University of Pisa

Department of Computer Science

Visual Analytics

Tracking Suspicious Fishing Activity: Solving the VAST Challenge 2024

Matilde Contestabile

Student ID: 639307

m.contestabile1@studenti.unipi.it

Abstract

This report describes the design and implementation of an interactive visual analytics system developed for the VAST Challenge 2024 (Mini-challenge 2). The system enables exploration of vessel activity and fish transactions, allowing users to identify patterns, anomalies, and potential illegal behavior. We detail the dataset, the analytical approaches applied during exploratory data analysis, and the rationale behind the visual design choices, including color, layout, and interaction strategies. Through dynamic visual representations, the system supports both high-level overviews and detailed investigation, facilitating the discovery of insights and supporting analytical tasks in complex data.

Academic Year 2024/2025

Table Of Contents

1 Introduction

1.1 Project Overview and VAST Challenge Context

This project addresses the analytical goals of the **Mini-Challenge 2 (MC2)** from the **VAST Challenge 2024** (available for consultation at `https://vast-challenge.github.io/2024/index.html`). The VAST Challenge is a well-established international competition focusing on advanced visual analytics. The 2024 edition is set in the fictional nation of Oceanus, where a vibrant commercial fishing industry is facing threats from unethical practices by certain actors. The *Challenge Overview* recites as follows:

> Welcome to Oceanus, an island nation with a healthy market for commercial fishing. Most companies in the region are united in following regulations and implementing sustainable fishing practices. But there are a few companies who are willing to cross ethical lines to increase their catch and their profits. Luckily, FishEye International maintains a watchful eye on fishing data. Their dedicated analysts have been processing data from various sources into a knowledge graph that they call CatchNet: the Oceanus Knowledge Graph.

In this context, **Mini-Challenge 2 (MC2)** focuses on analyzing the behavior of vessels operating within Oceanus using multiple datasets, including transponder pings, harbor visit records, and transaction logs. The central goal is to investigate **illegal fishing behaviors** by the company **SouthSeafood Express Corp** and to develop visualizations that support **anomaly detection in vessel movements and supply chains**. Here's the *overview* specific to the mini-challenge:

> In Oceanus, island life is defined by the coming and going of seafaring vessels, many of which are operated by commercial fishing companies. Typically, the movement of ships and goods are a sign of Oceanus's healthy economy. But mundane routines can be disrupted by a major event.
>
> FishEye International has discovered that SouthSeafood Express Corp was engaged in illegal fishing, prompting the need for advanced analysis. As part of this challenge, analysts must investigate this event using CatchNet data to uncover behavioral patterns, track fish product movements, and support future monitoring.

More specifically, the mini-challenge requires to address the following research questions through a series of targeted visual workflows and analytical strategies, with an emphasis on interactivity and clarity:

1. **Cargo Attribution:** Given the lack of direct vessel identifiers in port transaction records (due to wrong purchase of records by FishEye analysts), can we visually and analytically associate cargo deliveries with specific vessels? What seasonal or regional trends emerge in fish exports?
2. **Illegal Behavior Detection:** How do the trajectories and port interactions of SouthSeafood Express vessels differ from compliant vessels? When and where did violations occur?
3. **Behavioral Pattern Matching:** Are there other vessels whose behavior mirrors that of SouthSeafood Express? Can similar illegal activities be inferred?
4. **Post-Incident Trends:** Following the discovery of illegal fishing, have commercial fishing patterns across Oceanus shifted? Are new anomalies emerging?

Answers are available at Section 6.

1.2 Repository and Resources

The full implementation is publicly available in the GitHub repository:

`https://github.com/matildeec/VisualAnalytics2025`

The repository includes:

- `data-understanding/` – Notebook-based data inspection and cleaning scripts
- `platform/` – Visualization platform built with D3.js and Altair
- `notebooks/` – Exploratory and analytical notebooks addressing each research question
- `data/` – Cleaned and raw JSON graph files

To install, clone the repository and install the required dependencies:

```
git clone https://github.com/matildeec/VisualAnalytics2025.git
cd VisualAnalytics2025
pip install -r requirements.txt
```

The project is fully reproducible and structured for modular exploration and extension.

2 Data Understanding and Preprocessing

This section outlines the data exploration and cleaning process carried out for the VAST Challenge 2024 Mini-Challenge 2. Using a data mining approach and Python libraries, we prepared the knowledge graph dataset for visual exploration and anomaly detection, specifically, in a format suitable for JavaScript-based applications.

