Prototype

# **Prototyping our “🌊Covert Reef🪸” Project**

We worked like a beaver to come up with our prototype to emulate a water tight product like the 🦫 beaver’s dam. Here in this prototype section we will go through the preparatory work, visualisation design, tested r codes, and parameters and outputs. A Shiny prototype was also created subsequent to this. Shiny was used due to the many advantages such as automatic resizing. Click here for the Shiny prototype.

# **0) Visual Thematic Overview**

Start off with wordclouds at the top of the webpage to serve as a precursor to key themes:

* Shows a visually prominent wordcloud immediately at the top of the page;
* Previews key terms from the dataset (as a “hook”) to engage users and provide a thematic overview;
* Shiny compatible

library(shiny)  
library(wordcloud)  
library(RColorBrewer)  
  
# Data and color palette  
word <- c("vessels", "mining", "funding", "music", "reef", "conservation", "turtles",  
 "fuel", "permit", "?", "suspicious", "tourism", "lighting", "meetings",  
 "council", "harbor", "relationships", "communications", "operation", "underwater")  
freq <- c(5, 8, 7, 8, 9, 5, 6, 10, 8, 4, 3, 3, 5, 2, 3, 2, 5, 8, 3, 4)  
word\_data\_wc <- data.frame(word = word, freq = freq)  
  
ocean\_colors\_lp <- c(  
 "#1B1B3A", "#0072B2", "#009E73", "#D55E00", "#CC79A7",  
 "#882255", "#AA4499", "#004D40", "#333333"  
)  
  
# Define UI  
ui <- fluidPage(  
 titlePanel(" "),  
 plotOutput("wordcloudPlot", height = "500px") # Increased height  
)  
  
# Define Server  
server <- function(input, output, session) {  
 output$wordcloudPlot <- renderPlot({  
 par(bg = "#f0f0fb") # Set background inside renderPlot  
  
 wordcloud(  
 words = word\_data\_wc$word,  
 freq = word\_data\_wc$freq,  
 min.freq = 1,  
 max.words = 200,  
 random.order = FALSE,  
 colors = ocean\_colors\_lp,  
 rot.per = 0.20,  
 scale = c(4, 0.8)  
 )  
 })  
}  
  
# Launch App  
shinyApp(ui, server)



# **1) Setup and Preparatory Work**

## **1.1 Loading Packages to prepare for SHINY**

## **Packages supported in R CRAN:**

| **Utility Tools** | **Graphing Tools** |
| --- | --- |
| * [jsonlite](https://cran.r-project.org/web/packages/jsonlite/index.html) - To parse JSON * [tidyverse](https://www.tidyverse.org/) - Data science tools * [ggtext](https://cran.r-project.org/web/packages/ggtext/index.html) - Tools for text formatting * [knitr](https://cran.r-project.org/web/packages/knitr/index.html) - For better table displays * [lubridate](https://cran.r-project.org/web/packages/lubridate/vignettes/lubridate.html) - For processing date and time * [hms](https://cran.r-project.org/web/packages/hms/index.html) - For durations * [scales](https://cran.r-project.org/web/packages/scales/index.html) - For breaks and labels * [tidytext](https://cran.r-project.org/web/packages/tidytext/index.html) - For functions text mining * [tm](https://cran.r-project.org/web/packages/tm/index.html) - For text mining * [SnowBallC](https://cran.r-project.org/web/packages/SnowballC/index.html) - For Porter’s word stemming * [SmartEDA](https://cran.r-project.org/web/packages/SmartEDA/index.html) - EDA with some graphing * [reactable](https://cran.r-project.org/web/packages/reactable/index.html) - For interactive data tables * [highcharter](https://cran.r-project.org/web/packages/highcharter/index.html)- For wrapper function * [conflicted](https://cran.r-project.org/web/packages/conflicted/index.html) - For managing namespace conflicts | * [patchwork](https://cran.r-project.org/web/packages/patchwork/index.html) - For combining ggplot plots * [ggraph](https://ggraph.data-imaginist.com/) - For plotting network data * [tidygraph](https://cran.r-project.org/web/packages/tidygraph/index.html) - For graph manipulations * [igraph](https://cran.r-project.org/web/packages/igraph/index.html) - Contains functions for network analysis * [ggiraph](https://cran.r-project.org/web/packages/ggiraph/index.html) - Interactive plots * [plotly](https://cran.r-project.org/web/packages/plotly/index.html) - Interactive plots * [wordcloud](https://cran.r-project.org/web/packages/wordcloud/index.html)- For frequency representation of words * [ggh4x](https://cran.r-project.org/web/packages/ggh4x/index.html) - For axis, strip, and facet customizations * [visNetwork](https://cran.r-project.org/web/packages/visNetwork/index.html) - For interactive visualisation of networks. * [RColorBrewer](https://cran.r-project.org/web/packages/RColorBrewer/index.html) - Colour schemes for graphics * [circlize](https://cran.r-project.org/web/packages/circlize/index.html) - For circular plots * [ggalluvial](https://cran.r-project.org/web/packages/alluvial/index.html) - For alluvial diagrams * [networkD3](https://cran.r-project.org/web/packages/networkD3/index.html)- For D3-based interactive network graphs * [leaflet](https://cran.r-project.org/web/packages/leaflet/index.html)- For interactive maps |

* We will use the code chunk below uses p\_load() of pacman package to check if packages are installed in the computer. If they are, then they will be launched into R. Though it is not added here in pacman due to conflicts, networkD3, highcharter, leaflet, conflicted will also be added on Shiny.

pacman::p\_load(jsonlite, tidyverse, ggtext,   
 knitr, lubridate, hms, scales,   
 tidytext, tm, SnowballC,   
 patchwork, ggraph,   
 tidygraph, igraph, ggiraph,   
 SmartEDA, plotly, wordcloud,   
 ggh4x, visNetwork, RColorBrewer,   
 circlize, ggalluvial, reactable)

## **🧰 Part of Tidyverse**

* [stringr](https://cran.r-project.org/web/packages/stringr/index.html) - For wrapping text and string manipulation
* [readr](https://cran.r-project.org/web/packages/readr/index.html) - For reading rectangular data (CSV, TSV, etc)
* [dplyr](https://cran.r-project.org/web/packages/dplyr/index.html) - For transforming, filtering, summarising data
* [ggplot2](https://cran.r-project.org/web/packages/ggplot2/index.html) - For building data visualisation

## 🔄**Workarounds:**

These are some workarounds required for Shiny application:

* magick - For graphics and image processing will be replaced by renderImage() + impageOutput()
* We will use renderWordcloud2() (from the wordcloud2 package) and wordcloud2Output() in our UI.

## 🆚**Settle Conflict Issues:**

* conflicts\_prefer(shinydashboard::box)
* conflicts\_prefer(dplyr::filter)
* conflicts\_prefer(dplyr::lag)
* conflicts\_prefer(networkD3::JS)

## **1.2 Loading Data**

In the code chunk below, fromJSON() of jsonlite package was used to import mc3.json file into R and save the output object.

mc3\_data <- fromJSON("data/mc3\_graph.json")  
mc3\_schema <- fromJSON("data/MC3\_schema.json")

It contains graph data, where nodes can be accessed via nodes and edges via links. This dataset had many columns but we filtered the relevant data during wrangling.

## **1.3 Defining common variables**

We will also set some values for consistency throughout all graphs.

## Style and Colours

node\_legend\_colors\_plot <- c(  
 "Person" = "#88CCEE",  
 "Vessel" = "#D55E00",  
 "Organization" = "#117733",  
 "Location" = "#AA4499",  
 "Group"= "#CC79A7",  
 "Event" = "#DDCC77",  
 "Relationship" = "#AF8DC3",  
 "Nadia Conti" = "red"  
)  
  
node\_legend\_shapes\_plot <- c(  
 "Person" = "dot",  
 "Vessel" = "triangle",  
 "Organization" = "square",  
 "Location" = "diamond",  
 "Group" = "circle plus",  
 "Event" = "star",  
 "Relationship" = "square x",  
 "Nadia Conti" = "star"  
)  
  
STYLES <- list(  
 node\_label\_dark = "black",  
 font\_family = "Roboto Condensed"  
)

## **1.4 Inspecting knowledge graph structure**

In the code chunk below glimpse() is used to reveal the structure of mc3\_data knowledge graph.

## The Code

glimpse(mc3\_data)

## The Result

List of 5  
 $ directed : logi TRUE  
 $ multigraph: logi FALSE  
 $ graph :List of 4  
 ..$ mode : chr "static"  
 ..$ edge\_default: Named list()  
 ..$ node\_default: Named list()  
 ..$ name : chr "VAST\_MC3\_Knowledge\_Graph"  
 $ nodes :'data.frame': 1159 obs. of 31 variables:  
 ..$ type : chr [1:1159] "Entity" "Entity" "Entity" "Entity" ...  
 ..$ label : chr [1:1159] "Sam" "Kelly" "Nadia Conti" "Elise" ...  
 ..$ name : chr [1:1159] "Sam" "Kelly" "Nadia Conti" "Elise" ...  
 ..$ sub\_type : chr [1:1159] "Person" "Person" "Person" "Person" ...  
 ..$ id : chr [1:1159] "Sam" "Kelly" "Nadia Conti" "Elise" ...  
 ..$ timestamp : chr [1:1159] NA NA NA NA ...  
 ..$ monitoring\_type : chr [1:1159] NA NA NA NA ...  
 ..$ findings : chr [1:1159] NA NA NA NA ...  
 ..$ content : chr [1:1159] NA NA NA NA ...  
 ..$ assessment\_type : chr [1:1159] NA NA NA NA ...  
 ..$ results : chr [1:1159] NA NA NA NA ...  
 ..$ movement\_type : chr [1:1159] NA NA NA NA ...  
 ..$ destination : chr [1:1159] NA NA NA NA ...  
 ..$ enforcement\_type : chr [1:1159] NA NA NA NA ...  
 ..$ outcome : chr [1:1159] NA NA NA NA ...  
 ..$ activity\_type : chr [1:1159] NA NA NA NA ...  
 ..$ participants : int [1:1159] NA NA NA NA NA NA NA NA NA NA ...  
 ..$ thing\_collected :'data.frame': 1159 obs. of 2 variables:  
 .. ..$ type: chr [1:1159] NA NA NA NA ...  
 .. ..$ name: chr [1:1159] NA NA NA NA ...  
 ..$ reference : chr [1:1159] NA NA NA NA ...  
 ..$ date : chr [1:1159] NA NA NA NA ...  
 ..$ time : chr [1:1159] NA NA NA NA ...  
 ..$ friendship\_type : chr [1:1159] NA NA NA NA ...  
 ..$ permission\_type : chr [1:1159] NA NA NA NA ...  
 ..$ start\_date : chr [1:1159] NA NA NA NA ...  
 ..$ end\_date : chr [1:1159] NA NA NA NA ...  
 ..$ report\_type : chr [1:1159] NA NA NA NA ...  
 ..$ submission\_date : chr [1:1159] NA NA NA NA ...  
 ..$ jurisdiction\_type: chr [1:1159] NA NA NA NA ...  
 ..$ authority\_level : chr [1:1159] NA NA NA NA ...  
 ..$ coordination\_type: chr [1:1159] NA NA NA NA ...  
 ..$ operational\_role : chr [1:1159] NA NA NA NA ...  
 $ edges :'data.frame': 3226 obs. of 5 variables:  
 ..$ id : chr [1:3226] "2" "3" "5" "3013" ...  
 ..$ is\_inferred: logi [1:3226] TRUE FALSE TRUE TRUE TRUE TRUE ...  
 ..$ source : chr [1:3226] "Sam" "Sam" "Sam" "Sam" ...  
 ..$ target : chr [1:3226] "Relationship\_Suspicious\_217" "Event\_Communication\_370" "Event\_Assessment\_600" "Relationship\_Colleagues\_430" ...  
 ..$ type : chr [1:3226] NA "sent" NA NA ...

## **1.5 Extracting the edges and nodes tables**

Next, as\_tibble() of **tibble** package package is used to extract the nodes and links tibble data frames from *mc3* tibble dataframe into two separate tibble dataframes called *mc3\_nodes\_raw* and *mc3\_edges\_raw* respectively.

## The Code

mc3\_nodes\_raw <- as\_tibble(mc3\_data$nodes)  
mc3\_edges\_raw <- as\_tibble(mc3\_data$edges)

We also looked into the nodes and edges structure.

## Nodes structure

ExpData(data=mc3\_nodes\_raw,type=2)

Index Variable\_Name Variable\_Type Sample\_n Missing\_Count Per\_of\_Missing  
1 1 type character 1159 0 0.000  
2 2 label character 1159 0 0.000  
3 3 name character 72 1087 0.938  
4 4 sub\_type character 1159 0 0.000  
5 5 id character 1159 0 0.000  
6 6 timestamp character 770 389 0.336  
7 7 monitoring\_type character 70 1089 0.940  
8 8 findings character 70 1089 0.940  
9 9 content character 584 575 0.496  
10 10 assessment\_type character 33 1126 0.972  
11 11 results character 32 1127 0.972  
12 12 movement\_type character 33 1126 0.972  
13 13 destination character 41 1118 0.965  
14 14 enforcement\_type character 21 1138 0.982  
15 15 outcome character 19 1140 0.984  
16 16 activity\_type character 4 1155 0.997  
17 17 participants integer 1 1158 0.999  
18 18 thing\_collected data.frame -2314 2316 1158.000  
19 19 reference character 1 1158 0.999  
20 20 date character 5 1154 0.996  
21 21 time character 3 1156 0.997  
22 22 friendship\_type character 2 1157 0.998  
23 23 permission\_type character 55 1104 0.953  
24 24 start\_date character 90 1069 0.922  
25 25 end\_date character 6 1153 0.995  
26 26 report\_type character 19 1140 0.984  
27 27 submission\_date character 22 1137 0.981  
28 28 jurisdiction\_type character 13 1146 0.989  
29 29 authority\_level character 7 1152 0.994  
30 30 coordination\_type character 73 1086 0.937  
31 31 operational\_role character 38 1121 0.967  
 No\_of\_distinct\_values  
1 3  
2 92  
3 73  
4 25  
5 1159  
6 612  
7 6  
8 70  
9 581  
10 4  
11 33  
12 4  
13 26  
14 3  
15 20  
16 3  
17 1  
18 2  
19 2  
20 6  
21 4  
22 3  
23 4  
24 69  
25 6  
26 4  
27 23  
28 13  
29 3  
30 53  
31 35

## Edges structure

ExpData(data=mc3\_edges\_raw,type=2)

Index Variable\_Name Variable\_Type Sample\_n Missing\_Count Per\_of\_Missing  
1 1 id character 3169 57 0.018  
2 2 is\_inferred logical 3226 0 0.000  
3 3 source character 3226 0 0.000  
4 4 target character 3226 0 0.000  
5 5 type character 2204 1022 0.317  
 No\_of\_distinct\_values  
1 3170  
2 2  
3 1052  
4 1156  
5 4

# **2) Initial EDA for Nodes and Edges**

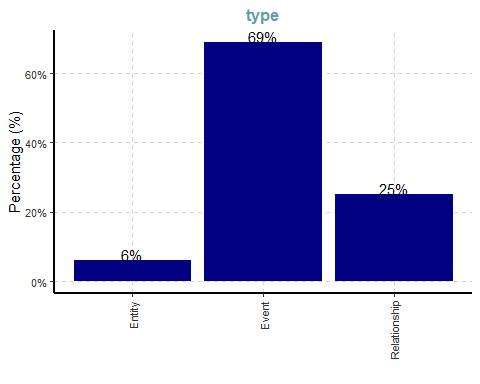
## **2.1 Nodes via Shiny**

## Nodes

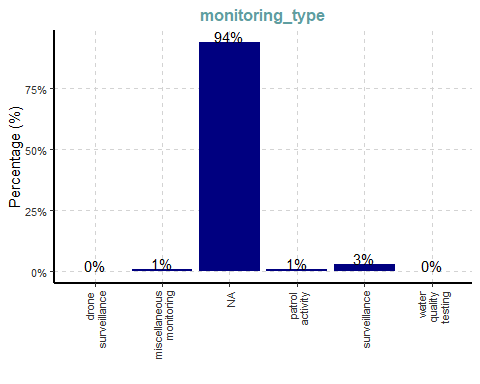
In the code chunk below, ExpCatViz() of SmartEDA package is used to reveal the frequency distribution of all categorical fields in mc3\_nodes tibble dataframe.

library(shiny)  
library(ggplot2)  
  
# Run ExpCatViz once at the top to avoid recomputing  
ExpCatViz(data=mc3\_nodes\_raw,  
 col="navyblue")

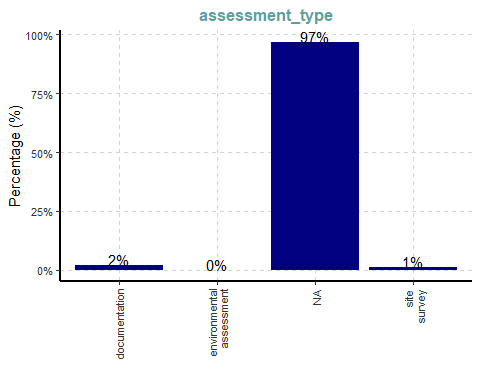
[[1]]



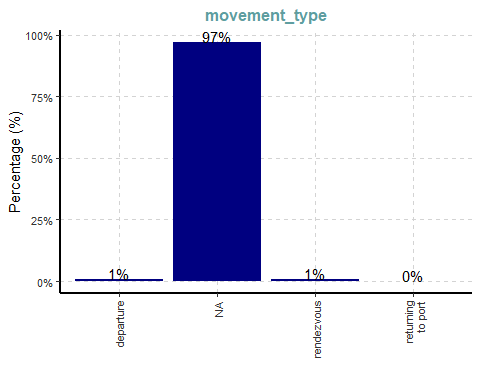
[[2]]



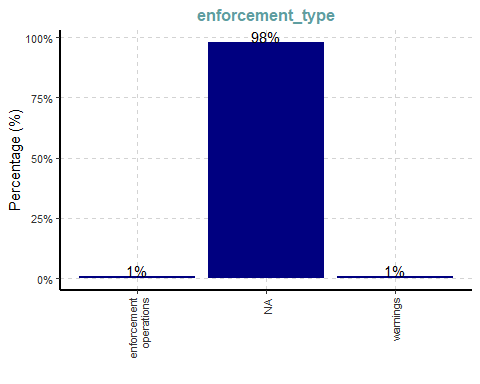
[[3]]



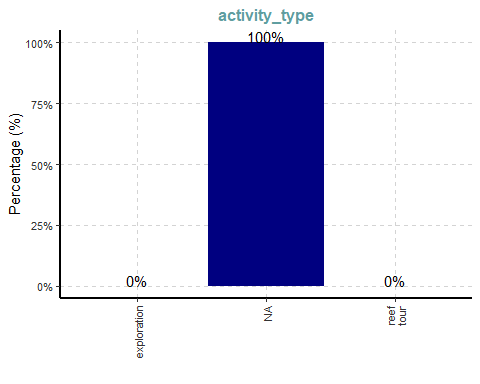
[[4]]



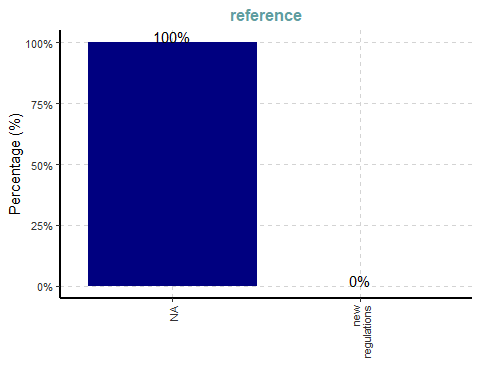
[[5]]



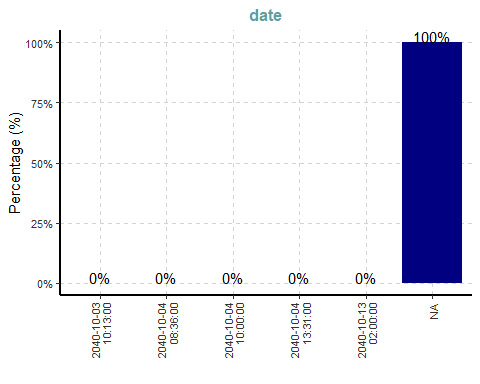
[[6]]



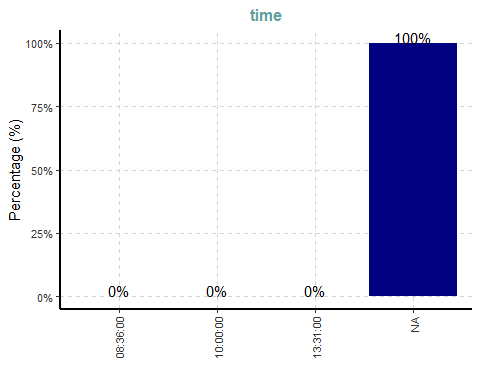
[[7]]



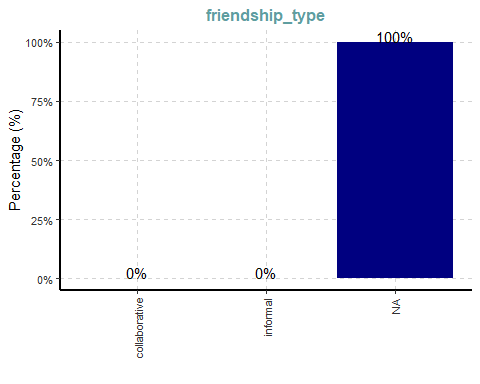
[[8]]



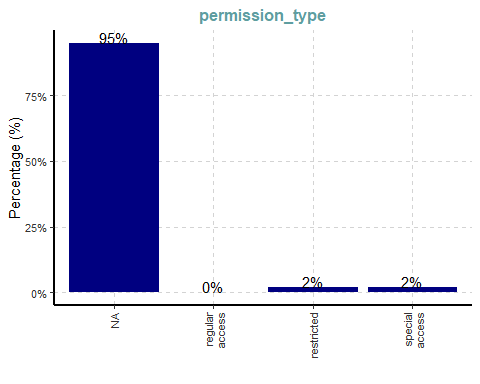
[[9]]



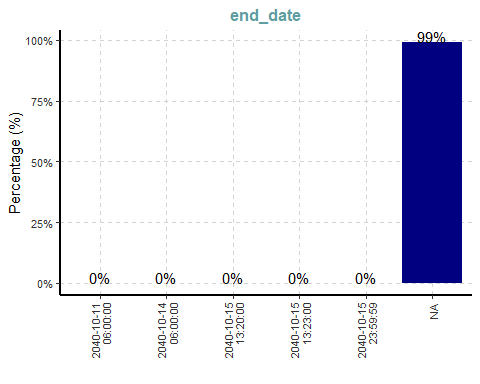
[[10]]



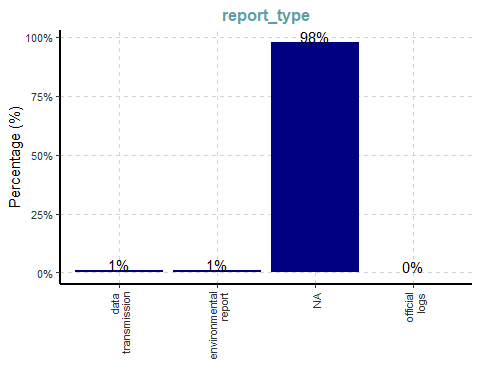
[[11]]



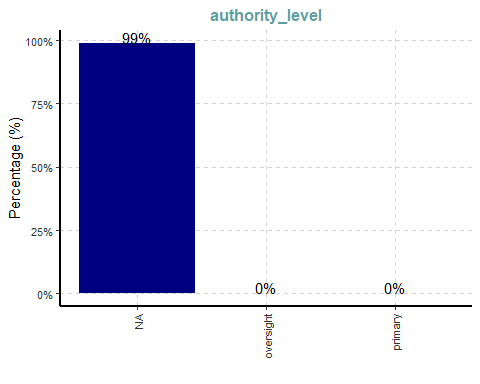
[[12]]



[[13]]



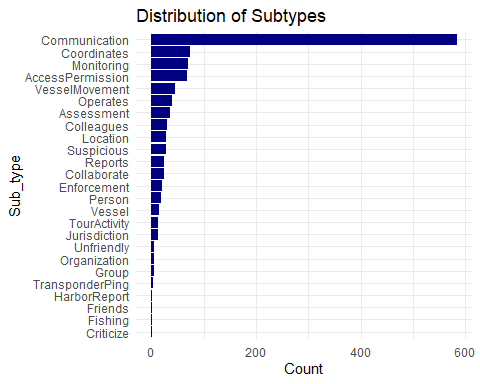
[[14]]



## Drilling into Node sub\_type

Code chunk below uses ggplot2 functions to reveal the frequency distribution of *sub\_type* field of *mc3\_nodes\_raw*.

