical point of view we can conclude that design and implementation from the greenfield require a lot of experience not only in programming but also in design and in human perception understanding. But as inhabitants of the earth, even though we do not see the consequence right away, we are making the world much more vulnerable place for our future generations.

References

[1] C. Moore. Encyclopedia Britannica - Plastic Pollution. [Online]. Available: https://www.britannica.com/science/plastic-pollution

[2] T. Lonne. How Ocean Pollution Affects Humans. [Online]. Available: https://www.divein.com/articles/ocean-pollution/

[3] L.-M. Lebreton, S. Greer, and B. J.C., "Numerical modelling of floating debris in the world's oceans," *Marine Pollution Bulletin*, 2012.

[4] Plastic Marine Pollution Global Dataset. [Online]. Available: https://figshare.com/articles/Plastic_Marine_Pollution_Global_Dataset/1015289

[5] E. v. Sebille, M. H. England, and F. Gary, "Origin, dynamics and evolution of ocean garbage patches from observed surface drifters," *Environmental Research Letters*, 2012.

[6] Plastic Adrift Data Set. [Online]. Available: http://plasticadrift.org/?lat=34lng=-22.9center=-5startmon=jandirection=fwd

[7] B. K. Maël, "Data Vizualisation course notes - COM-480," *EPFL*, 2017.

[8] d3-sankey library API reference. [Online]. Available: https://github.com/d3/d3-sankey

[9] d3-hierarchy library API reference. [Online]. Available: https://github.com/d3/d3-hierarchy

[10] Zoomable Circle Packing. [Online]. Available: https://bl.ocks.org/mbostock/7607535

[11] d3 library API reference. [Online]. Available: https://github.com/d3/d3/wiki/API-Reference

7 Peer assessment

Assessment by Ana:
To Sergii:

- Preparation: **yes**
- Contribution: **yes**
- Respect for others' ideas: **yes**
- Flexibility: **yes**

To Mohammad:

- Preparation: **yes**
- Contribution: **yes**
- Respect for others' ideas: **yes**
- Flexibility: **no**

Assessment by Sergii:
To Ana:

- Preparation: **yes**
- Contribution: **yes**
- Respect for others' ideas: **yes**
- Flexibility: **yes**

To Mohammad:

- Preparation: **no**
- Contribution: **yes**
- Respect for others' ideas: **yes**
- Flexibility: **no**

Assessment by Mohammad:
To Ana:

- Preparation: **yes**
- Contribution: **yes**
- Respect for others' ideas: **yes**
- Flexibility: **yes**

To Sergii:

- Preparation: **yes**
- Contribution: **yes**
- Respect for others' ideas: **yes**
- Flexibility: **yes**