2.1 Data Sources and File Structure

The challenge dataset consists of multiple files, with the central one being a JSON-encoded knowledge graph representing vessel activities, harbor transactions, and commercial fishing behaviors. Supplementary files provide metadata and geospatial context.

File Name	Description
`mc2.json`	Main knowledge graph, structured as a multigraph with typed nodes and edges representing entities and events (e.g., vessels, ports, transactions, pings).
`Oceanus Geography Nodes.json`	Metadata for geographic locations (e.g., ports, reefs, regions).
`Oceanus Geography.geojson`	Geospatial file for mapping entities in Oceanus.

Table 1: Overview of provided data files

To support understanding of the knowledge graph, the document `VAST2024 - MC2 Data Description.docx` is provided, detailing the graph structure and entity semantics.

2.2 Graph Structure Analysis and Preprocessing

The knowledge graph contained in `mc2.json` was imported using the `networkx` library, resulting in a directed multigraph. Nodes and edges include a `type` attribute, which was used to categorize and filter graph elements. Both nodes and edges were extracted into separate Pandas DataFrames to enable structured analysis.
Initial preprocessing involved removing metadata fields irrelevant to downstream analysis, specifically: `[_last_edited_by, _last_edited_date, _date_added, _raw_source, _algorithm]`.
Next, nodes (entities) and edges (events) were grouped by their `type`. Within each group, columns containing only missing values were discarded, and only relevant attributes were retained to support visualization and modeling tasks.
A summary table of the cleaned data files intended for use in the JavaScript implementation is provided at the end of this section.

2.2.1 Node Overview and Distribution

The knowledge graph contains a diverse set of node types. Table 2 summarizes the most relevant node types by category, showing the number of instances and representative attributes retained after cleaning.

Category	Node Type	Count	Attributes
Document	Entity.Document.DeliveryReport	5307	qty_tons, date
Vessel	Entity.Vessel.FishingVessel	178	name, flag_country, company, tonnage, length_overall
	Entity.Vessel.CargoVessel	100	name, flag_country, company, tonnage, length_overall
	Entity.Vessel.Ferry.Passenger	3	name, flag_country
	Entity.Vessel.Ferry.Cargo	2	name, flag_country
	Entity.Vessel.Tour	6	name, flag_country
	Entity.Vessel.Research	2	name, flag_country
	Entity.Vessel.Other	5	name, flag_country, length_overall
Location	Entity.Location.Point	12	name, description, activities, kind
	Entity.Location.City	6	name, activities, kind
	Entity.Location.Region	6	name, description, activities, kind, fish_species_present
Commodity	Entity.Commodity.Fish	10	name

Table 2: Updated Node Types with Attributes in the Knowledge Graph

2.2.2 Edge Types and Event Semantics

Edges in the knowledge graph represent interactions and events between entities, providing both temporal and relational context to the data. Table 3 summarizes the original edge types, including their counts, source and target entities, and retained attributes.

Event Type	Count	Source	Target	Attributes
Event.TransportEvent.TransponderPing	258,542	Entity.Location	Entity.Vessel	time, dwell
Event.Transaction	5,307	Entity.Document	Entity.Commodity	date
Event.Transaction	5,307	Entity.Document	Entity.Location	date
Event.HarborReport	2,487	Entity.Vessel	Entity.Location	date, data_author

Table 3: Original edge types with counts, sources, targets, and attributes.

From here, two critical issues become apparent that affect readability and understanding of the data. First, for **Event.HarborReport**, the edge direction is counterintuitive: while the information originates from portmasters, the current structure connects vessels (sources) to ports (targets). Reversing this direction would better reflect data provenance and better match with **Event.TransportEvent.TransponderPing**. Second, **Event.Transaction** edges are redundant: each delivery report is linked to both a location and a commodity via two separate edges, resulting in duplication and unnecessary complexity.
To address these issues, we redesigned the schema. Information about commodities, including species and quantities, has been moved into the delivery report entity itself, while **Event.Transaction** edges now exclusively connect delivery reports (sources) to locations (targets). The updated structure is summarized in Table 4.