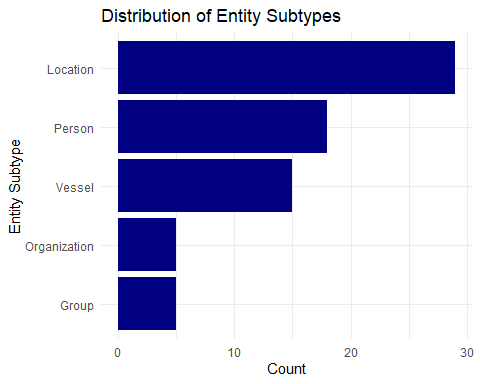
# Step 1: Count and reorder  
mc3\_nodes\_ordered <- mc3\_nodes\_raw %>%  
 count(sub\_type) %>%  
 arrange((n)) %>%  
 mutate(sub\_type = factor(sub\_type, levels = sub\_type))  
  
# Step 2: Plot with navy bars, sorted, and horizontal  
ggplot(mc3\_nodes\_ordered, aes(x = sub\_type, y = n)) +  
 geom\_col(fill = "navy") +  
 coord\_flip() +  
 labs(x = "Sub\_type", y = "Count",  
 title = "Distribution of Subtypes") +  
 theme\_minimal()



## Entity subtypes

In the code chunk below, the Entity subtypes are filtered.

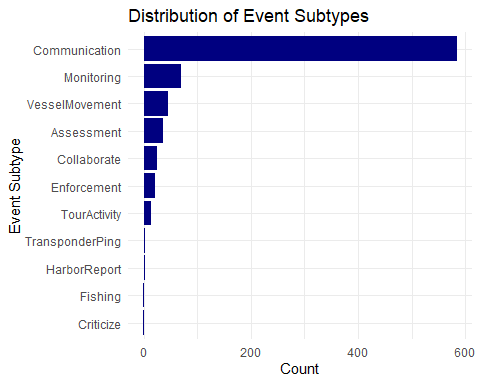
# Step 1: Filter for type == "Entity", count sub\_type, sort   
relationship\_subtypes <- mc3\_nodes\_raw %>%  
 filter(type == "Entity") %>%  
 count(sub\_type) %>%  
 arrange(n) %>%  
 mutate(sub\_type = factor(sub\_type, levels = sub\_type))   
  
# Step 2: Plot  
ggplot(relationship\_subtypes, aes(x = sub\_type, y = n)) +  
 geom\_col(fill = "navy") +  
 coord\_flip() +  
 labs(  
 x = "Entity Subtype",  
 y = "Count",  
 title = "Distribution of Entity Subtypes"  
 ) +  
 theme\_minimal()



## Event subtypes

In the code chunk below, the Event subtypes are filtered.

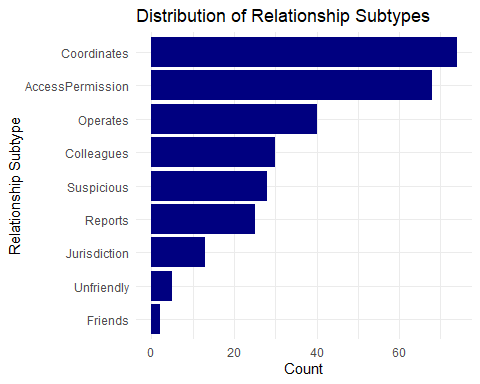
# Step 1: Filter for type == "Event", count sub\_type, sort   
relationship\_subtypes <- mc3\_nodes\_raw %>%  
 filter(type == "Event") %>%  
 count(sub\_type) %>%  
 arrange(n) %>%  
 mutate(sub\_type = factor(sub\_type, levels = sub\_type))   
  
# Step 2: Plot  
ggplot(relationship\_subtypes, aes(x = sub\_type, y = n)) +  
 geom\_col(fill = "navy") +  
 coord\_flip() +  
 labs(  
 x = "Event Subtype",  
 y = "Count",  
 title = "Distribution of Event Subtypes"  
 ) +  
 theme\_minimal()



## Relationship subtypes

In the code chunk below, the relationship subtypes are filtered.

# Step 1: Filter for type == "Relationship", count sub\_type, sort   
relationship\_subtypes <- mc3\_nodes\_raw %>%  
 filter(type == "Relationship") %>%  
 count(sub\_type) %>%  
 arrange(n) %>%  
 mutate(sub\_type = factor(sub\_type, levels = sub\_type))  
  
# Step 2: Plot  
ggplot(relationship\_subtypes, aes(x = sub\_type, y = n)) +  
 geom\_col(fill = "navy") +  
 coord\_flip() +  
 labs(  
 x = "Relationship Subtype",  
 y = "Count",  
 title = "Distribution of Relationship Subtypes"  
 ) +  
 theme\_minimal()



## 2.1.1 Findings from EDA

## Nodes

We will use the EDA findings to determine data to focus on or eliminate. From the bar charts and the original data on mc3\_nodes\_raw, it was observed that:

* Nodes were one of three types (Entity, Event, Relationship), where each of these types have their sub\_types. Majority were of event type, followed by relationship, and entity.
  + There were 25 subtypes. Communications made up the bulk of the sub\_type for Events. Coordinates made up the bulk of the sub\_type for Relationship. The additional node sub\_types not mentioned in the VAST 2025 MC3 Data Description under Node Attributes were: fishing, communication and coordinates.

## Event Types

* Observations of EDA from Event types:
  + Findings field were filled when there were monitoring\_type.
  + Content refers to radio communication content.
  + Results field were filled when there were assessment\_type performed.
  + When there is an enforcement\_type of enforcement operations or warnings, there might be an outcome at times.
  + When there is a movement\_type, there might be a place of destination at times.

## Relationship Types

* Observations of EDA from Relationship types:
  + When the subtype was coordinate, there were data in the field named coordination\_types.
  + When the subtype was operate, there were data in the field named operational\_roles.
  + When there is a jurisdiction\_type, there might be an authority\_level.
  + There are only restricted or special access data within permission\_types.
  + When there is a report\_type of data transmission or environmental report, there might be a submission\_date.

## Entity Types

* Observations of EDA from Entity types:
  + The 5 id under Group sub-types were not very useful information.

## Course of Action

* Elimination and directed focus:
  + Relative to the entire dataset, there were little assessment\_type (3%), movement\_type (2%), enforcement\_type (2%), permission\_type (4%), report\_type (2%), authority\_level (1%). We will direct our focus on other areas instead of these.
  + There were no to little useful data in the fields named: activity\_type, references, dates, time, and friendship\_type. These were not utilised.
  + We directed our focus on Event\_Communication, Event\_Monitoring, and Event\_VesselMovement.

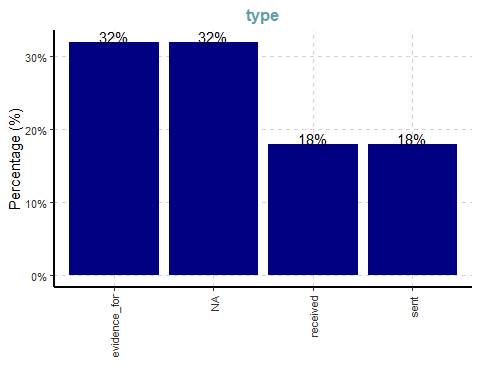
## **2.2 Edges**

The code chunk below used ExpCATViz() of SmartEDA package to reveal the frequency distribution of all categorical fields in mc3\_edges\_raw tibble dataframe.

## Frequency Distribution of Categorical Fields

ExpCatViz(data=mc3\_edges\_raw,  
 col="navyblue")

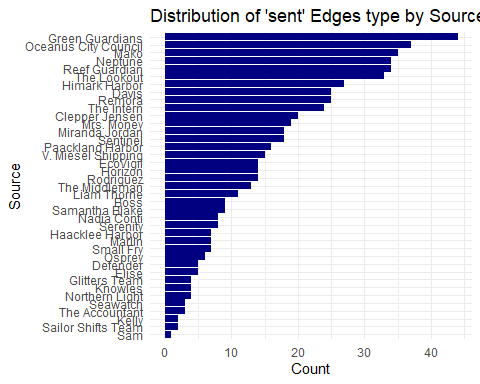
[[1]]



Entities are connected by edges to other Entities via an Event or Relationship node. The one exception to this is the Communication Event subtype, which is additionally linked to either an Event or Relationship node. The type field denotes the connector or edge type for the Entities, Event, and Relationship nodes. The edges are one of these: received, evidence\_for, sent, NA.

## Filter by type == sent

# Step 1: Filter for type == "sent"  
filtered\_edges <- mc3\_edges\_raw %>%  
 filter(type == "sent") %>%  
 count(source) %>%  
 arrange(desc(n)) %>%  
 mutate(source = factor(source, levels = rev(unique(source)))) # descending   
  
# Step 2: Plot  
ggplot(filtered\_edges, aes(x = source, y = n)) +  
 geom\_col(fill = "navy") +  
 coord\_flip() +  
 labs(  
 title = "Distribution of 'sent' Edges type by Source",  
 x = "Source",  
 y = "Count"  
 ) +  
 theme\_minimal()



**What we understood from the information provided by Vast Challenge on Directional Edges:**

* For relationship as colleagues node or friends node, the node will have arrows/ edges pointing towards the relationship node.
* For other relationships and events, the direction would be following the source and target.

# **3) Data Preparation**

## **3.1 Data Cleaning and Wrangling**

## Cleaning and wrangling nodes

* convert values in id field into character data type,
* exclude records with id value are na,
* exclude records with similar id values,
* exclude thing\_collected , time , date, friendship\_type field, and
* save the cleaned tibble dataframe into a new tibble datatable called mc3\_nodes\_cleaned.

mc3\_nodes\_cleaned <- mc3\_nodes\_raw %>%  
 mutate(id = as.character(id)) %>%  
 filter(!is.na(id)) %>%  
 distinct(id, .keep\_all = TRUE) %>%  
 select(-thing\_collected, -time, -date, -friendship\_type)

## Unique Node Count

# A tibble: 27 × 2  
 column unique\_count  
 <chr> <int>  
 1 id 1159  
 2 timestamp 612  
 3 content 581  
 4 label 92  
 5 name 73  
 6 findings 70  
 7 start\_date 69  
 8 coordination\_type 53  
 9 operational\_role 35  
10 results 33  
# ℹ 17 more rows

## Cleaning and wrangling edges

* renamed source and target fields to from\_id and to\_id respectively,
* converted values in from\_id and to\_id fields to character data type,
* excluded values in from\_id and to\_id which not found in the id field of mc3\_nodes\_cleaned,
* excluded records whereby from\_id and/or to\_id values are missing, and
* saved the cleaned tibble dataframe and called it mc3\_edges\_cleaned.

mc3\_edges\_cleaned <- mc3\_edges\_raw %>%  
 rename(from\_id = source,  
 to\_id = target) %>%  
 mutate(across(c(from\_id, to\_id), as.character)) %>%  
 # Parse to\_id to get supertype and sub\_type for target nodes (e.g., Event\_Communication)  
 separate(to\_id, into = c("to\_id\_supertype", "to\_id\_sub\_type", "to\_id\_num"),  
 sep = "\_", remove = FALSE, fill = "right", extra = "merge") %>%  
 # Filter to ensure from\_id and to\_id exist in mc3\_nodes\_cleaned (prevent orphaned edges)  
 filter(from\_id %in% mc3\_nodes\_cleaned$id,  
 to\_id %in% mc3\_nodes\_cleaned$id) %>%  
 filter(!is.na(from\_id), !is.na(to\_id))  
  
print("Columns in mc3\_edges\_cleaned after initial cleaning:")

[1] "Columns in mc3\_edges\_cleaned after initial cleaning:"

print(colnames(mc3\_edges\_cleaned))

[1] "id" "is\_inferred" "from\_id" "to\_id"   
[5] "to\_id\_supertype" "to\_id\_sub\_type" "to\_id\_num" "type"

print("Head of mc3\_edges\_cleaned after initial cleaning:")

[1] "Head of mc3\_edges\_cleaned after initial cleaning:"

print(head(mc3\_edges\_cleaned))

# A tibble: 6 × 8  
 id is\_inferred from\_id to\_id to\_id\_supertype to\_id\_sub\_type to\_id\_num type   
 <chr> <lgl> <chr> <chr> <chr> <chr> <chr> <chr>  
1 2 TRUE Sam Rela… Relationship Suspicious 217 <NA>   
2 3 FALSE Sam Even… Event Communication 370 sent   
3 5 TRUE Sam Even… Event Assessment 600 <NA>   
4 3013 TRUE Sam Rela… Relationship Colleagues 430 <NA>   
5 <NA> TRUE Sam Rela… Relationship Friends 272 <NA>   
6 <NA> TRUE Sam Rela… Relationship Colleagues 215 <NA>

## Unique Edges Count

# Find the number of unique types in each column  
unique\_counts <- mc3\_edges\_cleaned %>%  
 summarise\_all(n\_distinct) %>%  
 pivot\_longer(cols = everything(), names\_to = "column", values\_to = "unique\_count")  
  
# Print the unique counts for each column  
print(unique\_counts)

# A tibble: 8 × 2  
 column unique\_count  
 <chr> <int>  
1 id 3170  
2 is\_inferred 2  
3 from\_id 1052  
4 to\_id 1156  
5 to\_id\_supertype 71  
6 to\_id\_sub\_type 21  
7 to\_id\_num 860  
8 type 4

## Other preparatory work

Next, code chunk below will be used to create mapping of character id in mc3\_nodes\_cleaned to row index

node\_index\_lookup <- mc3\_nodes\_cleaned %>%  
 mutate(.row\_id = row\_number()) %>%  
 select(id, .row\_id)

Next, the code chunk below was used to join and convert from\_id and to\_id to integer indices. At the same time we also dropped rows with unmatched nodes.

mc3\_edges\_indexed <- mc3\_edges\_cleaned %>%  
 left\_join(node\_index\_lookup, by = c("from\_id" = "id")) %>%  
 rename(from = .row\_id) %>%  
 left\_join(node\_index\_lookup, by = c("to\_id" = "id")) %>%  
 rename(to = .row\_id) %>%  
 # Filter out edges where either source or target node was not found  
 filter(!is.na(from) & !is.na(to)) %>%  
 # Select all columns to carry forward to mc3\_edges\_final  
 select(from, to, id, is\_inferred, type, # Original edge attributes  
 from\_id, to\_id, to\_id\_supertype, to\_id\_sub\_type, to\_id\_num # Original IDs and parsed target type  
 )

Next the code chunk below was used to subset nodes to only those referenced by edges.

used\_node\_indices <- sort(unique(c(mc3\_edges\_indexed$from, mc3\_edges\_indexed$to)))  
mc3\_nodes\_final <- mc3\_nodes\_cleaned %>%  
 slice(used\_node\_indices) %>%  
 mutate(new\_index = row\_number())

We then used the code chunk below to rebuild lookup from old index to new index.

old\_to\_new\_index <- tibble(  
 old\_index = used\_node\_indices,  
 new\_index = seq\_along(used\_node\_indices)  
)

Lastly, the code chunk below was used to update edge indices to match new node table.

mc3\_edges\_final <- mc3\_edges\_indexed %>%  
 left\_join(old\_to\_new\_index, by = c("from" = "old\_index")) %>%  
 rename(from\_new = new\_index) %>%  
 left\_join(old\_to\_new\_index, by = c("to" = "old\_index")) %>%  
 rename(to\_new = new\_index) %>%  
 # Explicitly select all columns that are needed downstream  
 select(from = from\_new, to = to\_new,  
 id, is\_inferred, type,  
 from\_id, to\_id, to\_id\_supertype, to\_id\_sub\_type, to\_id\_num)

## **3.2 Building the tidygraph object**

## Build the object-tbl\_graph

mc3\_graph <- tbl\_graph(  
 nodes = mc3\_nodes\_final,  
 edges = mc3\_edges\_final,  
 directed = TRUE  
)

## Examining the object

str(mc3\_graph)

Classes 'tbl\_graph', 'igraph' hidden list of 10  
 $ : num 1159  
 $ : logi TRUE  
 $ : num [1:3226] 0 0 0 0 0 0 0 1 1 1 ...  
 $ : num [1:3226] 1137 356 746 894 875 ...  
 $ : NULL  
 $ : NULL  
 $ : NULL  
 $ : NULL  
 $ :List of 4  
 ..$ : num [1:3] 1 0 1  
 ..$ : Named list()  
 ..$ :List of 28  
 .. ..$ type : chr [1:1159] "Entity" "Entity" "Entity" "Entity" ...  
 .. ..$ label : chr [1:1159] "Sam" "Kelly" "Nadia Conti" "Elise" ...  
 .. ..$ name : chr [1:1159] "Sam" "Kelly" "Nadia Conti" "Elise" ...  
 .. ..$ sub\_type : chr [1:1159] "Person" "Person" "Person" "Person" ...  
 .. ..$ id : chr [1:1159] "Sam" "Kelly" "Nadia Conti" "Elise" ...  
 .. ..$ timestamp : chr [1:1159] NA NA NA NA ...  
 .. ..$ monitoring\_type : chr [1:1159] NA NA NA NA ...  
 .. ..$ findings : chr [1:1159] NA NA NA NA ...  
 .. ..$ content : chr [1:1159] NA NA NA NA ...  
 .. ..$ assessment\_type : chr [1:1159] NA NA NA NA ...  
 .. ..$ results : chr [1:1159] NA NA NA NA ...  
 .. ..$ movement\_type : chr [1:1159] NA NA NA NA ...  
 .. ..$ destination : chr [1:1159] NA NA NA NA ...  
 .. ..$ enforcement\_type : chr [1:1159] NA NA NA NA ...  
 .. ..$ outcome : chr [1:1159] NA NA NA NA ...  
 .. ..$ activity\_type : chr [1:1159] NA NA NA NA ...  
 .. ..$ participants : int [1:1159] NA NA NA NA NA NA NA NA NA NA ...  
 .. ..$ reference : chr [1:1159] NA NA NA NA ...  
 .. ..$ permission\_type : chr [1:1159] NA NA NA NA ...  
 .. ..$ start\_date : chr [1:1159] NA NA NA NA ...  
 .. ..$ end\_date : chr [1:1159] NA NA NA NA ...  
 .. ..$ report\_type : chr [1:1159] NA NA NA NA ...  
 .. ..$ submission\_date : chr [1:1159] NA NA NA NA ...  
 .. ..$ jurisdiction\_type: chr [1:1159] NA NA NA NA ...  
 .. ..$ authority\_level : chr [1:1159] NA NA NA NA ...  
 .. ..$ coordination\_type: chr [1:1159] NA NA NA NA ...  
 .. ..$ operational\_role : chr [1:1159] NA NA NA NA ...  
 .. ..$ new\_index : int [1:1159] 1 2 3 4 5 6 7 8 9 10 ...  
 ..$ :List of 8  
 .. ..$ id : chr [1:3226] "2" "3" "5" "3013" ...  
 .. ..$ is\_inferred : logi [1:3226] TRUE FALSE TRUE TRUE TRUE TRUE ...  
 .. ..$ type : chr [1:3226] NA "sent" NA NA ...  
 .. ..$ from\_id : chr [1:3226] "Sam" "Sam" "Sam" "Sam" ...  
 .. ..$ to\_id : chr [1:3226] "Relationship\_Suspicious\_217" "Event\_Communication\_370" "Event\_Assessment\_600" "Relationship\_Colleagues\_430" ...  
 .. ..$ to\_id\_supertype: chr [1:3226] "Relationship" "Event" "Event" "Relationship" ...  
 .. ..$ to\_id\_sub\_type : chr [1:3226] "Suspicious" "Communication" "Assessment" "Colleagues" ...  
 .. ..$ to\_id\_num : chr [1:3226] "217" "370" "600" "430" ...  
 $ :<environment: 0x000001c986f51008>   
 - attr(\*, "active")= chr "nodes"

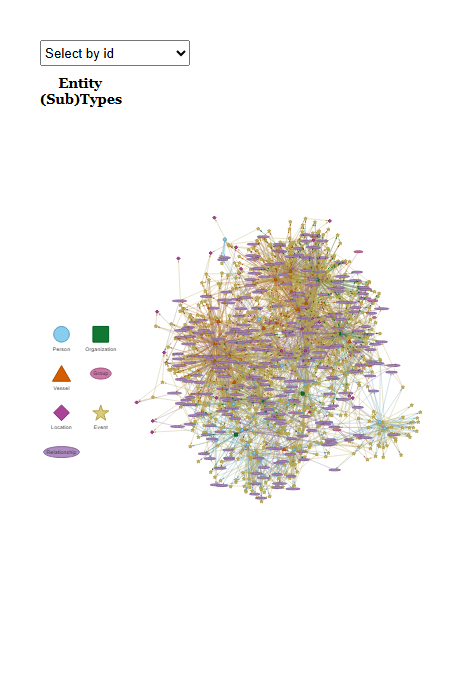
# **4) Knowledge Graphs**

## **VisNetwork**

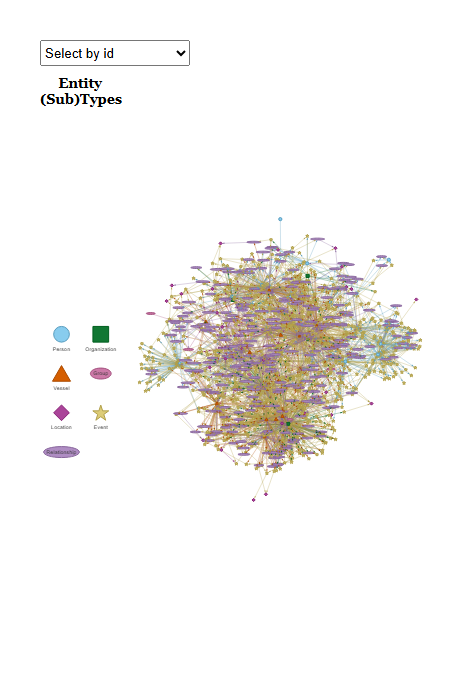
VisNetwork provides the user to understand relationships through interactivity. For instance:

* The individual nodes can be selected from the drop-down menu to view its connected nodes and edges.
* The hover tooltip provides additional details from fields such as content, coordination\_type, findings, destination, operational\_role, results, and jurisdiction\_type based on the related id information from mc3\_nodes\_final.

## The Graph- VisNetwork



# ---- 1. Define styles and legends ----  
  
event\_subtypes <- c(  
 "Communication", "Monitoring", "VesselMovement", "Assessment",  
 "Collaborate", "Endorsement", "TourActivity", "TransponderPing",  
 "Harbor Report", "Fishing", "Criticize"  
)  
  
relationship\_subtypes <- c(  
 "Coordinates", "AccessPermission", "Operates", "Colleagues",  
 "Suspicious", "Reports", "Jurisdiction", "Unfriendly", "Friends"  
)  
  
node\_legend\_colors\_plot <- c(  
 "Person" = "#88CCEE",  
 "Vessel" = "#D55E00",  
 "Organization" = "#117733",  
 "Location" = "#AA4499",  
 "Group"= "#CC79A7",  
 "Event" = "#DDCC77", # type level  
 "Relationship" = "#AF8DC3" # type level  
)  
  
node\_legend\_shapes\_plot <- c(  
 "Person" = "dot",  
 "Vessel" = "triangle",  
 "Organization" = "square",  
 "Location" = "diamond",  
 "Group" = "circle plus",  
 "Event" = "star", # type level  
 "Relationship" = "square x" # type level  
)  
  
STYLES <- list(  
 node\_label\_dark = "black",  
 font\_family = "Roboto Condensed"  
)  
  
# ---- 2. Prepare nodes ----  
nodes <- mc3\_nodes\_final %>%  
 mutate(  
 label = ifelse(is.na(name), id, name),  
   
 # These parts are for pulling the related data from other fields  
 tooltip\_extra = case\_when(  
 type == "Event" & sub\_type == "Communication" ~ content,  
 type == "Event" & sub\_type == "Monitoring" ~ findings,  
 type == "Event" & sub\_type == "VesselMovement" ~ destination,  
 type == "Event" & sub\_type == "Assessment" ~ results,  
 type == "Relationship" & sub\_type == "Coordinates" ~ coordination\_type,  
 type == "Relationship" & sub\_type == "Operates" ~ operational\_role,  
 type == "Relationship" & sub\_type == "Jurisdiction" ~ jurisdiction\_type,  
 TRUE ~ NA\_character\_  
 ),  
   
 title = paste0(  
 "<b>", label, "</b><br>",  
 "Type: ", type, "<br>",  
 "Sub-type: ", sub\_type, "<br>",  
 ifelse(!is.na(tooltip\_extra), paste0("<br><b>Details:</b> ", tooltip\_extra), "")  
 ),  
   
 # Fallback logic: if sub\_type is NA or not in styling list, use type instead  
 group = ifelse(sub\_type %in% names(node\_legend\_colors\_plot), sub\_type, type)  
 ) %>%  
 select(id, label, group, title) %>%  
 distinct()  
  
# ---- 3. Prepare directed edges (type == "sent") ----  
  
edges <- mc3\_edges\_final %>%  
 filter(from\_id %in% nodes$id & to\_id %in% nodes$id) %>%  
 select(from = from\_id, to = to\_id)  
  
# ---- 4. Build visNetwork ----  
  
net <- visNetwork(nodes, edges, width = "100%", height = "600px") %>%  
 visEdges(arrows = list(to = list(enabled = TRUE, scaleFactor = 1.5))) %>%  
 visOptions(highlightNearest = TRUE, nodesIdSelection = TRUE) %>%  
 visIgraphLayout(layout = "layout\_with\_fr") %>%  
 visNodes(font = list(  
 size = 14,  
 color = STYLES$node\_label\_dark,  
 face = STYLES$font\_family,  
 vadjust = -15  
 ))  
  
# ---- 5. Apply shape and color per group ----  
  
for (group\_name in names(node\_legend\_colors\_plot)) {  
 net <- net %>% visGroups(  
 groupname = group\_name,  
 color = node\_legend\_colors\_plot[[group\_name]],  
 shape = node\_legend\_shapes\_plot[[group\_name]]  
 )  
}  
# ---- 6. Add legend ----  
  
used\_groups <- unique(nodes$group)  
  
legend\_df <- tibble::tibble(  
 label = used\_groups,  
 shape = node\_legend\_shapes\_plot[used\_groups],  
 color = node\_legend\_colors\_plot[used\_groups]  
) %>%  
 distinct(label, .keep\_all = TRUE) # remove duplicates just in case  
  
net <- net %>% visLegend(  
 addNodes = legend\_df,  
 ncol = 2, # number of columns  
 position = "left",   
 main = "Entity (Sub)Types", # title  
 useGroups = FALSE # show custom legend entries  
)  
# ---- 7. Render ----  
net



# **5) Further Checking and Cleaning Data**

## 5.1 Checking Nodes

Check if mapping is correct for type and sub\_type.

mc3\_nodes\_cleaned %>%  
 group\_by(type, sub\_type) %>%  
 summarize(count = n()) %>%  
 arrange(-count) %>%  
 kable()

| type | sub\_type | count |
| --- | --- | --- |
| Event | Communication | 584 |
| Relationship | Coordinates | 74 |
| Event | Monitoring | 70 |
| Relationship | AccessPermission | 68 |
| Event | VesselMovement | 46 |
| Relationship | Operates | 40 |
| Event | Assessment | 36 |
| Relationship | Colleagues | 30 |
| Entity | Location | 29 |
| Relationship | Suspicious | 28 |
| Event | Collaborate | 25 |
| Relationship | Reports | 25 |
| Event | Enforcement | 21 |
| Entity | Person | 18 |
| Entity | Vessel | 15 |
| Event | TourActivity | 13 |
| Relationship | Jurisdiction | 13 |
| Entity | Group | 5 |
| Entity | Organization | 5 |
| Relationship | Unfriendly | 5 |
| Event | TransponderPing | 3 |
| Event | HarborReport | 2 |
| Relationship | Friends | 2 |
| Event | Criticize | 1 |
| Event | Fishing | 1 |

## 5.2.1 Cleaning Edges

# Split the 'from\_id' column  
mc3\_edges\_cleaned <- mc3\_edges\_cleaned %>%  
 separate(from\_id, into = c("from\_id\_supertype", "from\_id\_sub\_type", "from\_id\_id"), sep = "\_", remove = FALSE, extra = "drop")  
  
# Split the 'target' column into   
mc3\_edges\_cleaned <- mc3\_edges\_cleaned %>%  
 separate(to\_id, into = c("to\_id\_supertype", "to\_id\_sub\_type","to\_id\_id"), sep = "\_", remove = FALSE, extra = "drop")  
  
# Find the number of unique types in each column  
unique\_counts <- mc3\_edges\_cleaned %>%  
 summarise\_all(n\_distinct) %>%  
 pivot\_longer(cols = everything(), names\_to = "column", values\_to = "unique\_count")  
  
# Print the unique counts for each column  
print(unique\_counts)

# A tibble: 12 × 2  
 column unique\_count  
 <chr> <int>  
 1 id 3170  
 2 is\_inferred 2  
 3 from\_id 1052  
 4 from\_id\_supertype 48  
 5 from\_id\_sub\_type 20  
 6 from\_id\_id 822  
 7 to\_id 1156  
 8 to\_id\_supertype 71  
 9 to\_id\_sub\_type 21  
10 to\_id\_id 860  
11 to\_id\_num 860  
12 type 4

## 5.2.2 Checking Edges

# Check the mapping  
mc3\_edges\_cleaned %>%  
 group\_by(from\_id\_supertype, from\_id\_sub\_type) %>%  
 summarize(count = n()) %>%  
 arrange(-count) %>%  
 kable()

| from\_id\_supertype | from\_id\_sub\_type | count |
| --- | --- | --- |
| Event | Communication | 1620 |
| Green Guardians | NA | 90 |
| Relationship | Coordinates | 81 |
| Mako | NA | 78 |
| Reef Guardian | NA | 72 |
| Relationship | AccessPermission | 63 |
| Oceanus City Council | NA | 62 |
| Remora | NA | 62 |
| Event | Monitoring | 59 |
| Neptune | NA | 57 |
| EcoVigil | NA | 51 |
| Sentinel | NA | 47 |
| Davis | NA | 46 |
| The Lookout | NA | 44 |
| Event | VesselMovement | 43 |
| Relationship | Operates | 41 |
| The Intern | NA | 37 |
| Event | Assessment | 33 |
| Horizon | NA | 32 |
| V. Miesel Shipping | NA | 32 |
| Himark Harbor | NA | 31 |
| Mrs. Money | NA | 31 |
| Miranda Jordan | NA | 30 |
| Clepper Jensen | NA | 27 |
| Liam Thorne | NA | 26 |
| Relationship | Suspicious | 26 |
| Relationship | Reports | 25 |
| Rodriguez | NA | 25 |
| Event | Enforcement | 23 |
| Paackland Harbor | NA | 22 |
| Boss | NA | 21 |
| The Middleman | NA | 19 |
| Event | Collaborate | 17 |
| Nadia Conti | NA | 17 |
| Small Fry | NA | 17 |
| Marlin | NA | 15 |
| Osprey | NA | 15 |
| Samantha Blake | NA | 15 |
| Serenity | NA | 15 |
| Event | TourActivity | 14 |
| Relationship | Jurisdiction | 14 |
| Defender | NA | 12 |
| Knowles | NA | 12 |
| Seawatch | NA | 12 |
| Elise | NA | 10 |
| Glitters Team | NA | 9 |
| Haacklee Harbor | NA | 9 |
| Sailor Shifts Team | NA | 9 |
| Northern Light | NA | 8 |
| The Accountant | NA | 8 |
| Sam | NA | 7 |
| Relationship | Unfriendly | 5 |
| Conservation Vessels | NA | 4 |
| Kelly | NA | 4 |
| Event | HarborReport | 3 |
| Event | TransponderPing | 3 |
| Mariner’s Dream | NA | 3 |
| Recreational Fishing Boats | NA | 3 |
| Port Security | NA | 2 |
| Relationship | Colleagues | 2 |
| Tourists | NA | 2 |
| Diving Tour Operators | NA | 1 |
| Event | Criticize | 1 |
| Event | Fishing | 1 |
| Sailor Shift | NA | 1 |

# Check the mapping  
mc3\_edges\_cleaned %>%  
 group\_by(to\_id\_supertype, to\_id\_sub\_type) %>%  
 summarize(count = n()) %>%  
 arrange(-count) %>%  
 kable()

| to\_id\_supertype | to\_id\_sub\_type | count |
| --- | --- | --- |
| Event | Communication | 584 |
| Event | Monitoring | 240 |
| Relationship | AccessPermission | 225 |
| Relationship | Coordinates | 207 |
| Event | VesselMovement | 160 |
| Relationship | Colleagues | 160 |
| Relationship | Operates | 129 |
| Event | Assessment | 103 |
| Nemo Reef | NA | 102 |
| Mako | NA | 89 |
| Relationship | Reports | 72 |
| Oceanus City Council | NA | 68 |
| Relationship | Suspicious | 68 |
| Relationship | Jurisdiction | 63 |
| Event | Collaborate | 62 |
| Remora | NA | 60 |
| Event | Enforcement | 47 |
| Neptune | NA | 45 |
| Himark Harbor | NA | 38 |
| Reef Guardian | NA | 38 |
| Green Guardians | NA | 37 |
| Sentinel | NA | 36 |
| V. Miesel Shipping | NA | 35 |
| Event | TourActivity | 32 |
| Horizon | NA | 32 |
| Paackland Harbor | NA | 29 |
| Mrs. Money | NA | 26 |
| Boss | NA | 24 |
| EcoVigil | NA | 21 |
| Miranda Jordan | NA | 20 |
| Nadia Conti | NA | 20 |
| Protected areas | NA | 20 |
| Clepper Jensen | NA | 19 |
| Davis | NA | 18 |
| The Intern | NA | 18 |
| Liam Thorne | NA | 17 |
| Seawatch | NA | 16 |
| Event | TransponderPing | 12 |
| Sailor Shifts Team | NA | 12 |
| Relationship | Unfriendly | 11 |
| Serenity | NA | 11 |
| Marlin | NA | 10 |
| Restricted Zone | NA | 10 |
| Sam | NA | 10 |
| Restricted areas | NA | 9 |
| Rodriguez | NA | 9 |
| South Dock | NA | 9 |
| The Lookout | NA | 9 |
| Haacklee Harbor | NA | 7 |
| Samantha Blake | NA | 7 |
| The Middleman | NA | 7 |
| Eastern reefs | NA | 6 |
| Elise | NA | 6 |
| Knowles | NA | 6 |
| Northern quadrant | NA | 6 |
| Relationship | Friends | 6 |
| Western quadrant | NA | 6 |
| Dolphin Bay | NA | 5 |
| Eastern Boundary | NA | 5 |
| Northern Light | NA | 5 |
| Western Boundary | NA | 5 |
| Eastern quadrant | NA | 4 |
| City Officials | NA | 3 |
| Coral Point | NA | 3 |
| Defender | NA | 3 |
| E7 | NA | 3 |
| Event | HarborReport | 3 |
| Osprey | NA | 3 |
| Route C | NA | 3 |
| The Accountant | NA | 3 |
| Berth 14 | NA | 2 |
| Castaway Cove | NA | 2 |
| Eastern Islands | NA | 2 |
| Eastern Shoals | NA | 2 |
| Event | Criticize | 2 |
| Event | Fishing | 2 |
| Mariner’s Dream | NA | 2 |
| Southern Boundary | NA | 2 |
| Southern coastline | NA | 2 |
| Southern islands | NA | 2 |
| Azure Cove | NA | 1 |
| Conservation Vessels | NA | 1 |
| Eastern Coastline | NA | 1 |
| Glitters Team | NA | 1 |
| Kelly | NA | 1 |
| Port Security | NA | 1 |
| Recreational Fishing Boats | NA | 1 |
| Small Fry | NA | 1 |
| Southern quadrant | NA | 1 |

**Under Event-Communication types:** The edges target type and target subtypes matches the count of 584 for node to\_id\_supertype and node to\_id\_sub\_type. However, there were only 581 count for content within the original node file. We then looked into duplicates.

## 5.3 Checking for Duplicates

# checking for duplicates  
duplicate\_values1 <- mc3\_nodes\_cleaned %>%  
 count(content) %>%  
 filter(n > 1)  
  
# View duplicates  
print(duplicate\_values1)

# A tibble: 5 × 2  
 content n  
 <chr> <int>  
1 Boss, The Accountant here. Conservation vessels deploying underwater mi… 2  
2 Davis here to V. Miesel Shipping. Crew reallocation from Remora to Nept… 2  
3 Mrs. Money, this is Middleman. I've redirected Council's attention to o… 2  
4 Rodriguez, Davis here. Maintain current position with Mako at Nemo Reef… 2  
5 <NA> 575

There were 4 duplicates within the content column. Upon checking the original data, one was the sender and the other was the receiver who received the same content. We left the data as it was.

# **6) Tackling Question 1**

## 6.1 All Communications

Since the content column determines the important events and relationships, the communication sub\_type in the original nodes file would be useful. Thus, the communication to\_id\_sub\_type and from\_id\_sub\_type from the original edges file will also be useful.

### 6.1.1 Creation of a timeline of all radio communications in table format

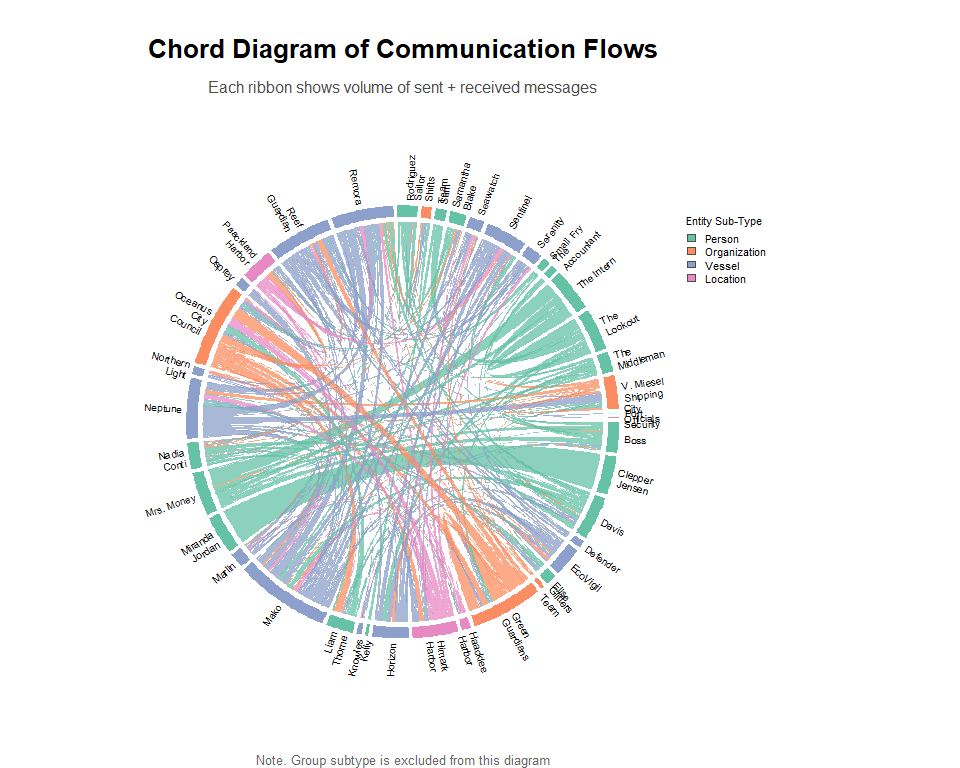
## The Table

|timestamp |sender\_name |recipient\_name |content |  
|:-------------------|:-----------|:--------------------|:------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|  
|2040-10-05 10:48:00 |Sam |Kelly |Hey Kelly, it's Sam. This permit approval seems fishy. Could you get details on who signed off on it while you're at the harbor? I need to understand these 'special access corridors' before my meeting with Elise tomorrow. |  
|2040-10-01 08:13:00 |Kelly |Sam |Sam, it's Kelly! Let's meet at Sunrise Point at 7 AM for birdwatching. Bring your new binoculars and some water. I've heard there might be some rare shorebirds passing through this weekend. Can't wait! |  
|2040-10-07 08:11:00 |Kelly |Sam |Hey Sam, it's Kelly! Got some photos of those crates - definitely marked 'fragile' and 'handle with care.' Neptune's crew is being super secretive. They're using special loading equipment I've never seen before. See you at 7 for birdwatching! |  
|2040-10-05 09:45:00 |Nadia Conti |Haacklee Harbor |Haacklee Harbor, this is Nadia Conti. I need to cancel the special access corridor arrangements for Nemo Reef immediately. Plans have changed due to unforeseen circumstances. Destroy all related documentation. I'll contact you when we're ready to proceed with alternative locations. |  
|2040-10-05 09:49:00 |Nadia Conti |Oceanus City Council |This is Nadia Conti. My cancellation was due to scheduling conflicts with our tourism development initiatives. I wasn't aware of any permit approvals. I'll submit revised documentation for alternative sustainable tourism proposals next week. |  
|2040-10-08 08:18:00 |Nadia Conti |Liam Thorne |Liam, Nadia here. Need your services urgently. Investigation brewing around Nemo Reef permits. Double your usual fee if you can ensure Harbor Master remains cooperative through next week. Meet at the usual place tomorrow, 10PM. |  
|2040-10-08 08:25:00 |Nadia Conti |Neptune |Neptune, this is Nadia. Need clarity on 'underwater foundation work' at Nemo Reef. This extends beyond our agreed scope. Meet me at the marina tomorrow at 6AM to discuss implications and additional resource requirements. |  
|2040-10-08 08:30:00 |Nadia Conti |Neptune |Neptune, Nadia here. Just confirming our 0600 meeting at the marina. I've reviewed the modified equipment specs with The Accountant. Please bring detailed timeline for foundation work and cost implications. We need to stay under radar. |  
|2040-10-08 11:23:00 |Nadia Conti |Marlin |Marlin, Nadia here. I understand you're inquiring about eastern shoal routes. Those are temporary diversions due to equipment transport needs. I'll have Davis provide the necessary documentation tonight. Nothing to be concerned about. |  
|2040-10-12 08:44:00 |Nadia Conti |Liam Thorne |Liam, Nadia here. Redirect all remaining operations from southwest immediately. Move equipment to our secondary location. I'll handle EcoVigil through proper channels. Meet me at the usual place at 2100 hours with updated documentation. |

## The Code

# --- 1. Extract All Communications ---  
# Logic: Sender (source) --sent--> Event\_Communication (target) --received--> Recipient (target)  
# This extracts all communication events  
  
# --- 2. Clean and Prepare Nodes ---  
mc3\_nodes\_cleaned <- mc3\_nodes\_raw %>%  
 mutate(id = as.character(id)) %>%  
 filter(!is.na(id)) %>%  
 distinct(id, .keep\_all = TRUE) %>%  
 # Rename 'type' to 'supertype' to reduce confusion with communication type  
 rename(supertype = type) %>%  
 # Select only columns that are needed and are consistently present  
 select(id, name, sub\_type, content, timestamp)   
  
# --- 3. Clean and Prepare Edges ---  
# Rename 'type' in edges to 'edge\_type' to avoid conflict with node 'supertype'  
mc3\_edges\_cleaned <- mc3\_edges\_raw %>%  
 rename(from\_id = source,  
 to\_id = target,  
 edge\_type = type) %>% # Renamed 'type' to 'edge\_type'  
 mutate(across(c(from\_id, to\_id), as.character)) %>%  
 # Filter out any edges where from\_id or to\_id are not in cleaned nodes  
 filter(from\_id %in% mc3\_nodes\_cleaned$id,  
 to\_id %in% mc3\_nodes\_cleaned$id)  
  
other\_communications\_df <- mc3\_edges\_cleaned %>%  
 filter(edge\_type == "sent") %>% # Start with 'sent' edges  
 # Join with nodes to get content and timestamp of the Event\_Communication node  
 left\_join(mc3\_nodes\_cleaned %>% select(id, content, timestamp),  
 by = c("to\_id" = "id")) %>%  
 rename(event\_id = to\_id, event\_content = content, event\_timestamp = timestamp) %>%  
 # Now, find the recipient of this communication event  
 left\_join(mc3\_edges\_cleaned %>%  
 filter(edge\_type == "received") %>%  
 select(event\_id\_match = from\_id, recipient\_id = to\_id),  
 by = c("event\_id" = "event\_id\_match")) %>%  
 # Join with nodes to get the sender's name and sub\_type  
 left\_join(mc3\_nodes\_cleaned %>% select(id, name, sub\_type),  
 by = c("from\_id" = "id")) %>%  
 rename(sender\_id\_actual = from\_id, sender\_name = name, sender\_sub\_type = sub\_type) %>%  
 # Join with nodes to get the recipient's name and sub\_type  
 left\_join(mc3\_nodes\_cleaned %>% select(id, name, sub\_type),  
 by = c("recipient\_id" = "id")) %>%  
 rename(recipient\_name = name, recipient\_sub\_type = sub\_type) %>%  
 # Select and rename final columns for all communications  
 select(  
 communication\_type = edge\_type, # This will be "sent" from original filter  
 sender\_id = sender\_id\_actual,  
 sender\_name,  
 sender\_sub\_type,  
 recipient\_id,  
 recipient\_name,  
 recipient\_sub\_type,  
 event\_id,  
 content = event\_content,  
 timestamp = event\_timestamp  
 )   
  
# create a timeline visualization or inspect content.  
print(knitr::kable(head(other\_communications\_df %>%  
 select(timestamp, sender\_name,   
 recipient\_name, content), 10),  
 format = "markdown", align = "l"))

### 6.1.2 Static Chord Diagram- All Communicators



**Findings:**

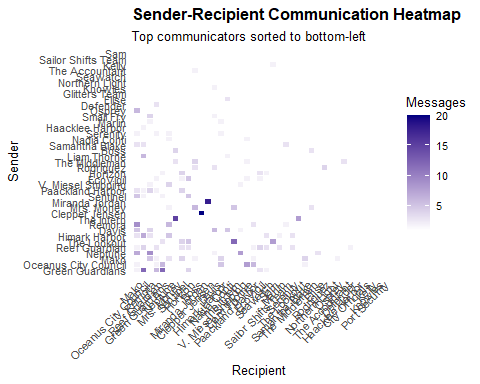
The thickness of each ribbon (chord) represents the magnitude of the relationship. A thicker ribbon represents more frequent communications (sent + received) between a sender and recipient.