Event Type	Count	Source	Target	Attributes
Event.TransportEvent.TransponderPing	258,542	Entity.Location	Entity.Vessel	time, dwell
Event.Transaction	5,307	Entity.Document	Entity.Location	date
Event.HarborReport	2,487	Entity.Location	Entity.Vessel	date, data_author

Table 4: Updated edge types

2.2.3 Cargo Attribution via Suspected Vessels Tagging

To address the challenge of associating cargo deliveries with specific vessels despite missing direct identifiers in the port transaction records (see Question 1 in 1.1), we introduced an attribute called `suspected_vessels` within the transactions dataset which was derived in an analytical way. This attribute records, for each transaction, a list of vessels potentially responsible for the cargo. The goal was to infer likely vessel–cargo associations based on spatiotemporal docking information and the presence of fish species in regions. The logic underlying this inference can be formalized as:

$$
\begin{aligned}
&\textit{Vessel is either 'CargoVessel', 'FishingVessel' or 'Other'} \\
&\wedge\ \textit{Vessel is docked in a harbor where a fish species was exported (within 1 day)} \\
&\wedge\ \textit{The same species is present in a region earlier visited by the vessel} \\
&\implies \textit{The fish species is likely the vessel's cargo.}
\end{aligned}
$$

Applying this rule, we flagged transactions meeting these conditions and annotated them with the corresponding `suspected_vessels`. A visual summary of the logic is provided in Figure 1.

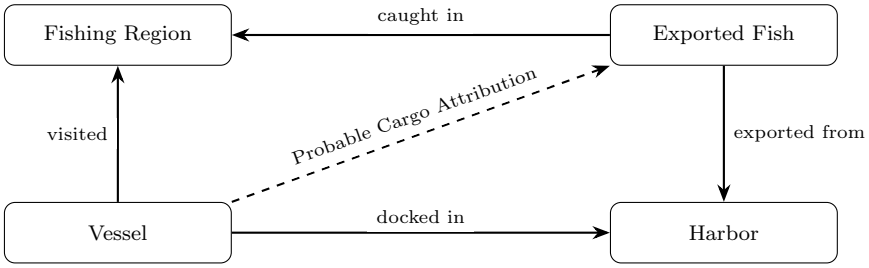

Figure 1: Flowchart showing derivation of the `suspected_vessels` attribute.

2.3 Resulting `.json` Files

Table 5 summarizes the cleaned and structured `.json` files generated from the original knowledge graph after preprocessing. These files are optimized for use in the JavaScript-based implementation and support efficient querying since together they construct a database with primary (underlined) and foreign keys.

File Name	Description	Keys
`vessels.json`	Vessel entities	`id`, `vessel_type`, `name`, `company`, `flag_country`, `tonnage`, `length_overall`
`commodities.json`	Traded fish	`id`, `name`
`documents.json`	Delivery reports	`id`, `commodity`, `qty_tons`
`locations.json`	Locations	`id`, `name`, `location_type`, `kind`, `description`, `activities`, `fish_species_present`
`transponder_pings.json`	Vessel GPS logs	`source`, `target`, `time`, `dwell`
`harbor_reports.json`	Port-exit records	`source`, `target`, `date`, `data_author`
`transactions.json`	Cargo exchanges	`source`, `target`, `date`, `suspected_vessels`

Table 5: Cleaned `.json` files and their key attributes. Underlined keys indicate primary identifiers.

To further facilitate the representation of vessel trajectories, a dedicated file named `trajectories.json` was created. It is structured as a dictionary, where the keys are vessel IDs and the values are ordered lists of transponder pings represented as JSON objects.

3 Exploratory Data Analysis

We conducted an initial exploratory data analysis (EDA) on the cleaned data using a combination of Python libraries such as `pandas`, `matplotlib`, `altair`, and `seaborn`. The purpose of this analysis was to uncover patterns in vessel activity, identify potential anomalies, and provide a foundation for more sophisticated visualizations later on.

3.1 Analytical Analysis of vessel activity

To effectively interpret vessel behavior, it was first necessary to understand the geographical context of the dataset. Certain regions are in fact designated as ecological preserves, where fishing activity is strictly prohibited. Vessels, however, were observed operating across both legal fishing grounds and these protected zones. Additionally, the dataset provides the list of species present in every region, allowing us to identify which species are exclusive to protected preserves. Using this information, we derived an analytical classification of *illegal species*, defined as those found exclusively within protected preserves. Table 6 summarizes this mapping: species highlighted in red appear only in illegal zones and therefore serve as indicators of potential illegal fishing activity.