Here, we have an overview of paired communicators who have higher frequencies. We also can see the links between communicators. These are the entities who communicated frequently with others that we might want to focus on:

* Person: The Intern, The Lookout, Clepper Jensen, Davis, Miranda Jordan, Mrs. Money.
* Organization: Oceanus City Council, Green Guardian
* Vessel: Reef Guardian, Neptune, Mako, Remora
* Location: Himark Habor
* Group: N/A

### 6.1.3 Heatmap of Correspondences

## Heatmap



## The Code

# Step 1: Count interactions  
adj\_df <- other\_communications\_df %>%  
 count(sender\_name, recipient\_name, name = "count")  
  
# Step 2: Compute total sent and received counts  
sender\_order <- adj\_df %>%  
 group\_by(sender\_name) %>%  
 summarise(total\_sent = sum(count)) %>%  
 arrange(desc(total\_sent)) %>%  
 pull(sender\_name)  
  
recipient\_order <- adj\_df %>%  
 group\_by(recipient\_name) %>%  
 summarise(total\_received = sum(count)) %>%  
 arrange(desc(total\_received)) %>%  
 pull(recipient\_name)  
  
# Step 3: Reorder factor levels  
adj\_df <- adj\_df %>%  
 mutate(  
 sender\_name = factor(sender\_name, levels = sender\_order),  
 recipient\_name = factor(recipient\_name, levels = recipient\_order)  
 )  
  
# Step 4: Plot heatmap  
ggplot(adj\_df, aes(x = recipient\_name, y = sender\_name, fill = count)) +  
 geom\_tile(color = "white") +  
 scale\_fill\_gradient(low = "white", high = "navyblue") +  
 labs(  
 title = "Sender-Recipient Communication Heatmap",  
 subtitle = "Top communicators sorted to bottom-left",  
 x = "Recipient",  
 y = "Sender",  
 fill = "Messages"  
 ) +  
 theme\_minimal(base\_size = 10) +  
 theme(  
 axis.text.x = element\_text(angle = 45, hjust = 1),  
 plot.title = element\_text(size = 12, face = "bold"),  
 plot.subtitle = element\_text(size = 10),  
 panel.grid = element\_blank()  
 )

**Findings:**

After extraction of the entities who communicated frequently (from the Static Chord Diagram), we tabled who they communicated with by using the heatmap. E.g. Name1 communicated with Name2.

| Name 1 Subtype | Name1 | Name2 |
| --- | --- | --- |
| Person | The Intern | The Lookout, Mrs. Money |
| Person | Clepper Jensen | Miranda Jordan |
| Person | Davis | Neptune |
| Person | Mrs. Money | The Intern, The Middleman, Boss |
| Vessel | Mako | Remora, Green Guardians, Oceanus City Council, Neptune, Reef Guardians, Himark Harbor, Davis, Sentinel, Paackland Habor, Samantha Blake, Serenity, Osprey |
| Vessel | Remora | Mako, Neptune, Himark Habor, Davis, Paackland Harbor, V. Miesel Shipping, Marlin, Small Fry |
| Vessel | Neptune | Mako, Himark Habor, Remora, Mrs Money, V. Miesel Shipping, Nadia, Serenity |
| Vessel | Reef Guardian | Green Guardians, Oceanus City Council, Mako, Paackland Harbor, EcoVigil, Serenity, Defender |
| Organization | Green Guardian | Green Guardians, Oceanus City Council, The Lookout, Sentinel, Horizon |
| Organization | Oceanus City Council | Green Guardians, Reef Guardians, Himark Harbor, Sentinel, Paackland Harbor, Liam Thorne, Samantha Blake, Haacklee Harbor |
| Location | Himark Habor | Oceanus City Council, Mako, Serenity, Marlin |

# **7) Tackling Question 2**

In section 7 and 8, we attempted to find answers to question 2 and 4, as we explored other interesting information.

# Question 2a)

### 7.1.2 Unique count from each sender subtype

## Unique Count

# A tibble: 4 × 3  
 sender\_sub\_type unique\_senders sender\_names   
 <chr> <int> <chr>   
1 Person 17 Boss, Clepper Jensen, Davis, Elise, Kelly, Lia…  
2 Vessel 14 Defender, EcoVigil, Horizon, Knowles, Mako, Ma…  
3 Organization 5 Glitters Team, Green Guardians, Oceanus City C…  
4 Location 3 Haacklee Harbor, Himark Harbor, Paackland Harb…

## The Code

sender\_names\_by\_type <- other\_communications\_df %>%  
 group\_by(sender\_sub\_type) %>%  
 summarise(  
 unique\_senders = n\_distinct(sender\_name),  
 sender\_names = paste(sort(unique(sender\_name)), collapse = ", ")  
 ) %>%  
 arrange(desc(unique\_senders)) # sort from largest to smallest  
  
# View the table  
print(sender\_names\_by\_type)

There were no Group sub-type as confirmed by this filtered table.

## 7.2 People and Vessels

Due to a large number of entity subtypes as people and vessels in 7.1.4, we then looked into the characters’ and vessel’s identity and their actual communication content.

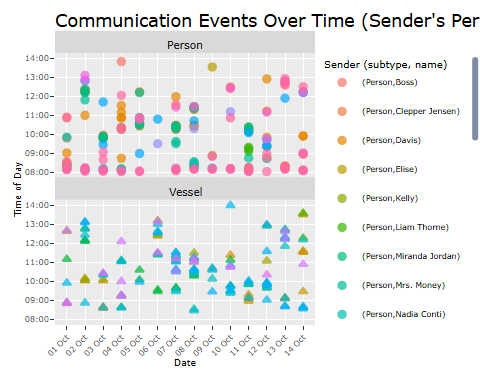
### 7.2.1 Filter by Vessel and Person only

## The Code

person\_vessel\_df <- other\_communications\_df %>%  
 filter(  
 (sender\_sub\_type == "Person" & recipient\_sub\_type == "Vessel") |  
 (sender\_sub\_type == "Vessel" & recipient\_sub\_type == "Person") |  
 (sender\_sub\_type == "Person" & recipient\_sub\_type == "Person") |  
 (sender\_sub\_type == "Vessel" & recipient\_sub\_type == "Vessel")  
 )

### 7.2.2 Plotted Timeline- People & Vessels

# --- FACTORING and DATETIME CLEANING ---  
person\_vessel\_df\_for\_plot <- person\_vessel\_df %>%  
 mutate(  
 timestamp = as.POSIXct(timestamp),  
 comm\_date = as.Date(timestamp),  
 comm\_time\_of\_day = hms::as\_hms(format(timestamp, "%H:%M:%S")),  
 sender\_sub\_type = factor(sender\_sub\_type, levels = c("Person", "Vessel")),  
 communicating\_pair\_sorted = paste(pmin(sender\_name, recipient\_name), pmax(sender\_name, recipient\_name), sep = " & ")  
 )  
  
# --- WRAPPING CONTENT AND TOOLTIP ---  
plot\_data1 <- person\_vessel\_df\_for\_plot %>%  
 mutate(  
 timestamp = as.POSIXct(timestamp),  
 date = as.Date(timestamp),  
 time = format(timestamp, "%H:%M:%S"),  
 wrapped\_content = str\_wrap(content, width = 50),  
 tooltip\_text = paste0(  
 "<b>Date:</b> ", date, "<br>",  
 "<b>Time:</b> ", time, "<br>",  
 "<b>From:</b> ", sender\_name, "<br>",  
 "<b>To:</b> ", recipient\_name, "<br>",  
 "<b>Event\_id:</b> ", event\_id, "<br><br>",  
 "<b>Content:</b><br>", wrapped\_content  
 )  
 )  
  
# Plot  
p <-ggplot(plot\_data1, aes(x = comm\_date, y = comm\_time\_of\_day)) +  
 geom\_point(aes(  
 color = sender\_id,  
 shape = sender\_sub\_type,  
 text = tooltip\_text  
 ),show.legend = c(color = TRUE, shape = FALSE),   
 size = 2, alpha = 0.7) +  
 scale\_shape\_manual(values = c("Person" = 16, "Vessel" = 17)) +  
 facet\_wrap(~ sender\_sub\_type, ncol = 1, scales = "fixed") +  
 scale\_y\_time(  
 limits = hms::as\_hms(c("08:00:00", "14:00:00")), # reversed to show time top-to-bottom  
 breaks = hms::as\_hms(c("08:00:00", "09:00:00", "10:00:00", "11:00:00", "12:00:00", "13:00:00", "14:00:00")),  
 labels = c("08:00", "09:00", "10:00", "11:00", "12:00", "13:00", "14:00")  
)+  
 scale\_x\_date(  
 date\_breaks = "1 day",  
 date\_labels = "%d %b"  
)+  
 labs(  
 title = "Communication Events Over Time (Sender's Perspective)",  
 x = "Date",  
 y = "Time of Day",  
 color = "Sender (subtype, name)"  
 ) +  
 theme\_grey() +  
 theme(  
 axis.text.y = element\_text(size = 6),  
 axis.title.y = element\_text(size = 7),  
 axis.ticks.y = element\_line(),  
 axis.text.x = element\_text(size = 6, angle = 45, hjust = 1),  
 axis.title.x = element\_text(margin = margin(t = 8), size = 7),  
 panel.spacing = unit(0.5, "lines"), # Applies to both x and y spacing  
 strip.text = element\_text(size = 8, face = "bold"),  
 legend.position = "bottom",  
 legend.text = element\_text(size = 6),  
 legend.title = element\_blank()  
 )  
  
# --- Convert to interactive plot ---  
ggplotly(p, tooltip = "tooltip\_text")



#### 7.2.2.1 Findings on People/ Vessels:

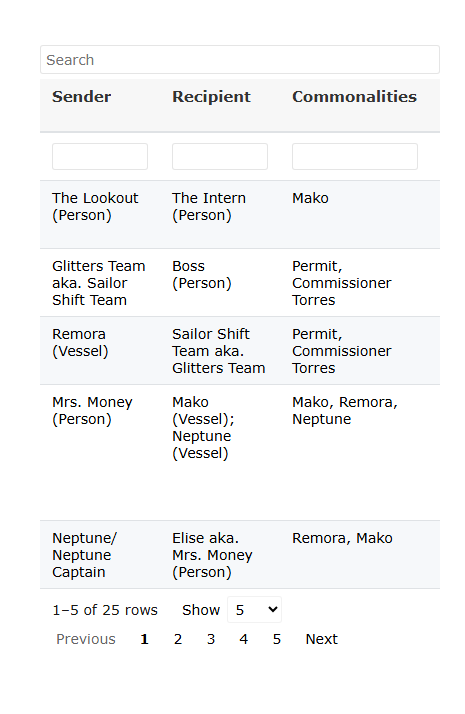
* Core logic:
  + If two names appear as sender and recipient in the same message, they cannot belong to the same person — i.e., they’re not aliases of each other.
  + If two names sent a message at the exact time, they cannot belong to the same person.
  + For instance, if Nadia sent a message to The Accountant, they would not be the same individual. If Nadia sent a message at 10am to The Accountant and The Lookout also sent a message at 10am to The Intern, Nadia and The Lookout cannot be the same person.
* Select only The Accountant, Mrs. Money, Elise: We see close timings between Mrs. Money and Elise on 8 Oct, and 10 Oct. These were on the same topic. Elise then disappears from radar on 10 Oct. She reappears as The Accountant and Mrs. Money on 11 Oct on the same topic and remains only as The Accountant till 14 Oct.
* Select only Liam and The Middleman: The Middleman disappeared on 7 Oct and appeared as Liam on 8 Oct. On 11 Oct Mrs. Money asked The Middleman if anything was found by conservation vessels. On the same day, Liam reappeared and replied Elise that nothing was found by them.
* Select only The Boss and Nadia: The Boss disappeared on 5 Oct and reappeared as Nadia on 8 Oct. Likely the same person.
* Select only Small Fry and Rodriguez: on 2 Oct Rodriguez corresponded with Remora and Mako on meeting at the slip #14. It happened again on 14 Oct as he took on dual roles and responded to the same message with different names. Likely the same person.
* Select only The Lookout and Sam: on 7 Oct Sam asked Kelly to get information on who authorized the permit. 2 minutes later, The Lookout (Kelly) responded to The Intern (Sam), that it was signed by Jensen from City Council.
* Seawatch only appeared on 10 Oct but Horizon talked to Seawatch on 8 Oct. Therefore, some other entity is Seawatch before or during 8 Oct. Defender told Seawatch on 3 Oct at 8.39am that it increased its patrol and informed Seawatch to maintain vigilance. The Lookout (Seawatch) responded to Sentinel (Defender) at 8.41am that it acknowledged the need for vigilance.
* After getting a general idea of the characters’ involvement, we looked at the interactions in 7.3.

## 7.3 Discussion/ Interpretation on all communications:

From the interactive plot, we were able to select certain names of individuals and/or vessels from the legend and read their conversations. We segmented entities based on our first cut of observations of the following people or vessels and rated them as suspicious or non-suspicious. We tabled out what we have learnt with variables: Sender, Recipient, Commonalities, Rationale, Suspicious?, Date, Event Id, in Section 7.3.1 and 7.3.2.

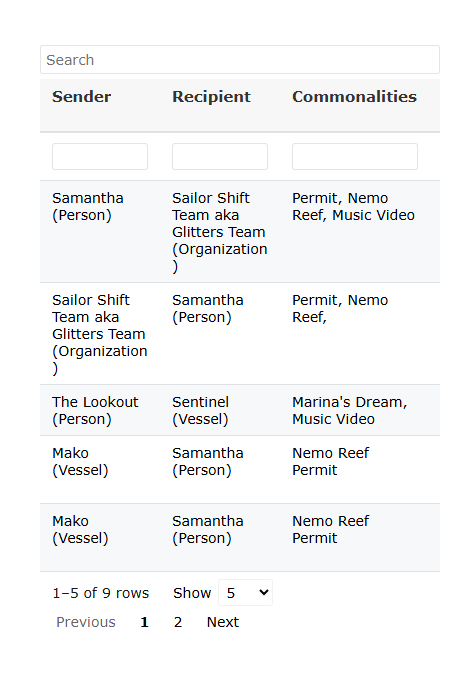
### 7.3.1 Suspicious Conversations

library(reactable)  
library(dplyr)  
library(stringr)  
library(tidyr) # For separate\_rows, if needed for 'Event Id' later  
  
# Create the data frame using tribble, consolidating multi-line entries  
df\_new\_data <- tribble(  
 ~Sender, ~Recipient, ~Commonalities, ~Rationale, ~Suspicious\_Status, ~Date, ~Event\_Id,  
 "The Lookout (Person)", "The Intern (Person)", "Mako", "Jensen from City Council signed a permit to set a corridor for Mako vessel.", "Jensen", "5/10", "371",  
 "Glitters Team aka. Sailor Shift Team", "Boss (Person)", "Permit, Commissioner Torres", "Glitters thanked their Boss who got Commissioner Torres to sign off the permit", "Nadia aka The Boss", "6/10", "389",  
 "Remora (Vessel)", "Sailor Shift Team aka. Glitters Team", "Permit, Commissioner Torres", "Davis thanked Nadia who got Commissioner Torres to sign off the permit", "Nadia aka The Boss", "6/10", "388",  
 "Mrs. Money (Person)", "Mako (Vessel); Neptune (Vessel)", "Mako, Remora, Neptune", "Mrs. Money is the Financial Controller who provides the payment protocols for Mako and Remora's captions. She also provides the encryption protocols and transfer devices. Mrs. Money coordinates payment protocols for Remora and Mako captains", "Elise aka. Mrs. Money", "7/10", "458 and 464",  
 "Neptune/ Neptune Captain", "Elise aka. Mrs. Money (Person)", "Remora, Mako", "Neptune told her to coordinate payment protocols for Remora and Mako captains", "Elise aka. Mrs. Money", "7/10", "456 and 518",  
 "Remona (Vessel)", "Rodriquez (Person)", "Sailor Shift", "Nemo Reef selected for Sailor Shift's music shoot.; Nadia personally handled permits for Nemo Reefs after meeting Davis.", "Both", "2/10", "153 and 328",  
 "Remona (Vessel)", "Nadia (Person)", "Delta 3, Sailor Shift", "Tourism facade created by Remona and awaiting installation of underwater lighting placements at Nemo Reef.", "Both", "14/10", "943",  
 "Davis (Person)", "Mako (Vessel)", "Rodriquez, Nadia, Neptune", "Rodriquez will help with logistics.", "Both", "2/10", "115",  
 "Davis (Person)", "Mako (Vessel)", "Rodriquez, Nadia, Neptune", "V. Miesel's Marine Research Permit is CR-7844. There is a 5 day deadline from 8 Oct.", "Both", "8/10", "548",  
 "Davis (Person)", "Mako (Vessel)", "Rodriquez, Nadia, Neptune", "V. Miesel approved their operational authority.", "Both", "12/10", "802",  
 "Davis (Person)", "Remona (Vessel)", "Client", "Davis wants to maintain client's privacy with Paackland Harbour.", "Both, and also Paackland Harbour", "4/10", "282",  
 "Mako (Vessel)", "Davis (Person)", "Permit CR-7844, Boss", "Davis is a captain.", "Both", "5/10", "349",  
 "Mako (Vessel)", "Boss (Person)", "Boss, Samantha", "Oceanus Council approved Mako's departure to Nemo Reef with minimal documentation.", "Oceanus Council", "5/10", "365",  
 "Boss (Person)", "Marko (Vessel)", "Mission, Boss", "Informed Marko to abort mission as conservation patrols are tracking it", "Both", "4/10", "316",  
 "Small Fry aka Rodriquez (Person)", "Remora (Vessel, Mako (Vessel)", "Slip #14, Glitters Team, Sailor Team", "Mako asked Rodriquez when he could come by slip #14.; Small Fry replied he would come by slip #14 in 30 minutes time.; Small Fry surprised that Glitters Team filing for paperwork downtown for protected area of Nemo Reef.", "Both", "2/10", "142, 143 and 156",  
 "Neptune", "Nadia (Person); Neptune (Vessel)", "Equipment specs", "Neptune asked to meet Nadia at 0600 and confirm the equipment specs approved by Elise.; Nadia confirmed the time with Neptune and agreed to show her reviewed equipment specs from The Accountant.", "Both; Elise aka The Accountant", "8/10", "537 and 738",  
 "Knowles (Vessel)", "Davis (Person)", "Boss", "Instructed by Mako to provide equipment transfer.", "Both", "14/10", "1001 and 1003",  
 "Sentinel(Vessel)", "Mako (Vessel)", "Permit", "Mako told Sentinel it was operating under NR-1045.", "Mako", "7/10", "467",  
 "Mako (Vessel)", "Sentinel (Vessel)", "Permit", "Sentinel asked the reasons for Mako operating under permit NR-1045", "Mako", "8/10", "574",  
 "Mako (Vessel)", "Sentinel (Vessel)", "Permit", "Mako told Sentinel it was operating under permit CR-7844 (marine equipment transport) not NR-1045", "Mako", "8/10", "575",  
 "Miranda (Person)", "Clepper Jesen (Person)", "Rodriquez", "Rodriquez filed for permits under Sailor Shifts Team for 2 commercial and 1 private yatch to perform environmental sampling.", "Rodriquez is suspicious. The communicators are not suspicious.", "2/10", "130",  
 "Miranda (Person)", "Clepper Jesen (Person)", "Rodriquez", "Rodriquez is connected to a mining consortium with previous violations in protected areas.", "Rodriquez is suspicious. The communicators are not suspicious", "2/10", "134",  
 "Miranda (Person)", "Clepper Jesen (Person)", "Rodriquez", "Rodriquez is connected to Council Knowles and they are related to V. Miesel Shipping which is possibly a shell company which has vessels.", "Rodriquez is suspicious. The communicators are not suspicious.", "3/10", "201",  
 "Miranda (Person)", "Clepper Jesen (Person)", "Rodriquez is suspicious", "Council Knowles' brother-in-law owns the offshore bank account to V. Miesel Shipping. The company is involved in rare earth extraction.", "Rodriquez is suspicious. The communicators are not suspicious.", "3/10", "204",  
 "Miranda (Person)", "Clepper Jesen (Person)", "Rodriquez is suspicious", "Knowles accepted cash from Rodriquez.", "Rodriquez is suspicious. The communicators are not suspicious.", "3/10", "206"  
)  
  
# Process the data (primarily for consistent whitespace and Event\_Id splitting if necessary)  
df\_processed <- df\_new\_data %>%  
 # Standardize delimiters and remove excess whitespace across all character columns.  
 mutate(  
 across(where(is.character), ~ str\_replace\_all(.x, "[,\n]+", ", ") %>% str\_trim())  
 ) %>%  
 # For Event\_Id, replace "and" with comma and then separate rows if needed  
 mutate(Event\_Id = str\_replace\_all(Event\_Id, " and ", ", ")) %>%  
 mutate(across(where(is.character), str\_trim)) %>%  
 mutate(Event\_Id = as.character(Event\_Id)) # Ensure Event\_Id is character  
  
# Create the reactable table  
reactable(  
 df\_processed,  
 filterable = TRUE,  
 searchable = TRUE,  
 paginationType = "numbers",  
 defaultPageSize = 5,  
 showPageSizeOptions = TRUE,  
 pageSizeOptions = c(5, 10, 20, 50, 100),  
 striped = TRUE,  
 highlight = TRUE,  
 columns = list(  
 Sender = colDef(name = "Sender", minWidth = 120),  
 Recipient = colDef(name = "Recipient", minWidth = 120),  
 Commonalities = colDef(name = "Commonalities", minWidth = 150),  
 Rationale = colDef(name = "Rationale", minWidth = 300),  
 # Rename Suspicious\_Status for display  
 Suspicious\_Status = colDef(name = "Suspicious?", minWidth = 150),  
 # Date column as text, no special date sorting  
 Date = colDef(name = "Date", minWidth = 80, align = "center"),  
 Event\_Id = colDef(name = "Event Id", minWidth = 80, align = "center")  
 ),  
 theme = reactableTheme(  
 borderColor = "#dfe2e5",  
 stripedColor = "#f6f8fa",  
 highlightColor = "#f0f5f9",  
 cellPadding = "8px 12px",  
 style = list(fontFamily = "Verdana, Geneva, sans-serif", fontSize = "14px"),  
 headerStyle = list(  
 "&.rt-th:hover" = list(backgroundColor = "#e0e6eb"),  
 fontSize = "15px",  
 fontWeight = 600,  
 color = "#333",  
 background = "#f7f7f7"  
 ),  
 rowSelectedStyle = list(backgroundColor = "#e6f2ff", "&:hover" = list(backgroundColor = "#e6f2ff")),  
 searchInputStyle = list(width = "100%", margin = "5px 0", padding = "5px"),  
 filterInputStyle = list(width = "100%", margin = "2px 0", padding = "4px")  
 )  
)



### 7.3.2 Non-suspicious conversations

# Load necessary packages  
library(reactable)  
library(dplyr)  
library(stringr)  
library(tidyr) # For separate\_rows  
library(lubridate) # For date parsing  
  
# 1. Manually parse the data into a tribble  
# Each row in the tribble corresponds to a logical entry from provided text.  
# Multi-line strings within a cell are captured using '\n'.  
# The 'Suspicious?' column is named 'Suspicious\_Status' in R to avoid special characters,  
# and will be renamed in the reactable column definition for display.  
df\_raw\_input <- tribble(  
 ~Sender, ~Recipient, ~Commonalities, ~Rationale, ~Suspicious\_Status, ~Date, ~Event\_Id,  
 "Samantha (Person)", "Sailor Shift Team aka Glitters Team (Organization)", "Permit, Nemo Reef, Music Video", "Samantha was assisting Sailor Shift Team to coordinate the permit and equipment rental for their music video and wanted confirmation on Nemo Reef as their intended location.", "Unsure", "1/10", "64",  
 "Sailor Shift Team aka Glitters Team (Organization)", "Samantha (Person)", "Permit, Nemo Reef,", "Glitters Team confirmed with Samantha that they would like to use Nemo Reef as their primary location", "Unsure", "1/10", "66",  
 "Mako (Vessel)", "Samantha (Person)", "Nemo Reef Permit", "Mako claims that Samantha has a coordination team. Preparing for an event in a week's time from 6 Oct.", "Samantha is not", "6/10", "380",  
 "Mako (Vessel)", "Samantha (Person)", "Nemo Reef Permit", "Samantha advised them to stop activities which might be illegal without permit", "Samantha is not", "10/10", "687",  
 "The Lookout (Person)", "Sentinel (Vessel)", "Marina's Dream, Music Video", "Spotted 3 suspicious vessels around Nemo Reef on 3/10.", "Both are not", "3/10", "171",  
 "The Lookout (Person)", "Sentinel (Vessel)", "Marina's Dream, Music Video", "Spotted music video production on 13/10", "Both are not", "13/10", "912",  
 "The Lookout (Person)", "Horizon (Vessel)", "Music Video", "Found it suspicious to spot lighting rigs and cameras at 12pm. Wanted Horizon to investigate", "Both are not", "13/10", "926",  
 "EcoVigil (Vessel)", "Liam (Person)", "ROVs", "Collection of water samples using ROVs", "EcoVigil is not", "11/10", "737 and 751" # "and" will be handled by separate\_rows  
)  
  