Fish Species	Cod Table	Wrasse Beds	Tuna Shelf	Ghoti Preserve	Nemo Reef	Don Limpet Preserve
gadusnspecificatae4ba	✓					
piscesfrigus900	✓	✓	✓		✓	✓
habeaspisces4eb	✓	✓	✓	✓	✓	✓
labridaenrefert9be		✓		✓	✓	
piscessatisb87				✓	✓	✓
piscisosseusb6d				✓		
thunnininveradb7			✓		✓	✓
piscesfoetidaae7						✓
piscissapidum9b7			✓			

Table 6: Matrix of fish species presence across regions. Rows in red indicate species found exclusively in ecological preserves (illegal). Shaded columns mark the illegal fishing zones.

After establishing a clear mapping of legal versus illegal zones and species, we turned our attention to temporal patterns in vessel behavior. Specifically, we examined three features: *dwell time* (how long a vessel remained near a given location), *gaps between transponder pings*, and *harbor visit frequencies*. The rationale was straightforward. Prolonged dwell times may signal suspicious activity, such as covert fishing or offloading. Unusual gaps in transponder signals could indicate attempts at evasion, while irregular harbor visits might point to atypical supply or unloading patterns.

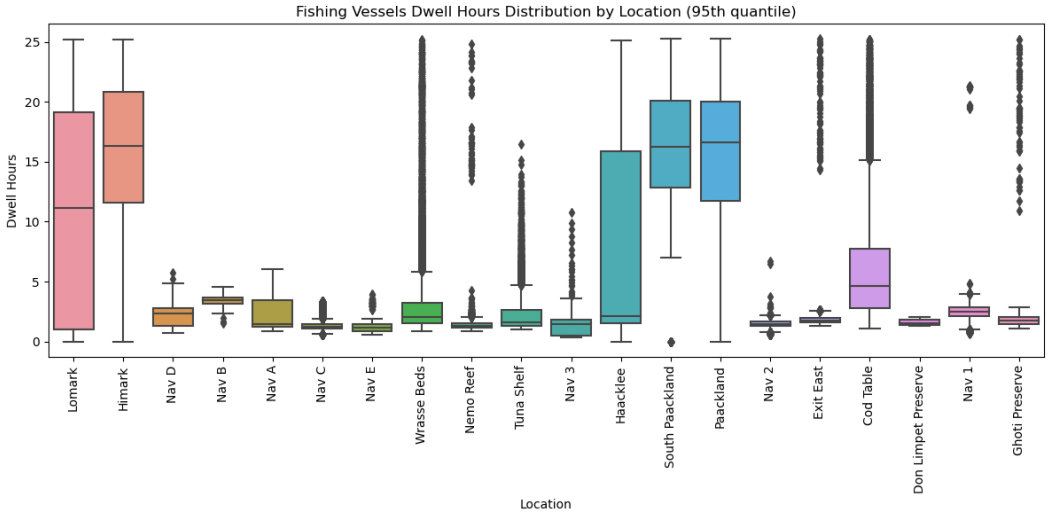

Figure 2: Dwell time of fishing vessels by location.

Of these features, however, the latter two proved less reliable as standalone indicators. Coverage gaps were common but did not consistently align with illegal fishing zones, and variations in harbor visit frequency were not necessarily suspicious when taken in isolation. By contrast, the dwell time analysis provided the clearest signals of anomalous vessel behavior and thus emerged as the most informative exploratory measure. Figure 2 illustrates these dwell time patterns across different locations.
The boxplots make clear that vessels indeed lingered significantly longer at certain locations than expected. While many values fell within a normal operational range, some extreme outliers stand out and warrant further investigation. This insight guided the direction of subsequent visualizations,

where the emphasis shifted toward tracking how vessels move, where they pause, and how these behaviors align with known illegal regions.

3.2 Temporal Analysis of Fish Deliveries

To better grasp the scale of fish deliveries, we aggregated transaction records to track quantities over time. This allowed us to identify seasonal dynamics, peak transaction periods, and potential export trends as requested in Q1. Figure 3 illustrates total transaction volumes for South Paackland as an example case. Noticeable spikes suggest periods of heightened demand, which may align with peak fishing activity or, in some cases, potential illegal operations.
In the visualization, illegal fish species are highlighted in shades of red, while legal species are shown in shades of blue, enabling a direct comparison of their relative contribution to overall trade flows. This perspective not only informed our analysis but also shaped the subsequent design of our visualization approach.

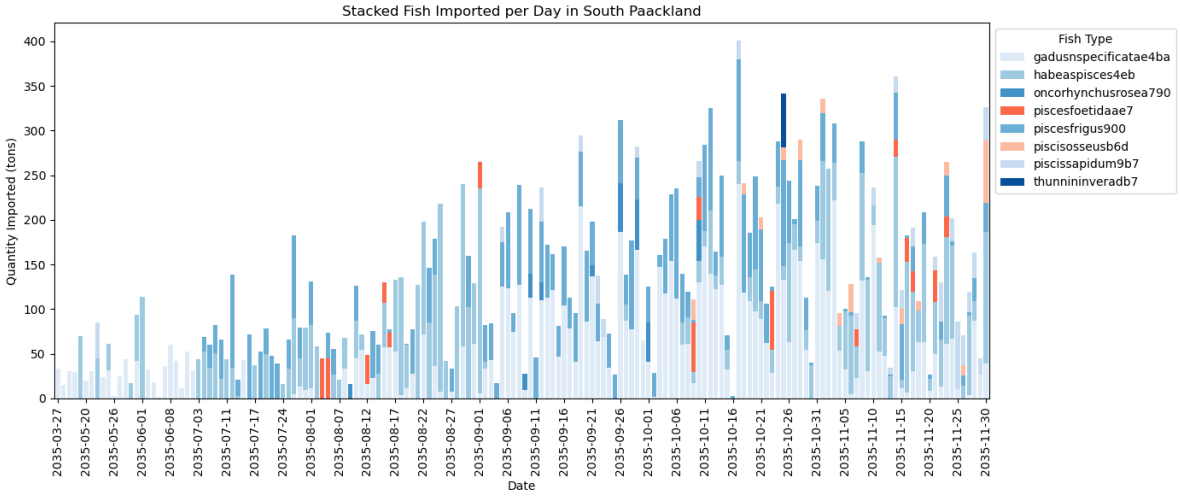

Figure 3: Transaction volume over time, representing fish deliveries. Illegal species are shown in shades of red, while legal species are shown in shades of blue.

3.3 Analytical Analysis of SouthSeafood Express Corp vessels

As part of our EDA, we conducted a focused investigation of the company identified as engaging in illegal activities, *SouthSeafood Express Corp*, as requested by the challenge. Table 7 summarizes the two vessels associated with this company, including their vessel IDs, names, and tonnage.

Vessel ID	Name	Tonnage
snappersnatcher7be	Snapper Snatcher	100
roachrobberdb6	Roach Robber	11,700

Table 7: Vessels associated with SouthSeafood Express Corp.

Both vessels were carefully examined for unusual patterns in their activity, including extended dwell times, abnormal harbor visit frequency, and irregular transaction volumes.
We first established the timeframe of activity prior to the discovery of illegal behavior: *SouthSeafood Express Corp* vessel pings ranged from `2035-02-01` to `2035-05-14`. This period serves as a reference for analyzing changes in behavior, relevant to questions such as those in Q4 regarding post-incident trends: whether commercial fishing patterns across Oceanus have shifted and whether new anomalies are emerging.
To assess whether dwell time and other features are indicative of illegal activity, we generated rankings of top dwellers, vessels with the longest ping gaps, and other relevant metrics. In the ranking of dwell times at illegal locations, `snappersnatcher7be` (Snapper Snatcher) ranked 76th, while `roachrobberdb6` (Roach Robber) did not appear among the top vessels. This suggests that although

Snapper Snatcher spent time at potentially suspicious locations, it was not among the vessels with the longest dwell times, indicating that further analysis is required to determine whether this behavior is significant.
Similarly, in the analysis of ping gaps, Snapper Snatcher ranked 173rd and Roach Robber 175th. Neither vessel exhibited substantial gaps in their transponder signals, implying that ping gaps alone are not a reliable indicator of evasion or suspicious activity in this context.
More interesting insights emerged from examining vessel routes based on transponder pings. Snapper Snatcher made multiple visits to *Exit East*, a region designated for deep-sea fishing. While this does not immediately stand out compared to general vessel activity, it raises questions about whether this route reflects typical operational behavior or unusual activity. Further investigation, particularly with complete route visualizations, will help clarify these patterns.
While detailed considerations of vessel behavior will be presented later as the full visualizations are presented, we provide the initial Altair plot that was generated to explore these routes.