# 2. Process the data for display and sorting  
df\_processed <- df\_raw\_input %>%  
 # Standardize delimiters and remove excess whitespace across all character columns.  
 # This flattens multi-line strings within cells and makes 'separate\_rows' more reliable.  
 mutate(  
 across(where(is.character), ~ str\_replace\_all(.x, "[,\n]+", ", ") %>% str\_trim())  
 ) %>%  
 # Handle the "and" in Event\_Id, converting it to a comma for consistent splitting.  
 mutate(Event\_Id = str\_replace\_all(Event\_Id, " and ", ", ")) %>%  
 # Now, use separate\_rows for Event\_Id if it contains multiple comma-separated values.  
 # 'Date' column appears to be single-valued in this dataset, so it's not included here.  
 # If Date could also have multiple values (e.g., "1/10, 2/10"), you would add it to separate\_rows.  
 separate\_rows(Event\_Id, sep = ", ", convert = FALSE) %>%  
 # Trim whitespace again after separation in case of leading/trailing spaces  
 mutate(across(where(is.character), str\_trim)) %>%  
 # Convert Date to actual date objects for proper sorting.  
 # Using `dmy` for day/month/year format and assuming the current year (2025).  
 mutate(  
 FullDate = paste0(Date, "/2025"), # Add the current year for correct parsing  
 ParsedDate = dmy(FullDate, quiet = TRUE) # Convert to Date object (Day Month Year)  
 ) %>%  
 # Order the data by parsed date and then event ID  
 arrange(ParsedDate, Event\_Id) %>%  
 select(-FullDate) %>% # Remove the helper column  
 # Ensure Event\_Id is character type for display purposes  
 mutate(Event\_Id = as.character(Event\_Id))  
  
# 3. Create the reactable table  
reactable(  
 df\_processed,  
 filterable = TRUE, # Enable column filters at the top of each column  
 searchable = TRUE, # Add a global search box above the table  
 paginationType = "numbers", # Display page numbers (e.g., 1, 2, 3 instead of Prev/Next)  
 defaultPageSize = 5, # Show 5 rows per page by default  
 showPageSizeOptions = TRUE, # Allow users to select different page sizes  
 pageSizeOptions = c(5, 10, 20, 50, 100), # Available page size options  
 striped = TRUE, # Add alternating row colors for better readability  
 highlight = TRUE, # Highlight the row when the user hovers over it  
 # Define the default sorting for the table when it first loads  
 defaultSorted = list(ParsedDate = "asc"), # Sort by the hidden ParsedDate column in ascending order  
 columns = list(  
 # The 'ParsedDate' column is used internally for correct date sorting but is not displayed  
 ParsedDate = colDef(  
 show = FALSE  
 ),  
 # Define how each visible column should be displayed and behave  
 Date = colDef(  
 name = "Date", # Display name for the column header  
 defaultSortOrder = "asc", # Ensure initial sort is ascending for the displayed date  
 minWidth = 80, # Minimum width for the column to ensure content fits  
 align = "center" # Align text to the center for better presentation of dates  
 ),  
 Sender = colDef(name = "Sender", minWidth = 120),  
 Recipient = colDef(name = "Recipient", minWidth = 120),  
 Commonalities = colDef(name = "Commonalities", minWidth = 150),  
 Rationale = colDef(name = "Rationale", minWidth = 300),  
 # Custom definition for the 'Suspicious\_Status' column to display its original name  
 Suspicious\_Status = colDef(name = "Suspicious?", minWidth = 150),  
 Event\_Id = colDef(  
 name = "Event ID", # Display name for the column header  
 minWidth = 80, # Minimum width  
 align = "center" # Align text to the center  
 )  
 ),  
 # Customize the visual theme of the reactable table for a consistent look and feel  
 theme = reactableTheme(  
 borderColor = "#dfe2e5", # Defines the color of table borders  
 stripedColor = "#f6f8fa", # Defines the background color of striped rows  
 highlightColor = "#f0f5f9", # Defines the background color when a row is hovered over  
 cellPadding = "8px 12px", # Sets the padding inside table cells  
 style = list(fontFamily = "Verdana, Geneva, sans-serif", fontSize = "14px"), # General font family and size for table content  
 headerStyle = list(  
 "&.rt-th:hover" = list(backgroundColor = "#e0e6eb"), # Hover effect for table headers  
 fontSize = "15px", # Font size for headers  
 fontWeight = 600, # Font weight for headers (bold)  
 color = "#333", # Text color for headers  
 background = "#f7f7f7" # Background color for headers  
 ),  
 rowSelectedStyle = list(backgroundColor = "#e6f2ff", "&:hover" = list(backgroundColor = "#e6f2ff")), # Style for selected rows  
 searchInputStyle = list(width = "100%", margin = "5px 0", padding = "5px"), # Style for the global search input box  
 filterInputStyle = list(width = "100%", margin = "2px 0", padding = "4px") # Style for individual column filter input boxes  
 )  
)



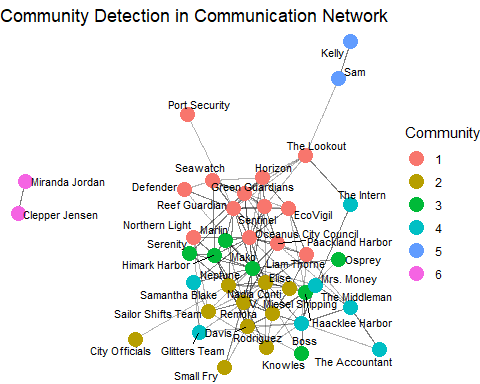
# Question 2b)

## 7.5 Community Detection and Centrality Measure- ggraph

### 7.5.1 Community Detection- All

Since we had an idea of the identity of the characters involved and their sub types, we wanted to uncover the communities they belonged to. We used the Louvian method for community detection to find communities that were densely connected internally but sparse to others.

## Community Detection



## The Code

set.seed(1234)   
  
# --- STEP 1: Prepare the edge list ---  
edge\_df <- other\_communications\_df %>%  
 select(sender\_id, recipient\_id) %>%  
 filter(!is.na(sender\_id) & !is.na(recipient\_id)) %>%  
 rename(from = sender\_id, to = recipient\_id) %>%  
 distinct()  
  
# Remove self-loops  
edge\_df <- edge\_df %>% filter(from != to)  
  
# --- STEP 2: Create the graph object (undirected) ---  
g <- tbl\_graph(edges = edge\_df, directed = FALSE)  
  
# --- STEP 3: Run Louvain community detection ---  
g <- g %>%  
 mutate(community = group\_louvain())  
  
# --- STEP 4: Visualize the graph ---  
ggraph(g, layout = "fr") +  
 geom\_edge\_link(alpha = 0.3) +  
 geom\_node\_point(aes(color = as.factor(community)), size = 5) +  
 geom\_node\_text(aes(label = name), repel = TRUE, size = 3) +  
 theme\_void() +  
 labs(title = "Community Detection in Communication Network",  
 color = "Community")

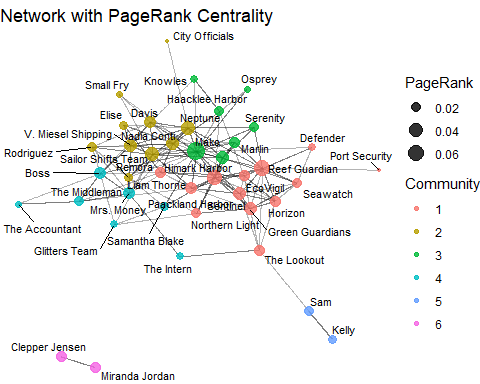
### 7.5.2 Centrality Measure- All

We then proceeded to use the PageRank centrality algorithm to assign a numerical weight to each node to reflect its importance. The community attribute we previously calculated was then assigned to this graph.

## PageRank Centrality Algorithm

# A tibble: 10 × 5  
 name pagerank degree betweenness closeness  
 <chr> <dbl> <dbl> <dbl> <dbl>  
 1 Mako 0.0756 37 220. 0.0179  
 2 Reef Guardian 0.0536 25 105. 0.0152  
 3 Oceanus City Council 0.0516 26 47.9 0.0147  
 4 Remora 0.0436 21 39.0 0.0139  
 5 Neptune 0.0410 19 56.2 0.0145  
 6 V. Miesel Shipping 0.0355 17 9.84 0.0130  
 7 Nadia Conti 0.0354 17 27.9 0.0141  
 8 Himark Harbor 0.0347 17 22.6 0.0130  
 9 Sentinel 0.0331 16 27.8 0.0143  
10 Green Guardians 0.0318 15 27.4 0.0143

## The Network



## The Code

set.seed(1234)   
  
# --- STEP: Compute Centrality Measures ---  
g <- g %>%  
 mutate(  
 pagerank = centrality\_pagerank(),  
 degree = centrality\_degree(),  
 betweenness = centrality\_betweenness(),  
 closeness = centrality\_closeness()  
 )  
  
# Show top 10 nodes by PageRank  
g %>%  
 as\_tibble() %>%  
 select(name, pagerank, degree, betweenness, closeness) %>%  
 arrange(desc(pagerank)) %>%  
 head(10)  
  
# Visualize by Centrality  
ggraph(g, layout = "fr") +  
 geom\_edge\_link(alpha = 0.3) +  
 geom\_node\_point(aes(size = pagerank, color = as.factor(community)), alpha = 0.8) +  
 geom\_node\_text(aes(label = name),  
 repel = TRUE,  
 size = 3,  
 max.iter = 5000) + # <--- Increased this value   
 theme\_void() +  
 labs(title = "Network with PageRank Centrality",  
 size = "PageRank", color = "Community")

#### **7.5.2.1 Findings**:

* There were around 6 closely associated groups. Community 5 (Clepper and Miranda) appeared to be segmented from the central group, due to the non-involvement from the nature of the investigative work.
* From the graph, we extracted the 11 influential nodes to focus on:
  + Community 1: Reef Guardian, EcoVigil
  + Community 2: Neptune, Remora, Nadia, V. Miesel Shipping, Davis
  + Community 3: Mako
  + Community 4: Mrs. Money, Boss, The Middleman
  + Community 5: N/A as they were not very influential at global level
  + Community 6: N/A as they were not very influential at global level

### 7.5.3 Wordclouds- Bigram

## The Wordclouds



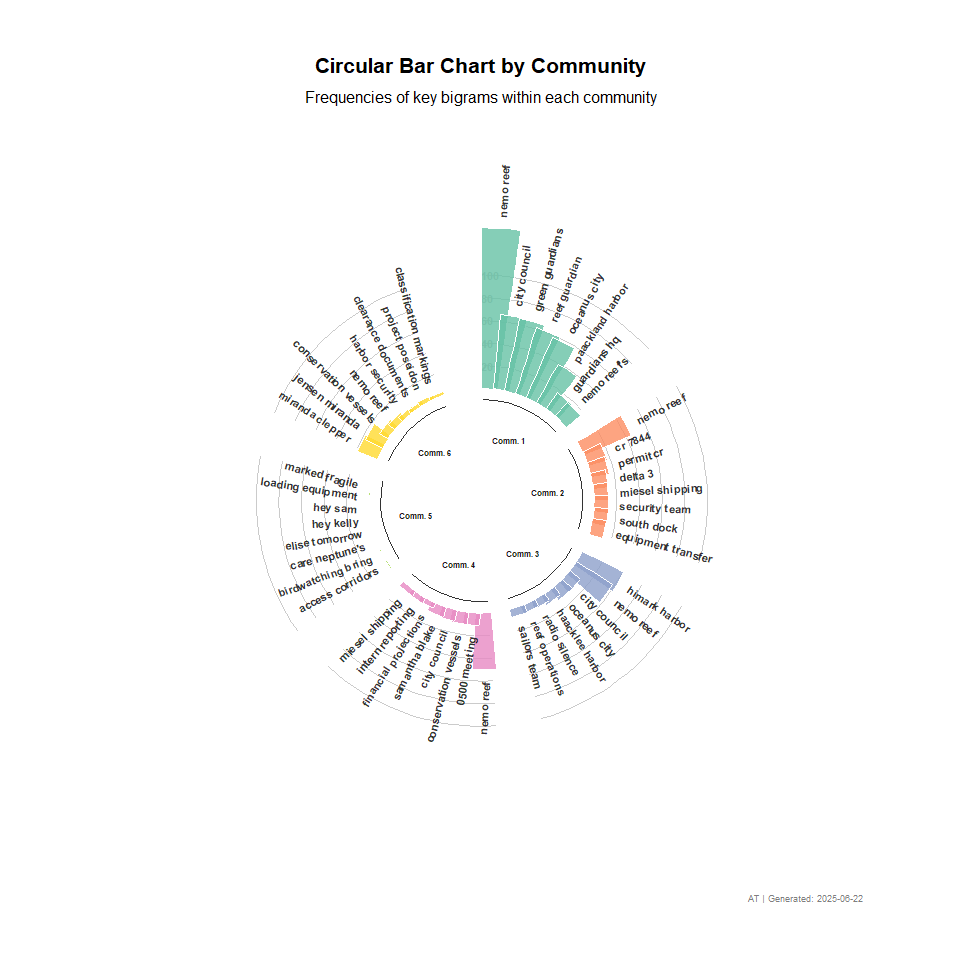
## The Code

# 1. Extract node community assignments  
g\_node\_communities <- g %>%  
 as\_tibble() %>%  
 select(sender\_name = name, community)  
  
# 2. Join communication content with community assignments  
content\_with\_community <- other\_communications\_df %>%  
 left\_join(g\_node\_communities, by = c("sender\_name")) %>%  
 filter(!is.na(community), !is.na(content))  
  
# 3. Unnest tokens for unigrams  
unigrams <- content\_with\_community %>%  
 unnest\_tokens(word, content, token = "words") %>%  
 anti\_join(stop\_words, by = "word") %>%  
 count(community, word, sort = TRUE)  
  
# 4. Unnest tokens for bigrams  
bigrams <- content\_with\_community %>%  
 unnest\_tokens(bigram, content, token = "ngrams", n = 2) %>%  
 separate(bigram, into = c("word1", "word2"), sep = " ") %>%  
 filter(!word1 %in% stop\_words$word,  
 !word2 %in% stop\_words$word) %>%  
 unite(bigram, word1, word2, sep = " ") %>%  
 count(community, bigram, sort = TRUE)  
  
# 5a. Plot word clouds per community  
par(mfrow = c(2, 3)) # 2 rows, 3 columns layout for 6 communities  
for (i in sort(unique(bigrams$community))) {  
 words <- bigrams %>% filter(community == i)  
   
 if (nrow(words) < 1) next # Skip if no words  
   
 suppressWarnings({  
 set.seed(432) # Set seed for reproducibility  
 wordcloud(words = words$bigram, # <-- FIXED here  
 freq = words$n,  
 max.words = min(20, nrow(words)),  
 scale = c(3, 0.5),  
 colors = brewer.pal(8, "Dark2"),  
 random.order = FALSE)  
 })  
 mtext(paste("Community", i), side = 3, line = 1, adj = 0.5, cex = 1.5, col = "black")  
}

* Certain words stood out more by their size in the unigram and bigrams. To view clearer frequencies and patterns among the community, we furthered the visualisations into circular barcharts of bigrams.

### 7.5.4 Circular barchart for Top Bigrams per Community

## The Circular Barchart



## allThe Code

# --- Configuration ---  
num\_top\_bigrams\_per\_community <- 8  
empty\_bar\_count <- 2 # gaps btw comm.  
#excluded\_community <- 5 # too little in community 5  
  
# --- 1. Prepare the Combined Dataset ---  
all\_communities\_data <- bigrams %>%  
# filter(community != excluded\_community) %>%  
 group\_by(community) %>%  
 arrange(desc(n)) %>%  
 slice\_head(n = num\_top\_bigrams\_per\_community) %>%  
 ungroup()  
  
all\_communities\_data$community <- as.factor(all\_communities\_data$community)  
  
to\_add <- data.frame(  
 bigram = NA,  
 n = NA,  
 community = rep(levels(all\_communities\_data$community), each = empty\_bar\_count)  
)  
  
plot\_data <- rbind(all\_communities\_data, to\_add) %>%  
 arrange(community)  
  
plot\_data$id <- seq\_len(nrow(plot\_data)) # Keep ID as numeric here  
  
# --- 2. Prepare Label Data ---  
label\_data <- plot\_data  
number\_of\_bar <- nrow(label\_data)  
label\_data$angle <- 90 - 360 \* (label\_data$id - 0.5) / number\_of\_bar  
label\_data$hjust <- ifelse(label\_data$angle < -90, 1, 0)  
label\_data$angle <- ifelse(label\_data$angle < -90, label\_data$angle + 180, label\_data$angle)  
  
# --- 3. Prepare Data for Baselines (Community Dividers) ---  
base\_data <- plot\_data %>%  
 group\_by(community) %>%  
 summarize(  
 start = min(id, na.rm = TRUE), # Keep as numeric  
 end = max(id, na.rm = TRUE) - empty\_bar\_count # Keep as numeric  
 ) %>%  
 rowwise() %>%  
 mutate(  
 title\_position = mean(c(start, end))  
 ) %>%  
 ungroup()  
  
# --- 4. Prepare Data for Grid Lines (Optional: Value Scales) ---  
max\_n\_value <- max(plot\_data$n, na.rm = TRUE)  
grid\_lines\_values <- c(20, 40, 60, 80, 100)  
grid\_lines\_values <- grid\_lines\_values[grid\_lines\_values <= max\_n\_value]  
  
grid\_segments\_data <- plot\_data %>%  
 group\_by(community) %>%  
 summarize(  
 start\_id = min(id, na.rm = TRUE), # Keep as numeric  
 end\_id = max(id, na.rm = TRUE) - empty\_bar\_count # Keep as numeric  
 )  
  
grid\_data\_final <- tibble()  
for(val in grid\_lines\_values) {  
 temp\_data <- grid\_segments\_data %>%  
 mutate(y\_value = val)  
 grid\_data\_final <- bind\_rows(grid\_data\_final, temp\_data)  
}  
  
# --- Data for grid line LABELS ---  
grid\_label\_data <- data.frame(  
 x\_pos = max(plot\_data$id, na.rm = TRUE) + 2, # Fixed x position outside the plot  
 y\_pos = grid\_lines\_values,  
 label\_text = as.character(grid\_lines\_values)  
)  
  
# --- 5. Make the Unified Plot ---  
p <- ggplot(plot\_data, aes(x = id, y = n, fill = community)) + # <--- x = id (numeric)  
 # Add background grid lines for value (e.g., 20, 40, 60, 80)  
 geom\_segment(data = grid\_data\_final,  
 aes(x = start\_id - 0.5, y = y\_value, xend = end\_id + 0.5, yend = y\_value),  
 inherit.aes = FALSE,  
 color = "grey", alpha = 0.8, linewidth = 0.3) +  
  
 # Add text showing the value of each grid line at a fixed position  
 geom\_text(data = grid\_label\_data,  
 aes(x = x\_pos, y = y\_pos, label = label\_text),  
 inherit.aes = FALSE,  
 color = "grey", size = 3, angle = 0, fontface = "bold", hjust = 0) +  
  
 # Bars for the bigrams (main plot elements)  
 geom\_bar(stat = "identity", alpha = 0.8, color = "white", linewidth = 0.1,  
 width = 1.4) + # <--- Add width=1 to remove space between bars if id is numeric  
  
 # Set limits for the y-axis, providing space for labels  
 ylim(-max\_n\_value \* 0.7, max\_n\_value \* 1.2) +  
  
 theme\_minimal() +  
 theme(  
 legend.position = "none",  
 axis.text = element\_blank(),  
 axis.title = element\_blank(),  
 panel.grid = element\_blank(),  
 plot.margin = unit(c(1.5, 1.5, 1.5, 1.5), "cm") # Top, Right, Bottom, Left margins  
 ) +  
 coord\_polar(start = 0) +  
  
 # Add bigram labels  
 geom\_text(  
 data = label\_data,  
 aes(x = id, y = n + 10, label = bigram, hjust = hjust), # <--- x = id (numeric)  
 color = "black", fontface = "bold", alpha = 0.8, size = 2.8,  
 angle = label\_data$angle, inherit.aes = FALSE  
 ) +  
  
 # Add base lines for each community segment  
 geom\_segment(  
 data = base\_data,  
 aes(x = start - 0.5, y = -10, xend = end + 0.5, yend = -10),  
 colour = "black", alpha = 0.8, linewidth = 0.6, inherit.aes = FALSE  
 ) +  
  
 # Add community group labels  
 geom\_text(  
 data = base\_data,  
 aes(x = title\_position, y = -40, label = paste("Comm.", community)),  
 colour = "black", alpha = 0.9, size = 2, fontface = "bold", inherit.aes = FALSE  
 )+  
 # --- Add the Title ---  
 labs(  
 title = "Circular Bar Chart by Community",  
 subtitle = "Frequencies of key bigrams within each community", # Updated subtitle  
 caption = paste0("AT | Generated: ", Sys.Date())  
 ) +  
 # Apply the Set2 Brewer palette  
 scale\_fill\_brewer(palette = "Set2") +  
 # --- Customize title appearance ---  
 theme(  
 plot.title = element\_text(hjust = 0.5, size = 16, face = "bold", margin = margin(b = 10)),  
 plot.subtitle = element\_text(hjust = 0.5, size = 12, margin = margin(b = 10)),  
 plot.caption = element\_text(hjust = 1, size = 7, color = "grey50")  
 )  
  
print(p)

### 7.5.5 Community Group Membership- All

The topic area was gathered from the bigram wordclouds and circular bar chart. The Group Name was created based on knowledge from the Members in the group and the topic area. These were the information from the 6 segmented groups:

## The Table

Community Group Membership Summary

| Group Number | Topic Area | Members | Group Name |
| --- | --- | --- | --- |
| 1 | green guardians, city council, paackland harbor, green guardians, nemo reef, water quality | Defender, EcoVigil, Green Guardians, Horizon, Liam Thorne, Northern Light, Oceanus City Council, Paackland Harbor, Port Security, Reef Guardian, Seawatch, Sentinel, The Lookout | Conservationist Group |
| 2 | nemo reef, miesel shipping, permit, cr 7844, delta 3 | City Officials, Davis, Elise, Nadia Conti, Neptune, Remora, Rodriguez, Sailor Shifts Team, Small Fry, V. Miesel Shipping | Sailor Shift |
| 3 | nemo reef, himark harbor, samantha blake, radio silence | Haacklee Harbor, Himark Harbor, Knowles, Mako, Marlin, Osprey, Serenity | Maritime |
| 4 | money, 0500 meeting, nemo reef, intern reporting, conservation activity, financial projections | Boss, Glitters Team, Mrs. Money, Samantha Blake, The Accountant, The Intern, The Middleman | Suspicious Characters |
| 5 | sam, kelly, loading equipment | Kelly, Sam | Sam & Kelly |
| 6 | miranda reporting, jensen, project poseidon, conservation vessels, nemo reef | Clepper Jensen, Miranda Jordan | Hacklee Herald |

## The Code

set.seed(1234)  
  
# 6. Create a tidy summary table of members per community  
grouped\_members <- g %>%  
 as\_tibble() %>%  
 select(name, community) %>%  
 group\_by(community) %>%  
 summarise(  
 Members = paste(sort(name), collapse = ", "),  
 .groups = "drop"  
 ) %>%  
 mutate(  
 `Group Number` = community,  
 `Topic Area` = case\_when(  
 community == 1 ~ "green guardians, city council, paackland harbor, green guardians, nemo reef, water quality",  
 community == 2 ~ "nemo reef, miesel shipping, permit, cr 7844, delta 3",  
 community == 3 ~ "nemo reef, himark harbor, samantha blake, radio silence",  
 community == 4 ~ "money, 0500 meeting, nemo reef, intern reporting, conservation activity, financial projections",  
 community == 5 ~ "sam, kelly, loading equipment",  
 community == 6 ~ "miranda reporting, jensen, project poseidon, conservation vessels, nemo reef",  
 TRUE ~ "Other"  
 ),  
 `Group Name` = case\_when(  
 community == 1 ~ "Conservationist Group",  
 community == 2 ~ "Sailor Shift",  
 community == 3 ~ "Maritime",  
 community == 4 ~ "Suspicious Characters",  
 community == 5 ~ "Sam & Kelly",  
 community == 6 ~ "Hacklee Herald",  
 TRUE ~ "Miscellaneous"  
 )  
 ) %>%  
 select(`Group Number`, `Topic Area`, `Members`, `Group Name`)  
  
# Show the summary table in a clean format  
kable(grouped\_members, caption = "Community Group Membership Summary", align = "l")

#### **7.5.5.1 Findings:**

* We focused on the 11 nodes in the suspicious groups named:
  + Sailor Shift (Influential Nodes: Neptune, Remora, Nadia, V. Miesel Shipping, Davis);
  + Maritime (Influential Node: Mako);
  + Conservationist Group: (Influential Nodes: Reef Guardian, EcoVigil); and
  + Suspicious Characters (Influential Nodes: Mrs. Money, Boss, The Middleman).
* We held back on the slightly less influential nodes such as: Sam & Kelly, and Hacklee Herald which was where Clepper Jensen worked as a journalist.

### 7.5.6 Interactive Chord Diagram by Community

In 7.1, we previously created a static chord diagram and heatmap to identify node pairs with higher interactions, and the interactions across entity subtypes.

Here, the interactive chord diagram showed the correspondences among communities at every two hour intervals.

# Get community membership from graph object  
community\_df <- g %>%  
 as\_tibble() %>%  
 select(name, community)

#### 7.5.6.1 Findings:

We noticed some cross community direct and indirect communication occured mainly among influential nodes, suggesting collaboration. These are **some** sample linkages with arrows regardless of sent or received:

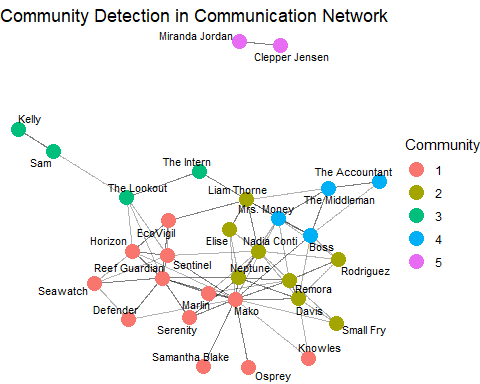
| Community X | Node Linkages (Community X -> Community X -> Community Y) |
| --- | --- |
| Suspicious Characters | Mrs. Money -> Intern -> The Lookout  Liam -> Paackland Harbor -> The Middleman  Glitters Team -> Boss -> Mako  Glitters Team -> Samantha Blake -> Sailor Shifts Team |
| Sailor Shift | Neptune -> Elise -> Mako  Neptune -> Davis -> Mako  Remora -> Neptune -> Boss  Rodriguez -> Remora -> Mako  Remora -> Small Fry -> Mako  Davis -> Remora -> Paackland Harbor  V. Miesel Shipping -> Neptune -> Mako |
| Sam & Kelly | Kelly -> Sam - > The Lookout |
| Maritime | Mako -> Himark Harbor -> Oceanus City Council |
| Hacklee Herald | N/A (Only Direct Community X to X communications) |
| Conservationist Group | Reef Guardian -> Oceanus City Council -> Nadia  Reef Guardian -> Paackland Harbor -> Mako  Oceanus City Council -> Liam -> Nadia |

We also noticed that at times, certain individuals sent messages but there were no response back. This could possibly be due to the pseudonyms being used to send or reply to the same content. For instance, there was a message from Davis to Rodriguez on 14 Oct around 1200-1400 but there was no response by Rodriguez. By looking at the content field, we then found out that he was Small Fry due to the responses he provided to Davis which was originally addressed to Rodriguez.

### 7.5.7 Community Detection- People & Vessels

Since our nodes (apart from V. Miesel Shipping) were mostly of people and vessels, in order for us to drill even deeper into people and vessels, we recreated the process for only people and vessels.

## Community Detection



## The Code

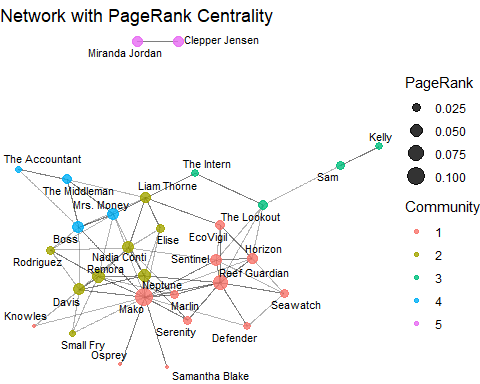
set.seed(1234)  
  
# --- STEP 1: Prepare the edge list ---  
edge\_df\_pv <- person\_vessel\_df %>%  
 select(sender\_id, recipient\_id) %>%  
 filter(!is.na(sender\_id) & !is.na(recipient\_id)) %>%  
 rename(from = sender\_id, to = recipient\_id) %>%  
 distinct()  
  
# Optional: remove self-loops  
edge\_df\_pv <- edge\_df\_pv %>% filter(from != to)  
  
# --- STEP 2: Create the graph object (undirected) ---  
g\_pv <- tbl\_graph(edges = edge\_df\_pv, directed = FALSE)  
  
# --- STEP 3: Run Louvain community detection ---  
g\_pv <- g\_pv %>%  
 mutate(community = group\_louvain())  
  
# --- STEP 4: Visualize the graph ---  
ggraph(g\_pv, layout = "fr") +  
 geom\_edge\_link(alpha = 0.3) +  
 geom\_node\_point(aes(color = as.factor(community)), size = 5) +  
 geom\_node\_text(aes(label = name), repel = TRUE, size = 3) +  
 theme\_void() +  
 labs(title = "Community Detection in Communication Network",  
 color = "Community")

### 7.5.8 Centrality Measure- People & Vessels

## PageRank Centrality Algorithm

# A tibble: 10 × 5  
 name pagerank degree betweenness closeness  
 <chr> <dbl> <dbl> <dbl> <dbl>  
 1 Mako 0.105 26 146. 0.0244  
 2 Reef Guardian 0.0741 19 57.2 0.0208  
 3 Neptune 0.0536 14 16.2 0.02   
 4 Remora 0.0477 12 9.94 0.0182  
 5 Davis 0.0452 11 10.6 0.0182  
 6 Mrs. Money 0.0429 10 34.6 0.0192  
 7 Boss 0.0425 10 34.1 0.0185  
 8 Sentinel 0.0410 10 25.8 0.02   
 9 Nadia Conti 0.0402 10 15.0 0.0185  
10 Horizon 0.0376 9 11.6 0.0192

## The Network



## The Code

set.seed(1234)  
  
# --- STEP: Compute Centrality Measures ---  
g\_pv <- g\_pv %>%  
 mutate(  
 pagerank = centrality\_pagerank(),  
 degree = centrality\_degree(),  
 betweenness = centrality\_betweenness(),  
 closeness = centrality\_closeness()  
 )  
  
# Show top 10 nodes by PageRank  
g\_pv %>%  
 as\_tibble() %>%  
 select(name, pagerank, degree, betweenness, closeness) %>%  
 arrange(desc(pagerank)) %>%  
 head(10)  
  
# Visualize by Centrality  
ggraph(g\_pv, layout = "fr") +  
 geom\_edge\_link(alpha = 0.3) +  
 geom\_node\_point(aes(size = pagerank, color = as.factor(community)), alpha = 0.8) +  
 geom\_node\_text(aes(label = name), repel = TRUE, size = 3) +  
 theme\_void() +  
 labs(title = "Network with PageRank Centrality",  
 size = "PageRank", color = "Community")

#### **7.5.8.1 Findings:**

* There were 5 closely associated groups. Community 5 (Clepper and Miranda) appeared to be segmented from the central group, due to the non-involvement from the nature of their investigative work.
* From the graph, we extracted the 8 influential nodes to focus on:
  + Community 1: Mako
  + Community 2: Neptune, Remora, Nadia, Davis
  + Community 3: N/A as they were not very influential at global level
  + Community 4: Mrs. Money, Boss, The Middleman
  + Community 5: N/A as they were not very influential at global level

### 7.5.9 Wordclouds- Bigrams

We focused on bigrams here to get more contextual data from two instead of one word.

## The Wordclouds

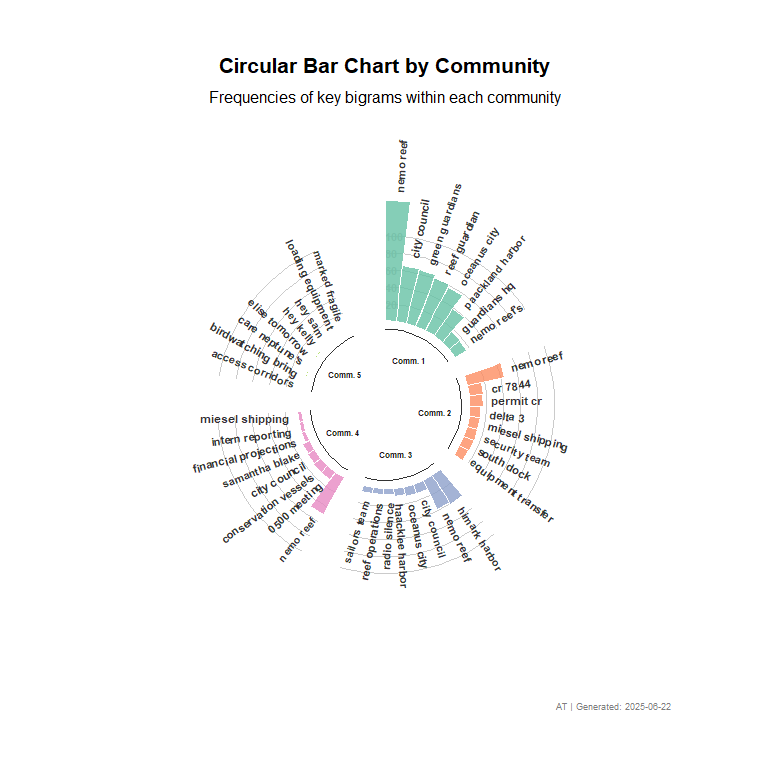


## The Code

# 5b. Plot word clouds per community  
valid\_communities <- unique(g\_pv %>% as\_tibble() %>% pull(community))  
  
par(mfrow = c(2, 3)) # Still allocate 6 slots, but you can adjust this  
for (i in sort(valid\_communities)) {  
 words <- bigrams %>% filter(community == i)  
   
 if (nrow(words) < 1) next  
   
 suppressWarnings({  
 set.seed(432) # Set seed for reproducibility  
 wordcloud(words = words$bigram, # <-- FIXED here  
 freq = words$n,  
 max.words = min(30, nrow(words)),  
 scale = c(3, 0.5),  
 colors = brewer.pal(8, "Dark2"),  
 random.order = FALSE)  
 })  
 mtext(paste("Community", i), side = 3, line = 1, adj = 0.5, cex = 1.5, col = "black")  
}

### 7.5.10 Circular barchart for Top Bigrams per Community

## The Circular Barchart



## The Code

set.seed(1234)  
  
set.seed(1234)  
  
valid\_communities <- g\_pv %>%  
 as\_tibble() %>%  
 distinct(community) %>%  
 pull(community)  
  
bigrams <- bigrams %>% filter(community %in% valid\_communities)  
  
# --- Configuration ---  
num\_top\_bigrams\_per\_community <- 8  
empty\_bar\_count <- 2 # gaps btw comm.  
#excluded\_community <- 5 # too little in community 5  
  
# --- 1. Prepare the Combined Dataset ---  
all\_communities\_data <- bigrams %>%  
# filter(community != excluded\_community) %>%  
 group\_by(community) %>%  
 arrange(desc(n)) %>%  
 slice\_head(n = num\_top\_bigrams\_per\_community) %>%  
 ungroup()  
  
all\_communities\_data$community <- as.factor(all\_communities\_data$community)  
  
to\_add <- data.frame(  
 bigram = NA,  
 n = NA,  
 community = rep(levels(all\_communities\_data$community), each = empty\_bar\_count)  
)  
  
plot\_data <- rbind(all\_communities\_data, to\_add) %>%  
 arrange(community)  
  
plot\_data$id <- seq\_len(nrow(plot\_data)) # Keep ID as numeric here  
  
# --- 2. Prepare Label Data ---  
label\_data <- plot\_data  
number\_of\_bar <- nrow(label\_data)  
label\_data$angle <- 90 - 360 \* (label\_data$id - 0.5) / number\_of\_bar  
label\_data$hjust <- ifelse(label\_data$angle < -90, 1, 0)  
label\_data$angle <- ifelse(label\_data$angle < -90, label\_data$angle + 180, label\_data$angle)  
  
# --- 3. Prepare Data for Baselines (Community Dividers) ---  
base\_data <- plot\_data %>%  
 group\_by(community) %>%  
 summarize(  
 start = min(id, na.rm = TRUE), # Keep as numeric  
 end = max(id, na.rm = TRUE) - empty\_bar\_count # Keep as numeric  
 ) %>%  
 rowwise() %>%  
 mutate(  
 title\_position = mean(c(start, end))  
 ) %>%  
 ungroup()  
  
# --- 4. Prepare Data for Grid Lines (Optional: Value Scales) ---  
max\_n\_value <- max(plot\_data$n, na.rm = TRUE)  
grid\_lines\_values <- c(20, 40, 60, 80, 100)  
grid\_lines\_values <- grid\_lines\_values[grid\_lines\_values <= max\_n\_value]  
  
grid\_segments\_data <- plot\_data %>%  
 group\_by(community) %>%  
 summarize(  
 start\_id = min(id, na.rm = TRUE), # Keep as numeric  
 end\_id = max(id, na.rm = TRUE) - empty\_bar\_count # Keep as numeric  
 )  
  
grid\_data\_final <- tibble()  
for(val in grid\_lines\_values) {  
 temp\_data <- grid\_segments\_data %>%  
 mutate(y\_value = val)  
 grid\_data\_final <- bind\_rows(grid\_data\_final, temp\_data)  
}  
  
# --- Data for grid line LABELS ---  
grid\_label\_data <- data.frame(  
 x\_pos = max(plot\_data$id, na.rm = TRUE) + 2, # Fixed x position outside the plot  
 y\_pos = grid\_lines\_values,  
 label\_text = as.character(grid\_lines\_values)  
)  
  
# --- 5. Make the Unified Plot ---  
p <- ggplot(plot\_data, aes(x = id, y = n, fill = community)) + # <--- x = id (numeric)  
 # Add background grid lines for value (e.g., 20, 40, 60, 80)  
 geom\_segment(data = grid\_data\_final,  
 aes(x = start\_id - 0.5, y = y\_value, xend = end\_id + 0.5, yend = y\_value),  
 inherit.aes = FALSE,  
 color = "grey", alpha = 0.8, linewidth = 0.3) +  
  
 # Add text showing the value of each grid line at a fixed position  
 geom\_text(data = grid\_label\_data,  
 aes(x = x\_pos, y = y\_pos, label = label\_text),  
 inherit.aes = FALSE,  
 color = "grey", size = 3, angle = 0, fontface = "bold", hjust = 0) +  
  
 # Bars for the bigrams (main plot elements)  
 geom\_bar(stat = "identity", alpha = 0.8, color = "white", linewidth = 0.1,  
 width = 1) + # <--- Add width=1 to remove space between bars if id is numeric  
  
 # Set limits for the y-axis, providing space for labels  
 ylim(-max\_n\_value \* 0.7, max\_n\_value \* 1.2) +  
  
 theme\_minimal() +  
 theme(  
 legend.position = "none",  
 axis.text = element\_blank(),  
 axis.title = element\_blank(),  
 panel.grid = element\_blank(),  
 plot.margin = unit(c(1.5, 1.5, 1.5, 1.5), "cm") # Top, Right, Bottom, Left margins  
 ) +  
 coord\_polar(start = 0) +  
  
 # Add bigram labels  
 geom\_text(  
 data = label\_data,  
 aes(x = id, y = n + 10, label = bigram, hjust = hjust), # <--- x = id (numeric)  
 color = "black", fontface = "bold", alpha = 0.8, size = 2.8,  
 angle = label\_data$angle, inherit.aes = FALSE  
 ) +  
  
 # Add base lines for each community segment  
 geom\_segment(  
 data = base\_data,  
 aes(x = start - 0.5, y = -10, xend = end + 0.5, yend = -10),  
 colour = "black", alpha = 0.8, linewidth = 0.6, inherit.aes = FALSE  
 ) +  
  
 # Add community group labels  
 geom\_text(  
 data = base\_data,  
 aes(x = title\_position, y = -40, label = paste("Comm.", community)),  
 colour = "black", alpha = 0.9, size = 2, fontface = "bold", inherit.aes = FALSE  
 )+  
 # --- Add the Title ---  
 labs(  
 title = "Circular Bar Chart by Community",  
 subtitle = "Frequencies of key bigrams within each community", # Updated subtitle  
 caption = paste0("AT | Generated: ", Sys.Date())  
 ) +  
 # Apply the Set2 Brewer palette  
 scale\_fill\_brewer(palette = "Set2") +  
 # --- Customize title appearance ---  
 theme(  
 plot.title = element\_text(hjust = 0.5, size = 16, face = "bold", margin = margin(b = 10)),  
 plot.subtitle = element\_text(hjust = 0.5, size = 12, margin = margin(b = 10)),  
 plot.caption = element\_text(hjust = 1, size = 7, color = "grey50")  
 )  
  
print(p)

### 7.5.11 Community Group Membership- People & Vessels

The topic area was gathered from the bigram wordclouds and circular bar chart. The Group Name was created based on knowledge from the Members in the group and the topic area. These are the information for the 5 segmented groups:

## The Table

Community Group Membership Summary

| Group Number | Topic Area | Members | Group Name |
| --- | --- | --- | --- |
| 1 | nemo reef, himark harbor, samantha blake, city council | Defender, EcoVigil, Horizon, Knowles, Mako, Marlin, Osprey, Reef Guardian, Samantha Blake, Seawatch, Sentinel, Serenity | Conservationist Group |
| 2 | south dock, equipment transfer, security team, nemo reef, delta3, cr 7844 | Davis, Elise, Liam Thorne, Nadia Conti, Neptune, Remora, Rodriguez, Small Fry | Permit |
| 3 | intern reporting, conservation vessels, nemo reef | Kelly, Sam, The Intern, The Lookout | Pseudonym |
| 4 | 10am tomorrow, 0500 tomorrow, funding channels, alternative funding | Boss, Mrs. Money, The Accountant, The Middleman | Suspicious |
| 5 | classification markings, project poseidon, clearance documents, harbor security | Clepper Jensen, Miranda Jordan | Hacklee Herald |

## The Code

set.seed(1234)  
# 6. Create a tidy summary table of members per community  
grouped\_members <- g\_pv %>%  
 as\_tibble() %>%  
 select(name, community) %>%  
 group\_by(community) %>%  
 summarise(  
 Members = paste(sort(name), collapse = ", "),  
 .groups = "drop"  
 ) %>%  
 mutate(  
 `Group Number` = community,  
 `Topic Area` = case\_when(  
 community == 1 ~ "nemo reef, himark harbor, samantha blake, city council",  
 community == 2 ~ "south dock, equipment transfer, security team, nemo reef, delta3, cr 7844",  
 community == 3 ~ "intern reporting, conservation vessels, nemo reef",  
 community == 4 ~ "10am tomorrow, 0500 tomorrow, funding channels, alternative funding",  
 community == 5 ~ "classification markings, project poseidon, clearance documents, harbor security",  
 TRUE ~ "Other"  
 ),  
 `Group Name` = case\_when(  
 community == 1 ~ "Conservationist Group",  
 community == 2 ~ "Permit",  
 community == 3 ~ "Pseudonym",  
 community == 4 ~ "Suspicious",  
 community == 5 ~ "Hacklee Herald",  
 TRUE ~ "Miscellaneous"  
 )  
 ) %>%  
 select(`Group Number`, `Topic Area`, `Members`, `Group Name`)  
  
# Show the summary table in a clean format  
kable(grouped\_members, caption = "Community Group Membership Summary", align = "l")

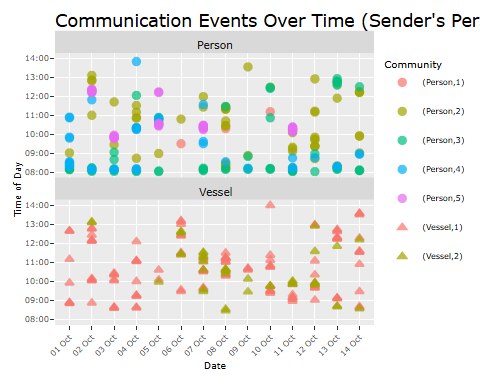
#### **7.5.11.1 Findings:**

* Movements and changes in membership since section 7.5.6:
  + Mako moved from Maritime to Conservationist.
  + Samantha Blake moved from the Suspicious Characters to Conservationist.
  + Sam and Kelly moved from their own Community to Pseudonym.
  + The Lookout moved from Conservationist Group to Pseudonym.
  + The Intern moved from Suspicious Characters to Pseudonym.
  + Sailor Shift community renamed to Permit
* From 11 nodes in section 7.5.6.1, we further narrowed down on the 8 nodes in the suspicious groups named:
  + Permit (Influential Nodes: Neptune, Remora, Nadia, Davis);
  + Conservationist Group (Influential Node: Mako);
  + Suspicious (Influential Nodes: Mrs. Money, Boss, The Middleman).
* We held back on the slightly less influential nodes such as: Hacklee Herald which was where Clepper Jensen worked as a journalist.

### 7.5.12 Plotted Community Timeline- People & Vessels

We then decided to plot the timeline based on community colours to uncover patterns.

# --- FACTORING and DATETIME CLEANING ---  
person\_vessel\_df\_for\_plot <- person\_vessel\_df %>%  
 mutate(  
 timestamp = as.POSIXct(timestamp),  
 comm\_date = as.Date(timestamp),  
 comm\_time\_of\_day = hms::as\_hms(format(timestamp, "%H:%M:%S")),  
 sender\_sub\_type = factor(sender\_sub\_type, levels = c("Person", "Vessel")),  
 communicating\_pair\_sorted = paste(pmin(sender\_name, recipient\_name), pmax(sender\_name, recipient\_name), sep = " & ")  
 )  
  
# Get community membership from graph object  
community\_df <- g\_pv %>%  
 as\_tibble() %>%  
 select(name, community)  
  
# --- WRAPPING CONTENT AND TOOLTIP ---  
plot\_data1 <- person\_vessel\_df\_for\_plot %>%  
 mutate(  
 timestamp = as.POSIXct(timestamp),  
 date = as.Date(timestamp),  
 time = format(timestamp, "%H:%M:%S"),  
 wrapped\_content = str\_wrap(content, width = 50),  
 tooltip\_text = paste0(  
 "<b>Date:</b> ", date, "<br>",  
 "<b>Time:</b> ", time, "<br>",  
 "<b>From:</b> ", sender\_name, "<br>",  
 "<b>To:</b> ", recipient\_name, "<br>",  
 "<b>Event\_id:</b> ", event\_id, "<br><br>",  
 "<b>Content:</b><br>", wrapped\_content  
 )  
 )  
  
# Merge with plot\_data1 using sender\_id == name  
plot\_data1 <- plot\_data1 %>%  
 left\_join(community\_df, by = c("sender\_id" = "name"))  
  
# Plot  
p <- ggplot(plot\_data1, aes(x = comm\_date, y = comm\_time\_of\_day)) +  
 geom\_point(aes(  
 color = as.factor(community),  
 shape = sender\_sub\_type,  
 text = tooltip\_text  
 ), show.legend = c(color = TRUE, shape = FALSE),  
 size = 2, alpha = 0.7) +  
 scale\_shape\_manual(values = c("Person" = 16, "Vessel" = 17)) +  
 facet\_wrap(~ sender\_sub\_type, ncol = 1, scales = "fixed") +  
 scale\_y\_time(  
 limits = hms::as\_hms(c("08:00:00", "14:00:00")), # reversed to show time top-to-bottom  
 breaks = hms::as\_hms(c("08:00:00", "09:00:00", "10:00:00", "11:00:00", "12:00:00", "13:00:00", "14:00:00")),  
 labels = c("08:00", "09:00", "10:00", "11:00", "12:00", "13:00", "14:00")  
)+  
 scale\_x\_date(  
 date\_breaks = "1 day",  
 date\_labels = "%d %b"  
)+  
 labs(  
 title = "Communication Events Over Time (Sender's Perspective)",  
 x = "Date",  
 y = "Time of Day",  
 color = "Community"  
 ) +  
 theme\_grey() +  
 theme(  
 axis.text.y = element\_text(size = 6),  
 axis.title.y = element\_text(size = 7),  
 axis.ticks.y = element\_line(),  
 axis.text.x = element\_text(size = 6, angle = 45, hjust = 1),  
 axis.title.x = element\_text(margin = margin(t = 10), size = 7),  
 panel.spacing = unit(0.5, "lines"), # Applies to both x and y spacing  
 strip.text = element\_text(size = 8, face = "bold"),  
 legend.position = "bottom",  
 legend.text = element\_text(size = 6),  
 legend.title = element\_blank()  
 )  
  
# --- Convert to interactive plot ---  
ggplotly(p, tooltip = "tooltip\_text")



### 7.5.13 Discussion/ Interpretation:

We mainly focused on the conversations by 8 influential nodes and some related nodes:

* **Conservation Group** (Comm.1): Samantha Blake informed Mako to stop operations on 8 and 10th Oct. Serenity is a private luxury yacht. Osprey was likely a tourism vessel looking for charter from Mako for their tourists.
* **Permit** (Comm. 2): Neptune, Remora, Nadia, and Davis were working on Nemo Reef operation. This referred to the Music Video Production for Sailor Shift on 14 Oct.
* **Pseudonym** (Comm. 3): Other than communicating among themselves, The Lookout appeared to have also externally corresponded with Sentinel, Reef Guardian and Horizon (conservation based topics), while The Intern also externally corresponded with Mrs. Money.
* **Suspicious** (Comm. 4): The Middleman had access to Council documents. Mrs. Money had funding from sources that would not flag out to regulators for her operations. Mrs. Money was investigating V. Miesel’s structures. On 5 Oct, Boss told Mrs. Money to disguise financial trails through tourism ventures and destroy evidence of Nemo Reef operations.
* **Hacklee Herald** (Comm. 5): Conversations between Clepper and his intern Miranda which ended on 11 Oct. Miranda mentioned an Oceanus City Council Member meeting with unmarked vessels at night.