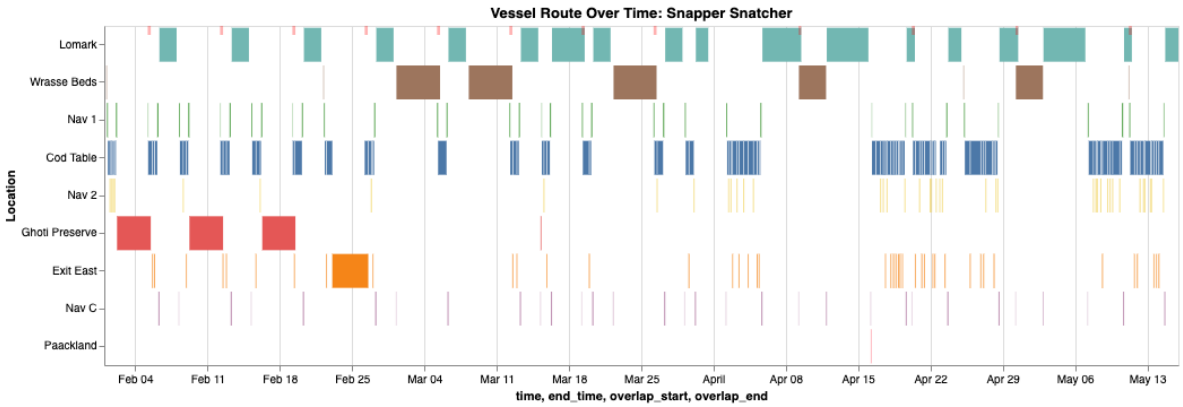

Figure 4: Route of `Snapper Snatcher`

4 Design and Architecture of the Visual Analytics Interface

The visual analytics interface was designed to support anomaly detection, pattern recognition, and hypothesis generation with the ultimate goal of answering the four research questions reported in Section 1.1. Guided by established principles of visual encoding, interaction design, and narrative visualization[1], the design process consisted of three main stages. First, we surveyed the state-of-the-art tools to gain inspiration for tracking fishing vessels. Next, we identified the key visual variables necessary to address the research questions posed by the VAST Challenge. Finally, these design decisions were translated into a prototype using Figma, which then served as the blueprint for the implementation of the interactive system in JavaScript.
This section provides a detailed account of the visual analytics system we developed. We begin by reviewing existing platforms that informed our design choices, then explain the rationale for structuring the interface into distinct pages and how this layout supports user exploration and narrative flow. Finally, we describe the content and visual variables of each page, illustrating how they work together to answer the challenge's research questions.

4.1 State-of-the-Art Interfaces

To first gain inspiration, we explored existing platforms addressing similar challenges:

- **VesselFinder**[2]: a real-time vessel tracking platform that visualizes vessel locations worldwide using AIS data, providing an intuitive overview of maritime activity.
- **GlobalFishingWatch**[3]: a platform for visualizing global fishing activity, allowing users to monitor vessel movements and identify potential illegal fishing operations.

[1] Wilke, C. (2019). *Fundamentals of Data Visualization*. Available at `https://clauswilke.com/dataviz/`
[2] Available at `https://www.vesselfinder.com/`
[3] Available at `https://globalfishingwatch.org/map`

These platforms shaped our initial thinking, particularly in how vessel routes could be represented on a geographic map. However, the structure of our dataset imposed important constraints that made it impossible to reproduce the same level of detailed route visualizations. Although we had geographic coordinates available to reconstruct a map of Oceanus (via the provided `.geojson` file of Oceanus), the vessel GPS pings did not form a continuous trajectory. Instead, they resembled a series of discrete points – often multiple pings from the same location – linked only by timestamps and dwell times. As a result, we lacked the granular path data necessary to accurately reconstruct "real ocean routes" between locations *A, B*, and *C*.
Rather than forcing an incomplete or misleading geographic reconstruction of exact travel paths, we chose to design visualizations conceptually similar to the static plots created in our Jupyter Notebooks as part of EDA (see Section 3), focusing on patterns and trends in vessel pings – such as timing, frequency, and distribution. This approach allowed us to emphasize meaningful behavioral patterns over potentially inaccurate (and ultimately insignificant) geographic details.