# **8) Tackling Question 3**

# Question 3a)

## 8.1 Entities Breakdown

**Core Logic:**

* If two names appear as sender and recipient in the same message, they cannot belong to the same person — i.e., they’re not aliases of each other.
* If two names sent a message at the exact time, they cannot belong to the same person.

We created Alluvial Diagrams to chart: real\_identity → observed\_name → community

## Extraction of the list of Entity names

unique\_names <- sort(unique(c(other\_communications\_df$sender\_name,   
 other\_communications\_df$recipient\_name)))  
unique\_names

[1] "Boss" "City Officials" "Clepper Jensen"   
 [4] "Davis" "Defender" "EcoVigil"   
 [7] "Elise" "Glitters Team" "Green Guardians"   
[10] "Haacklee Harbor" "Himark Harbor" "Horizon"   
[13] "Kelly" "Knowles" "Liam Thorne"   
[16] "Mako" "Marlin" "Miranda Jordan"   
[19] "Mrs. Money" "Nadia Conti" "Neptune"   
[22] "Northern Light" "Oceanus City Council" "Osprey"   
[25] "Paackland Harbor" "Port Security" "Reef Guardian"   
[28] "Remora" "Rodriguez" "Sailor Shifts Team"   
[31] "Sam" "Samantha Blake" "Seawatch"   
[34] "Sentinel" "Serenity" "Small Fry"   
[37] "The Accountant" "The Intern" "The Lookout"   
[40] "The Middleman" "V. Miesel Shipping"

## Manual Mapping

name\_mapping <- tibble::tibble(  
 observed\_name = c(  
 "Boss", "City Officials", "Clepper Jensen", "Davis", "Defender", "EcoVigil",  
 "Elise", "Glitters Team", "Green Guardians", "Haacklee Harbor", "Himark Harbor", "Horizon",  
 "Kelly", "Knowles", "Liam Thorne", "Mako", "Marlin", "Miranda Jordan",  
 "Mrs. Money", "Nadia Conti", "Neptune", "Northern Light", "Oceanus City Council", "Osprey",  
 "Paackland Harbor", "Port Security", "Reef Guardian", "Remora", "Rodriguez", "Sailor Shift Team",  
 "Sam", "Samantha Blake", "Seawatch", "Sentinel", "Serenity", "Small Fry",  
 "The Accountant", "The Intern", "The Lookout", "The Middleman", "V. Miesel Shipping"  
 ),  
 real\_identity = c(  
 "Nadia Conti", "Oceanus City Council", "Clepper Jensen", "Captain Davis", "Sentinel", "EcoVigil",  
 "Elise", "Sailor Shift Team", "Green Guardians", "Harbor Authority","Harbor Authority", "Horizon",  
 "Kelly", "Knowles", "Liam Thorne", "Mako", "Small Fishing Vessel", "Miranda Jordan",  
 "Elise", "Nadia Conti", "Neptune", "Commercial Vessel", "Oceanus City Council", "Tourism Vessel",  
 "Harbor Authority", "Oceanus City Council", "Reef Guardian", "Remora", "Rodriguez", "Sailor Shift Team",  
 "Sam", "Samantha Blake", "Kelly", "Sentinel", "Private Luxury Yacht", "Rodriguez",  
 "Elise", "Sam", "Kelly", "Liam Thorne", "V. Miesel Shipping"  
 ),  
 community = c(  
 "V. Miesel Shipping", "City Council", "Haacklee Herald", "V. Miesel Shipping", "Local Conservationist Group", "Local Conservationist Group",  
 "V. Miesel Shipping", "Sailor Shift Team", "Local Conservationist Group", "Harbor Authority", "Harbor Authority", "Local Conservationist Group",  
 "Local Conservationist Group", "V. Miesel Shipping", "City Council", "V. Miesel Shipping", "Maritime", "Haacklee Herald",  
 "V. Miesel Shipping", "V. Miesel Shipping", "V. Miesel Shipping", "Maritime", "City Council", "Maritime",  
 "Harbor Authority", "City Council", "Local Conservationist Group", "V. Miesel Shipping", "V. Miesel Shipping", "Sailor Shift Team",  
 "V. Miesel Shipping", "Sailor Shift Team", "Local Conservationist Group", "Local Conservationist Group", "Maritime", "V. Miesel Shipping",  
 "V. Miesel Shipping", "V. Miesel Shipping", "Local Conservationist Group", "City Council", "V. Miesel Shipping"  
 )  
)  
  
multi\_members <- tibble::tibble(  
 observed\_name = c(  
 # City Council members (additional to existing)  
 "Commissioner Blake", "Commissioner Torres", "Council Knowles", "The Middleman", "Jensen from City Council", "Liam Thorne",  
  
 # Sailor Shift Team (may already exist, but we ensure all)  
 "Boss", "Council Knowles", "Davis", "Glitters Team", "Liam Thorne", "Mako", "Mrs. Money", "Nadia Conti", "Neptune",  
 "Remora", "Rodriguez", "Sam", "Samantha Blake", "Small Fry", "The Accountant", "The Intern", "The Middleman", "Elise",  
  
 # Influential Families  
 "Council Knowles", "V. Miesel Shipping",  
  
 # Conservationist Group  
 "Defender", "EcoVigil", "Green Guardians", "Horizon", "Kelly", "Reef Guardian", "Seawatch", "Sentinel", "The Lookout"  
 ),  
 real\_identity = c(  
 "Commissioner Blake", "Commissioner Torres", "Council Knowles", "Liam Thorne", "Clepper Jensen", "Liam Thorne",  
  
 "Nadia Conti", "Council Knowles", "Captain Davis", "Sailor Shift Team", "Liam Thorne", "Mako", "Elise", "Nadia Conti", "Neptune",  
 "Remora", "Rodriguez", "Sam", "Samantha Blake", "Rodriguez", "Elise", "Sam", "Liam Thorne", "Elise",  
  
 "Council Knowles", "V. Miesel Shipping",  
  
 "Sentinel", "EcoVigil", "Green Guardians", "Horizon", "Kelly", "Reef Guardian", "The Lookout", "Sentinel", "Kelly"  
 ),  
 community = c(  
 rep("City Council", 6),  
 rep("Sailor Shift Team", 18),  
 rep("Influential Families", 2),  
 rep("Local Conservationist Group", 9)  
 )  
)

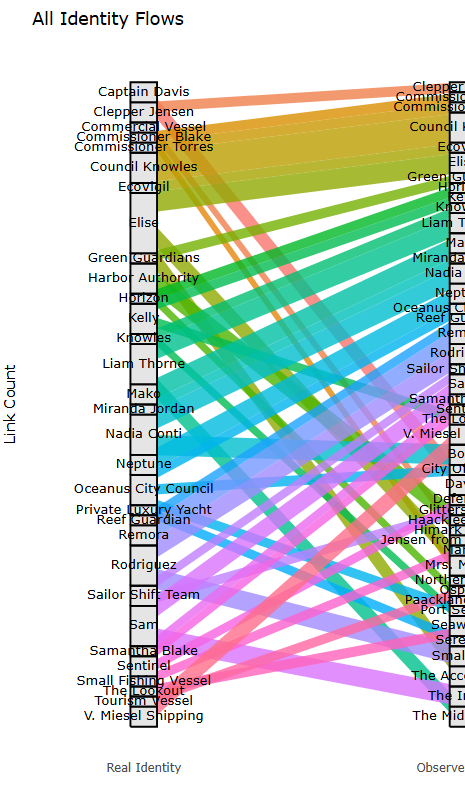
## Entities- Interactive Alluvial Diagrams

Hover to see details for individual identities.

Real Identity from analysis -> Observed Name from data -> Community

This will probably be a drop down for each name in Shiny.

# --- Libraries ---  
library(dplyr)  
library(ggplot2)  
library(ggalluvial)  
library(stringr)  
library(plotly)  
  
# --- 1. Build Alluvial Dataset ---  
alluvial\_data <- name\_mapping %>%  
 bind\_rows(multi\_members) %>% # Combine the two data sources  
 distinct() %>% # Remove any duplicate rows after combining  
 filter(!is.na(real\_identity) & !is.na(community)) %>%  
 count(real\_identity, observed\_name, community, name = "value") %>%  
 mutate(tooltip = paste0( # Create the tooltip string  
 "Real Identity: ", real\_identity, "<br>",  
 "Observed Name: ", observed\_name, "<br>",  
 "Community: ", community, "<br>",  
 "Count: ", value  
 ))  
  
# --- 3. Create static ggplot object (original version) ---  
p <- ggplot(alluvial\_data,  
 aes(axis1 = real\_identity, axis2 = observed\_name, axis3 = community,  
 y = value,  
 text = tooltip # Map the custom tooltip string to the 'text' aesthetic  
 )) +  
 geom\_alluvium(aes(fill = real\_identity), width = 1/12, alpha = 0.8) +  
 geom\_stratum(width = 1/12, fill = "grey90", color = "black", aes(text = NULL)) + # No dual-role fill here  
 geom\_text(stat = "stratum", aes(label = after\_stat(stratum)), size = 3.5, hjust = 0) +  
 scale\_x\_discrete(  
 limits = c("Real Identity", "Observed Name", "Community"),  
 expand = c(.05, .25)  
 ) +  
 labs(  
 title = "All Identity Flows",  
 x = NULL, y = "Link Count"  
 ) +  
 theme\_minimal() +  
 theme(  
 legend.position = "none", # No legend for fill here  
 panel.grid = element\_blank(),  
 axis.text.y = element\_blank(),  
 axis.ticks = element\_blank(),  
 plot.margin = margin(10, 80, 10, 10),  
 axis.text.x = element\_text(face = "bold")  
 )  
  
# --- 4. Convert to interactive Plotly object ---  
# 'tooltip = "text"' tells Plotly to use the content of the 'text' aesthetic  
interactive\_plot <- ggplotly(p, tooltip = "text", height = 800, width = 1000)  
interactive\_plot # This line will print the Plotly object in Quarto output



# Question 3b)

We created a graph with the character’s original name, pseudonyms, and supplemented with any background information we learnt of. For instance, we learnt that Davis was a captain, or that Serenity was a private luxury yacht.Here, it is easier to determine who is using which pseudonyms by clicking on the real identity drop down panel which will then only segregate to the individual’s Real Identity, Observed Name, and Community.

# Question 3c)

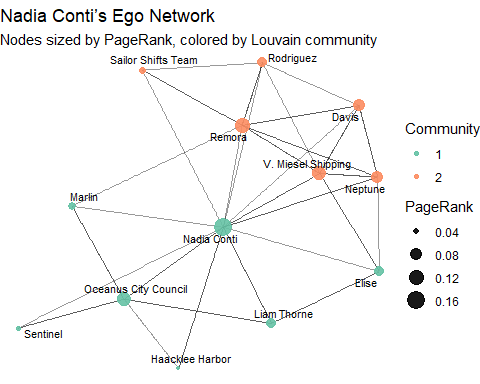
Understanding pseudonyms significantly reshapes our interpretation of the events in Oceanus. Without resolving aliases, the communication network appears fragmented — it may seem like dozens of separate individuals are involved. However, by mapping pseudonyms to real identities, we discover that a small number of actors are coordinating more activity than initially apparent. For example, a person using multiple pseudonyms may appear in many places at once — suggesting high influence or deception. This reveals orchestrated behavior, such as coordinated logistics, manipulation of event timelines, or masking involvement in controversial operations.

# **9) Tackling Question 4**

# Question 4a)

## 9.1 Nadia’s Ego Network- Louvain community

## Ego Network



## The Code

set.seed(1234)  
  
# Assume g\_full includes Nadia — not g from other\_communications\_df  
g\_igraph <- as.igraph(g)  
  
# Confirm Nadia exists  
if (!"Nadia Conti" %in% V(g\_igraph)$name) stop("Nadia Conti not found in the graph.")  
  
# Get ego subgraph  
nadia\_ego\_igraph <- make\_ego\_graph(g\_igraph, order = 1, nodes = which(V(g\_igraph)$name == "Nadia Conti"), mode = "all")[[1]]  
  
# Convert to tidygraph  
nadia\_ego\_graph <- as\_tbl\_graph(nadia\_ego\_igraph)  
  
# Convert to undirected for Louvain  
nadia\_ego\_graph <- nadia\_ego\_graph %>%   
 to\_undirected() %>%   
 activate(nodes) %>%  
 mutate(  
 community = group\_louvain(),  
 pagerank = centrality\_pagerank()  
 )  
  
# Plot Nadia's ego network  
ggraph(nadia\_ego\_graph, layout = "fr") +  
 geom\_edge\_link(alpha = 0.4) +  
 geom\_node\_point(aes(size = pagerank, color = as.factor(community)), alpha = 0.9) +  
 geom\_node\_text(aes(label = name), repel = TRUE, size = 3) +  
 scale\_color\_brewer(palette = "Set2") +  
 theme\_void() +  
 labs(  
 title = "Nadia Conti’s Ego Network",  
 subtitle = "Nodes sized by PageRank, colored by Louvain community",  
 color = "Community",  
 size = "PageRank"  
 )

* We wanted to find out if there were sub communities within Nadia’s direct network that worked closely together.
* The orange community were possibly involved in Sailor Shifts’s music video, while the green community were likely regarding ensuring compliance to authorities such as officials, the harbour and conservation team.
* Nadia, Elise, and Marlin were the orange nodes that directly linked to the green nodes.

## 9.2 Nadia’s Sent and Received Ego Networks- VizNetwork

### 9.2.1 Data Preparation

## Preparation- Nodes & Edges

[1] "Nadia Conti's ID: Nadia Conti"

[1] "Nadia Conti's Sub\_type: Person"

[1] "--- Nadia's Sent Communications ---"

|communication\_type |sender\_id |recipient\_id |recipient\_name |recipient\_sub\_type |event\_id |content |timestamp |sender\_name |sender\_sub\_type |  
|:------------------|:-----------|:--------------------|:--------------------|:------------------|:-----------------------|:------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|:-------------------|:-----------|:---------------|  
|sent |Nadia Conti |Haacklee Harbor |Haacklee Harbor |Location |Event\_Communication\_331 |Haacklee Harbor, this is Nadia Conti. I need to cancel the special access corridor arrangements for Nemo Reef immediately. Plans have changed due to unforeseen circumstances. Destroy all related documentation. I'll contact you when we're ready to proceed with alternative locations. |2040-10-05 09:45:00 |Nadia Conti |Person |  
|sent |Nadia Conti |Oceanus City Council |Oceanus City Council |Organization |Event\_Communication\_334 |This is Nadia Conti. My cancellation was due to scheduling conflicts with our tourism development initiatives. I wasn't aware of any permit approvals. I'll submit revised documentation for alternative sustainable tourism proposals next week. |2040-10-05 09:49:00 |Nadia Conti |Person |  
|sent |Nadia Conti |Liam Thorne |Liam Thorne |Person |Event\_Communication\_529 |Liam, Nadia here. Need your services urgently. Investigation brewing around Nemo Reef permits. Double your usual fee if you can ensure Harbor Master remains cooperative through next week. Meet at the usual place tomorrow, 10PM. |2040-10-08 08:18:00 |Nadia Conti |Person |  
|sent |Nadia Conti |Neptune |Neptune |Vessel |Event\_Communication\_536 |Neptune, this is Nadia. Need clarity on 'underwater foundation work' at Nemo Reef. This extends beyond our agreed scope. Meet me at the marina tomorrow at 6AM to discuss implications and additional resource requirements. |2040-10-08 08:25:00 |Nadia Conti |Person |  
|sent |Nadia Conti |Neptune |Neptune |Vessel |Event\_Communication\_538 |Neptune, Nadia here. Just confirming our 0600 meeting at the marina. I've reviewed the modified equipment specs with The Accountant. Please bring detailed timeline for foundation work and cost implications. We need to stay under radar. |2040-10-08 08:30:00 |Nadia Conti |Person |  
|sent |Nadia Conti |Marlin |Marlin |Vessel |Event\_Communication\_584 |Marlin, Nadia here. I understand you're inquiring about eastern shoal routes. Those are temporary diversions due to equipment transport needs. I'll have Davis provide the necessary documentation tonight. Nothing to be concerned about. |2040-10-08 11:23:00 |Nadia Conti |Person |  
|sent |Nadia Conti |Liam Thorne |Liam Thorne |Person |Event\_Communication\_795 |Liam, Nadia here. Redirect all remaining operations from southwest immediately. Move equipment to our secondary location. I'll handle EcoVigil through proper channels. Meet me at the usual place at 2100 hours with updated documentation. |2040-10-12 08:44:00 |Nadia Conti |Person |  
|sent |Nadia Conti |V. Miesel Shipping |V. Miesel Shipping |Organization |Event\_Communication\_847 |This is Nadia. Documentation for permit #CR-7844 is complete. Meeting The Middleman at 2100 to handle final details. Recommend accelerating timeline due to EcoVigil's ROV approval. Shifting operations from southwest immediately. Will update after meeting. |2040-10-12 11:19:00 |Nadia Conti |Person |

[1] "--- Nadia's Received Communications ---"

|communication\_type |sender\_id |sender\_name |sender\_sub\_type |recipient\_id |event\_id |content |timestamp |recipient\_name |recipient\_sub\_type |  
|:------------------|:--------------------|:--------------------|:---------------|:------------|:-----------------------|:--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|:-------------------|:--------------|:------------------|  
|received |Haacklee Harbor |Haacklee Harbor |Location |Nadia Conti |Event\_Communication\_330 |Haacklee Harbor to Nadia Conti. Following your visit yesterday regarding the Nemo Reef event logistics, we've prepared the necessary documentation. Harbor staff is ready to facilitate the special access corridor arrangements as discussed. Please confirm timeline for implementation. |2040-10-05 09:44:00 |Nadia Conti |Person |  
|received |Oceanus City Council |Oceanus City Council |Organization |Nadia Conti |Event\_Communication\_333 |Ms. Conti, this is Oceanus City Council. We need clarification regarding your canceled Nemo Reef event arrangements at Haacklee Harbor. Please explain your documentation destruction request immediately. This relates to our newly expedited permit approvals. |2040-10-05 09:48:00 |Nadia Conti |Person |  
|received |Sailor Shifts Team |Sailor Shifts Team |Organization |Nadia Conti |Event\_Communication\_520 |Hi Nadia, this is the Sailor Shifts Team. Received your message about permit assistance - thank you! We urgently need to discuss tomorrow's staffing requirements. Can you confirm how many additional crew members we should bring for the setup? |2040-10-07 11:57:00 |Nadia Conti |Person |  
|received |Davis |Davis |Person |Nadia Conti |Event\_Communication\_521 |Davis, Nadia here. Let's meet at 7PM at the marina office to review documentation. I've been working with alternative channels for permits. Bring all shipping manifests - we'll need to create a clean paper trail immediately. |2040-10-07 12:00:00 |Nadia Conti |Person |  
|received |Elise |Elise |Person |Nadia Conti |Event\_Communication\_528 |Nadia, Elise here. Meeting at Nemo Reef 0500 tomorrow to establish payment protocols. Sam uncovered V. Miesel shipping lanes overlapping with Mako by 40%. Neptune mentioned 'underwater foundation work' - outside our original scope. Need your assessment. |2040-10-08 08:15:00 |Nadia Conti |Person |  
|received |Liam Thorne |Liam Thorne |Person |Nadia Conti |Event\_Communication\_535 |Nadia, Liam here. Meeting confirmed for tomorrow at 10PM. I've redirected Harbor Master's attention and implemented new patrol schedules that work in our favor. Council suspects nothing about Nemo Reef. Bring payment as discussed. |2040-10-08 08:24:00 |Nadia Conti |Person |  
|received |Neptune |Neptune |Vessel |Nadia Conti |Event\_Communication\_537 |Neptune to Nadia. I'm aware of the foundation work concerns. We're delivering the additional heavy equipment today as requested. Will meet you at 0600 as planned to discuss resource adjustments and review modified equipment specifications that Elise has approved funding for. |2040-10-08 08:27:00 |Nadia Conti |Person |  
|received |Davis |Davis |Person |Nadia Conti |Event\_Communication\_582 |Nadia, Davis here. I'll be at the marina office at 7PM with all shipping manifests. Could you bring copies of permit #CR-7844? Marlin's asking about unusual vessel routes near eastern shoals - might need to address this. |2040-10-08 11:21:00 |Nadia Conti |Person |  
|received |Davis |Davis |Person |Nadia Conti |Event\_Communication\_585 |Davis, Marlin here again. Nadia mentioned you'd provide documentation about those eastern shoal diversions tonight. Just checking if that's still coming through. Need to understand these new patterns while my vessel's being repaired. |2040-10-08 11:26:00 |Nadia Conti |Person |  
|received |Elise |Elise |Person |Nadia Conti |Event\_Communication\_601 |Nadia, Elise here. Situation escalating. Permanent underwater construction confirmed at Nemo Reef. Sam reports concrete forms suggesting structures beyond equipment installation. Need urgent clarification on real scope and V. Miesel's involvement before tomorrow's meeting. Prepare contingencies. |2040-10-09 08:54:00 |Nadia Conti |Person |