4.2 Interface Structure and Narrative Flow

We explain why the interface was structured into two/three distinct pages, how each page supports a specific stage of the user's analytical journey, and how this multi-page layout promotes clarity, overview, and progressive disclosure of detail.
Interactive filtering, brushing, and zooming further enhanced the perceptual scalability of the tool, aligning with Shneiderman's Visual Information-Seeking Mantra: *"Overview first, zoom and filter, then details on demand"*[4].
To support the investigative narrative of illegal fishing operations, we adopted the paradigm of **reader-driven narrative visualization**, as described in Segel and Heer's taxonomy [**?**]. This approach empowers users to explore the data at their own pace, guided by tools rather than a predefined story path. The key components include:

- **Interactive Timelines:** Allow analysts to explore vessel behavior over time and detect anomalous sequences.
- **Linked Views:** Harbor visit timelines, transaction plots, and geospatial maps are linked to enable coordinated interaction across multiple data dimensions.
- **Dynamic Tooltips:** Provide contextual data (e.g., vessel metadata, transaction details) upon hovering, aiding micro-level analysis.
- **Anomaly Highlighting:** Automatically flags suspicious patterns such as extended dwell in protected areas or unmatched cargo deliveries.

Although the platform is mostly reader-driven, we also integrated elements of **author-driven storytelling** in the form of preloaded bookmarks and filters highlighting the behavior of known violators (e.g., SouthSeafood Express Corp). This hybrid approach balances analyst freedom with investigative focus.

4.3 Visual Encoding and Interaction Design

We describe the key visual variables (color, shape, size, position, animation) and interaction mechanisms chosen for each page, explaining how these choices address the VAST Challenge research questions and support anomaly detection, pattern recognition, and hypothesis generation.

5 Use Case Example

Provide a step-by-step walk-through of how an analyst might use your system to complete a specific analytical task relevant to the challenge questions.
Using a concrete example (e.g., a SouthSeafood Express Corp vessel), we illustrate the expected user workflow: how an analyst can move through the pages, filter and interact with the data, and combine multiple views to uncover potential illegal behavior.

[4]See `https://data.europa.eu/apps/data-visualisation-guide/`

6 Research Questions

List the specific research questions you are addressing from the MiniChallenge. For each question, include a short description of how you plan to approach it.

> FishEye analysts need your help to perform geographic and temporal analysis of the Catch-Net data so they can prevent illegal fishing from happening again. Your task is to develop new visual analytics tools and workflows that can be used to discover and understand signatures of different types of behavior. Can you use your tool to visualize a signature of SouthSeafood Express Corp's illegal behavior? FishEye needs your help to develop a workflow to find other instances of illegal behavior.
>
> 1. FishEye analysts have long wanted to better understand the flow of commercially caught fish through Oceanus's many ports. But as they were loading data into Catch-Net, they discovered they had purchased the wrong port records. They wanted to get the ship off-load records, but they instead got the port-exit records (essentially trucks/trains leaving the port area). Port exit records do not include which vessel that delivered the products. Given this limitation, develop a visualization system to associate vessels with their probable cargos. Which vessels deliver which products and when? What are the seasonal trends and anomalies in the port exit records?
> 2. Develop visualizations that illustrate the inappropriate behavior of SouthSeafood Express Corp vessels. How do their movement and catch contents compare to other fishing vessels? When and where did SouthSeafood Express Corp vessels perform their illegal fishing? How many different types of suspicious behaviors are observed? Use visual evidence to justify your conclusions.
> 3. To support further Fisheye investigations, develop visual analytics workflows that allow you to discover other vessels engaging in behaviors similar to SouthSeafood Express Corp's illegal activities? Provide visual evidence of the similarities.
> 4. How did fishing activity change after SouthSeafood Express Corp was caught? What new behaviors in the Oceanus commercial fishing community are most suspicious and why?
>
> Note: the VAST challenge is focused on visual analytics and graphical figures should be included with your response to each question. Please include a reasonable number of figures for each question (no more than about 6) and keep written responses as brief as possible (around 250 words per question). Participants are encouraged to new visual representations rather than relying on traditional or existing approaches.

7 Conclusion and Reflections

(Write this once your findings and visuals are finalized.)
Summarize:

- Key insights discovered
- Effectiveness of the visual workflows
- Challenges faced in interpreting incomplete or noisy data
- Future improvements or extensions (e.g., web dashboards, live geo-visualizations)

7.1 Future Improvements

Future iterations of the platform could integrate machine learning-based anomaly detection, more robust temporal aggregation controls, and user-customizable dashboards. Integrating user feedback loops could also support collaborative visual sensemaking [**?**].