[1] "--- Nadia's Full Communication Timeline (Combined) ---"

|communication\_type |sender\_id |recipient\_id |recipient\_name |recipient\_sub\_type |event\_id |content |timestamp |sender\_name |sender\_sub\_type |communicating\_pair\_sorted |  
|:------------------|:--------------------|:--------------------|:--------------------|:------------------|:-----------------------|:------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|:-------------------|:--------------------|:---------------|:--------------------------------|  
|received |Haacklee Harbor |Nadia Conti |Nadia Conti |Person |Event\_Communication\_330 |Haacklee Harbor to Nadia Conti. Following your visit yesterday regarding the Nemo Reef event logistics, we've prepared the necessary documentation. Harbor staff is ready to facilitate the special access corridor arrangements as discussed. Please confirm timeline for implementation. |2040-10-05 09:44:00 |Haacklee Harbor |Location |Haacklee Harbor\_Nadia Conti |  
|sent |Nadia Conti |Haacklee Harbor |Haacklee Harbor |Location |Event\_Communication\_331 |Haacklee Harbor, this is Nadia Conti. I need to cancel the special access corridor arrangements for Nemo Reef immediately. Plans have changed due to unforeseen circumstances. Destroy all related documentation. I'll contact you when we're ready to proceed with alternative locations. |2040-10-05 09:45:00 |Nadia Conti |Person |Haacklee Harbor\_Nadia Conti |  
|received |Oceanus City Council |Nadia Conti |Nadia Conti |Person |Event\_Communication\_333 |Ms. Conti, this is Oceanus City Council. We need clarification regarding your canceled Nemo Reef event arrangements at Haacklee Harbor. Please explain your documentation destruction request immediately. This relates to our newly expedited permit approvals. |2040-10-05 09:48:00 |Oceanus City Council |Organization |Nadia Conti\_Oceanus City Council |  
|sent |Nadia Conti |Oceanus City Council |Oceanus City Council |Organization |Event\_Communication\_334 |This is Nadia Conti. My cancellation was due to scheduling conflicts with our tourism development initiatives. I wasn't aware of any permit approvals. I'll submit revised documentation for alternative sustainable tourism proposals next week. |2040-10-05 09:49:00 |Nadia Conti |Person |Nadia Conti\_Oceanus City Council |  
|received |Sailor Shifts Team |Nadia Conti |Nadia Conti |Person |Event\_Communication\_520 |Hi Nadia, this is the Sailor Shifts Team. Received your message about permit assistance - thank you! We urgently need to discuss tomorrow's staffing requirements. Can you confirm how many additional crew members we should bring for the setup? |2040-10-07 11:57:00 |Sailor Shifts Team |Organization |Nadia Conti\_Sailor Shifts Team |  
|received |Davis |Nadia Conti |Nadia Conti |Person |Event\_Communication\_521 |Davis, Nadia here. Let's meet at 7PM at the marina office to review documentation. I've been working with alternative channels for permits. Bring all shipping manifests - we'll need to create a clean paper trail immediately. |2040-10-07 12:00:00 |Davis |Person |Davis\_Nadia Conti |  
|received |Elise |Nadia Conti |Nadia Conti |Person |Event\_Communication\_528 |Nadia, Elise here. Meeting at Nemo Reef 0500 tomorrow to establish payment protocols. Sam uncovered V. Miesel shipping lanes overlapping with Mako by 40%. Neptune mentioned 'underwater foundation work' - outside our original scope. Need your assessment. |2040-10-08 08:15:00 |Elise |Person |Elise\_Nadia Conti |  
|sent |Nadia Conti |Liam Thorne |Liam Thorne |Person |Event\_Communication\_529 |Liam, Nadia here. Need your services urgently. Investigation brewing around Nemo Reef permits. Double your usual fee if you can ensure Harbor Master remains cooperative through next week. Meet at the usual place tomorrow, 10PM. |2040-10-08 08:18:00 |Nadia Conti |Person |Liam Thorne\_Nadia Conti |  
|received |Liam Thorne |Nadia Conti |Nadia Conti |Person |Event\_Communication\_535 |Nadia, Liam here. Meeting confirmed for tomorrow at 10PM. I've redirected Harbor Master's attention and implemented new patrol schedules that work in our favor. Council suspects nothing about Nemo Reef. Bring payment as discussed. |2040-10-08 08:24:00 |Liam Thorne |Person |Liam Thorne\_Nadia Conti |  
|sent |Nadia Conti |Neptune |Neptune |Vessel |Event\_Communication\_536 |Neptune, this is Nadia. Need clarity on 'underwater foundation work' at Nemo Reef. This extends beyond our agreed scope. Meet me at the marina tomorrow at 6AM to discuss implications and additional resource requirements. |2040-10-08 08:25:00 |Nadia Conti |Person |Nadia Conti\_Neptune |

[1] "--- Checking: Number of nodes and edges in Nadia's Ego Network Graph ---"

[1] "Nodes: 13 Edges: 17"

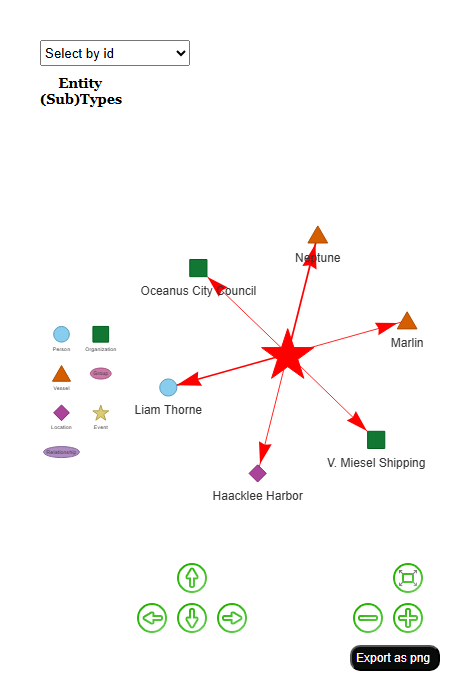
[1] "---------------------------------------------------------------------"

## The Code

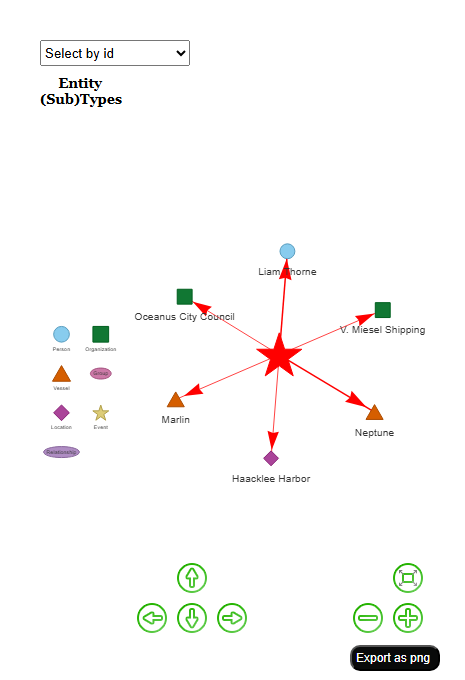
# --- 2. Clean and Prepare Nodes ---  
mc3\_nodes\_cleaned <- mc3\_nodes\_raw %>%  
 mutate(id = as.character(id)) %>%  
 filter(!is.na(id)) %>%  
 distinct(id, .keep\_all = TRUE) %>%  
 # Rename 'type' to 'supertype' to reduce confusion with communication type  
 rename(supertype = type) %>%  
 # Select only columns that are needed and are consistently present  
 select(id, name, sub\_type, content, timestamp) # Keep timestamp as character for now  
  
# --- 3. Clean and Prepare Edges ---  
# Rename 'type' in edges to 'edge\_type' to avoid conflict with node 'supertype'  
mc3\_edges\_cleaned <- mc3\_edges\_raw %>%  
 rename(from\_id = source,  
 to\_id = target,  
 edge\_type = type) %>% # Renamed 'type' to 'edge\_type'  
 mutate(across(c(from\_id, to\_id), as.character)) %>%  
 # Filter out any edges where from\_id or to\_id are not in cleaned nodes  
 filter(from\_id %in% mc3\_nodes\_cleaned$id,  
 to\_id %in% mc3\_nodes\_cleaned$id)  
  
# --- 4. Identify Nadia Conti's ID and Sub\_type ---  
nadia\_info <- mc3\_nodes\_cleaned %>%  
 filter(name == "Nadia Conti") %>%  
 select(id, sub\_type) # Corrected: Select sub\_type here, not supertype  
  
nadia\_id <- nadia\_info %>% pull(id)  
nadia\_sub\_type <- nadia\_info %>% pull(sub\_type) # New variable for Nadia's sub\_type  
  
  
if (length(nadia\_id) == 0) {  
 stop("Nadia Conti not found in the nodes data. Please check the 'name' column or the ID.")  
} else if (length(nadia\_id) > 1) {  
 warning("Multiple entries found for Nadia Conti. Using the first one.")  
 nadia\_id <- nadia\_id[1]  
 nadia\_sub\_type <- nadia\_sub\_type[1] # Ensure sub\_type is also taken for the first one  
}  
  
print(paste("Nadia Conti's ID:", nadia\_id))  
print(paste("Nadia Conti's Sub\_type:", nadia\_sub\_type))  
  
# --- 5. Extract Nadia's Sent Communications ---  
# Logic: Nadia (source) --sent--> Event\_Communication (target) --received--> Recipient (target)  
nadia\_sent\_communications <- mc3\_edges\_cleaned %>%  
 filter(from\_id == nadia\_id, edge\_type == "sent") %>%  
 # Join with nodes to get content and timestamp of the Event\_Communication node  
 left\_join(mc3\_nodes\_cleaned %>% select(id, content, timestamp),  
 by = c("to\_id" = "id")) %>%  
 rename(event\_id = to\_id, event\_content = content, event\_timestamp = timestamp) %>%  
 # Now, find the recipient of this communication event  
 left\_join(mc3\_edges\_cleaned %>%  
 filter(edge\_type == "received") %>%  
 select(event\_id\_match = from\_id, recipient\_id = to\_id),  
 by = c("event\_id" = "event\_id\_match")) %>%  
 # Join with nodes to get the recipient's name AND sub\_type  
 left\_join(mc3\_nodes\_cleaned %>% select(id, name, sub\_type), # Select sub\_type here  
 by = c("recipient\_id" = "id")) %>%  
 rename(recipient\_name = name, recipient\_sub\_type = sub\_type) %>% # Rename sub\_type  
 # Select and rename final columns for sent communications  
 select(  
 communication\_type = edge\_type, # This will be "sent"  
 sender\_id = from\_id,  
 recipient\_id,  
 recipient\_name,  
 recipient\_sub\_type, # Include in final select  
 event\_id,  
 content = event\_content,  
 timestamp = event\_timestamp # Timestamp is still character here  
 ) %>%  
 mutate(  
 sender\_name = !!nadia\_id, # Explicitly set sender\_name to Nadia's ID/name using !!  
 sender\_sub\_type = !!nadia\_sub\_type # Assign Nadia's sub\_type as a constant value using !!  
 )  
  
print("--- Nadia's Sent Communications ---")  
print(kable(head(nadia\_sent\_communications, 10), format = "markdown", align = "l"))  
  
  
# --- 6. Extract Nadia's Received Communications ---  
# Logic: Sender (source) --sent--> Event\_Communication (target) --received--> Nadia (target)  
nadia\_received\_communications <- mc3\_edges\_cleaned %>%  
 filter(to\_id == nadia\_id, edge\_type == "received") %>%  
 # The source of this edge is the Event\_Communication node  
 rename(event\_id = from\_id) %>%  
 # Join with nodes to get content and timestamp of the Event\_Communication node  
 left\_join(mc3\_nodes\_cleaned %>% select(id, content, timestamp),  
 by = c("event\_id" = "id")) %>%  
 rename(event\_content = content, event\_timestamp = timestamp) %>%  
 # Now, find the original sender of this communication event  
 left\_join(mc3\_edges\_cleaned %>%  
 filter(edge\_type == "sent") %>%  
 select(event\_id\_match = to\_id, sender\_id = from\_id),  
 by = c("event\_id" = "event\_id\_match")) %>%  
 # Join with nodes to get the sender's name AND sub\_type  
 left\_join(mc3\_nodes\_cleaned %>% select(id, name, sub\_type), # Select sub\_type here  
 by = c("sender\_id" = "id")) %>%  
 rename(sender\_name = name, sender\_sub\_type = sub\_type) %>% # Rename sub\_type  
 # Select and rename final columns for received communications  
 select(  
 communication\_type = edge\_type, # This will be "received"  
 sender\_id,  
 sender\_name,  
 sender\_sub\_type, # Include in final select  
 recipient\_id = to\_id,  
 event\_id,  
 content = event\_content,  
 timestamp = event\_timestamp  
 ) %>%  
 mutate(  
 recipient\_name = !!nadia\_id, # Explicitly set recipient\_name to Nadia's ID/name using !!  
 recipient\_sub\_type = !!nadia\_sub\_type # Assign Nadia's sub\_type as a constant value using !!  
 )  
  
print("--- Nadia's Received Communications ---")  
print(kable(head(nadia\_received\_communications, 10), format = "markdown", align = "l"))  
  
# --- 7. Combine Sent and Received Communications for Full Timeline ---  
nadia\_full\_communications\_timeline <- bind\_rows(  
 nadia\_sent\_communications,  
 nadia\_received\_communications  
) %>%  
 arrange(timestamp) %>%  
 # CRITICAL FIX: Explicitly convert timestamp to POSIXct after bind\_rows  
 # Use as.POSIXct with the observed format string  
 mutate(timestamp = as.POSIXct(timestamp, format = "%Y-%m-%d %H:%M:%S")) %>%  
 # NEW: Create a sorted communicating pair for consistent coloring  
 rowwise() %>% # Process row by row  
 mutate(communicating\_pair\_sorted = paste(sort(c(sender\_name, recipient\_name)), collapse = "\_")) %>%  
 ungroup() # Return to normal data frame operations  
  
print("--- Nadia's Full Communication Timeline (Combined) ---")  
print(kable(head(nadia\_full\_communications\_timeline, 10), format = "markdown", align = "l"))  
  
# --- 8. Prepare Data for Ego Network Visualization (Direct Person-to-Person/Entity) ---  
# Build nodes and edges directly from nadia\_full\_communications\_timeline,  
# focusing on direct sender-recipient connections.  
# Nodes for the ego network graph: Collect all unique sender and recipient IDs  
all\_ego\_person\_entity\_ids <- nadia\_full\_communications\_timeline %>%  
 select(id = sender\_id) %>%  
 bind\_rows(nadia\_full\_communications\_timeline %>% select(id = recipient\_id)) %>%  
 distinct(id) %>%  
 filter(!is.na(id)) # Ensure no NA IDs  
  
# Filter mc3\_nodes\_cleaned to get attributes for these person/entity nodes  
ego\_nodes\_for\_graph <- mc3\_nodes\_cleaned %>%  
 filter(id %in% all\_ego\_person\_entity\_ids$id) %>%  
 filter(!is.na(id)) %>% # Ensure no NA IDs in nodes for graph  
 # Add attributes for visualization  
 mutate(  
 is\_nadia = ifelse(id == nadia\_id, TRUE, FALSE),  
 # Create a new column for legend grouping that includes Nadia as a distinct category  
 node\_legend\_group = ifelse(is\_nadia, "Nadia Conti", sub\_type),  
 node\_size = ifelse(is\_nadia, 20, 18), # Further increased node sizes: Nadia 20, others 10  
 display\_name = ifelse(is\_nadia, "Nadia Conti", name) # Use actual name for others  
 ) %>%  
 filter(!is.na(sub\_type)) # Ensure nodes have a sub\_type for consistent plotting  
  
# Calculate communication counts for each node for hover text  
node\_comm\_summary <- nadia\_full\_communications\_timeline %>%  
 group\_by(id = sender\_id) %>%  
 summarise(sent\_count = n(), .groups = 'drop') %>%  
 full\_join(nadia\_full\_communications\_timeline %>%  
 group\_by(id = recipient\_id) %>%  
 summarise(received\_count = n(), .groups = 'drop'),  
 by = "id") %>%  
 mutate(  
 sent\_count = replace\_na(sent\_count, 0),  
 received\_count = replace\_na(received\_count, 0)  
 )  
  
# Join communication summary and create hover text for nodes  
ego\_nodes\_for\_graph <- ego\_nodes\_for\_graph %>%  
 left\_join(node\_comm\_summary, by = "id") %>%  
 mutate(  
 hover\_text = paste(  
 "Name:", display\_name, "<br>",  
 "Type:", node\_legend\_group, "<br>",  
 "Sent Communications:", sent\_count, "<br>",  
 "Received Communications:", received\_count  
 )  
 )  
  
# Edges for the ego network graph: Direct Sender -> Recipient edges  
# Aggregate to count occurrences for edge thickness  
ego\_edges\_for\_graph <- nadia\_full\_communications\_timeline %>%  
 # Group by sender, recipient, their names, and communication type to count interactions  
 group\_by(from = sender\_id, to = recipient\_id, sender\_name, recipient\_name, communication\_type) %>%  
 summarise(  
 count\_of\_comm = n(), # Number of times this specific communication happened  
 content\_sample = paste(head(content, 1), collapse = "; "), # Sample content  
 timestamp\_min = min(timestamp, na.rm = TRUE),  
 timestamp\_max = max(timestamp, na.rm = TRUE),  
 .groups = 'drop'  
 ) %>%  
 # Add the 'nadia\_role\_in\_comm' column for coloring based on Nadia's perspective  
 mutate(  
 nadia\_role\_in\_comm = case\_when(  
 from == nadia\_id & communication\_type == "sent" ~ "Nadia Sent",  
 to == nadia\_id & communication\_type == "received" ~ "Nadia Received",  
 TRUE ~ "Other Communication" # For communications not directly involving Nadia as sender/recipient  
 )  
 ) %>%  
 # Ensure 'from' and 'to' are character and non-NA  
 mutate(  
 from = as.character(from),  
 to = as.character(to)  
 ) %>%  
 filter(!is.na(from) & !is.na(to)) %>%  
 # Filter out edges where 'from' or 'to' IDs are NOT in the final ego\_nodes\_for\_graph  
 filter(from %in% ego\_nodes\_for\_graph$id, to %in% ego\_nodes\_for\_graph$id) %>%  
 # Create hover text for edges  
 mutate(  
 hover\_text = paste(  
 "From:", sender\_name, "<br>",  
 "To:", recipient\_name, "<br>",  
 "Type:", communication\_type, "<br>",  
 "Count:", count\_of\_comm, "<br>",  
 "First:", format(timestamp\_min, "%Y-%m-%d %H:%M:%S"), "<br>",  
 "Last:", format(timestamp\_max, "%Y-%m-%d %H:%M:%S"), "<br>",  
 "Content Sample:", content\_sample  
 )  
 )  
  
# Create the tbl\_graph object for the ego network  
nadia\_ego\_network\_graph <- tbl\_graph(nodes = ego\_nodes\_for\_graph, edges = ego\_edges\_for\_graph, directed = TRUE)  
  
print("--- Checking: Number of nodes and edges in Nadia's Ego Network Graph ---")  
print(paste("Nodes:", gorder(nadia\_ego\_network\_graph), "Edges:", gsize(nadia\_ego\_network\_graph)))  
print("---------------------------------------------------------------------")

### 9.2.2 Nadia’s Sent and Received Ego Networks

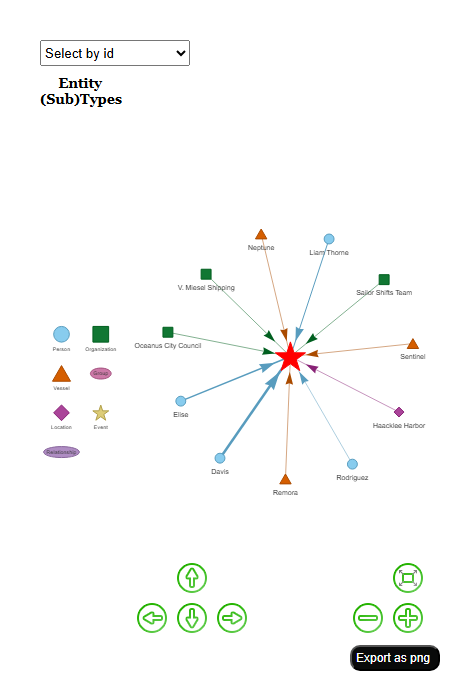
### 9.2.2.1 Sent Communications Network



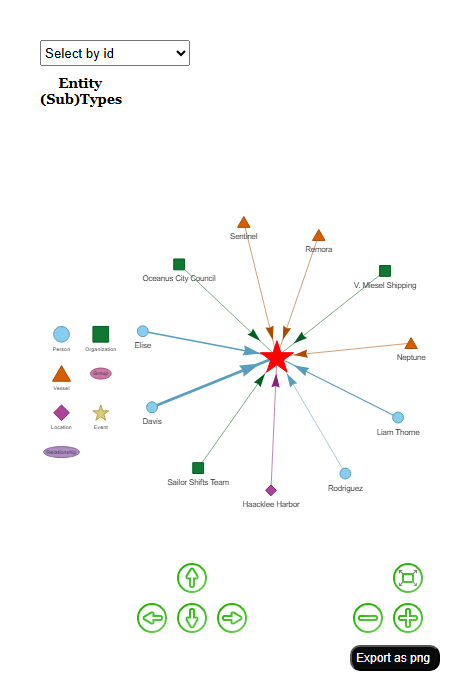
# Define custom colors for node types (reusing from previous code)  
node\_legend\_colors\_plot <- c(  
 "Person" = "#88CCEE", # Blue  
 "Vessel" = "#D55E00", # Orange  
 "Organization" = "#117733", # Green  
 "Location" = "#AA4499", # Purple  
 "Nadia Conti" = "red" # Special color for Nadia  
)  
  
# Define custom shapes for node subtypes  
node\_legend\_shapes\_plot <- c(  
 "Person" = "dot", # Circle for visNetwork  
 "Vessel" = "triangle",  
 "Organization" = "square",  
 "Location" = "diamond",  
 "Nadia Conti" = "star" # Star shape for Nadia in visNetwork  
)  
  
# Separate the edges into two sets: Nadia Sent and Nadia Received  
edges\_sent <- ego\_edges\_for\_graph %>%  
 filter(nadia\_role\_in\_comm == "Nadia Sent")  
  
edges\_received <- ego\_edges\_for\_graph %>%  
 filter(nadia\_role\_in\_comm == "Nadia Received")  
  
# Function to build visNetwork plot for a given edge set  
build\_visnetwork\_plot <- function(edge\_df, title\_label) {  
 # Prepare nodes from edge list  
 node\_ids <- unique(c(edge\_df$from, edge\_df$to))  
  
 nodes <- ego\_nodes\_for\_graph %>%  
 filter(id %in% node\_ids) %>%  
 mutate(  
 label = ifelse(display\_name == "Nadia Conti", "", display\_name),  
 title = paste0(  
 "<b>", display\_name, "</b><br>",  
 "Type: ", node\_legend\_group, "<br>",  
 "Sent: ", sent\_count, "<br>",  
 "Received: ", received\_count  
 ),  
 group = ifelse(display\_name == "Nadia Conti", "Nadia Conti", node\_legend\_group)  
 ) %>%  
 select(id, label, title, group, value = node\_size)  
  
 # Prepare edges with arrow and tooltip  
 edges <- edge\_df %>%  
 filter(!is.na(from) & !is.na(to)) %>%  
 mutate(  
 arrows = "to",  
 title = paste0(  
 "<b>From:</b> ", sender\_name, "<br>",  
 "<b>To:</b> ", recipient\_name, "<br>",  
 "<b>Type:</b> ", communication\_type, "<br>",  
 "<b>Count:</b> ", count\_of\_comm, "<br>",  
 "<b>First:</b> ", format(timestamp\_min, "%Y-%m-%d %H:%M:%S"), "<br>",  
 "<b>Last:</b> ", format(timestamp\_max, "%Y-%m-%d %H:%M:%S"), "<br>",  
 "<b>Content Sample:</b><br><div style='max-width:300px;white-space:normal;'>", content\_sample, "</div>"  
 )  
 ) %>%  
 select(from, to, arrows, title, width = count\_of\_comm)  
  
 # Create visNetwork  
 visNetwork(nodes, edges, width = "100%", height = "600px") %>%  
 visEdges(smooth = FALSE, arrows = list(to = list(enabled = TRUE, scaleFactor = 1.5))) %>%  
 visOptions(highlightNearest = TRUE, nodesIdSelection = TRUE) %>%  
 visIgraphLayout(layout = "layout\_with\_fr") %>%  
 visNodes(font = list(size = 14)) %>%  
 visGroups(groupname = "Person", color = node\_legend\_colors\_plot[["Person"]], shape = node\_legend\_shapes\_plot[["Person"]]) %>%  
 visGroups(groupname = "Vessel", color = node\_legend\_colors\_plot[["Vessel"]], shape = node\_legend\_shapes\_plot[["Vessel"]]) %>%  
 visGroups(groupname = "Organization", color = node\_legend\_colors\_plot[["Organization"]], shape = node\_legend\_shapes\_plot[["Organization"]]) %>%  
 visGroups(groupname = "Location", color = node\_legend\_colors\_plot[["Location"]], shape = node\_legend\_shapes\_plot[["Location"]]) %>%  
 visGroups(groupname = "Nadia Conti", color = node\_legend\_colors\_plot[["Nadia Conti"]], shape = node\_legend\_shapes\_plot[["Nadia Conti"]]) %>%  
 visLegend(  
 addNodes = legend\_df,  
 ncol = 2,  
 position = "left",  
 main = "Entity (Sub)Types",  
 useGroups = FALSE) %>%  
 visLayout(randomSeed = 123) %>%  
 visNetwork::visExport() %>%  
 visNetwork::visPhysics(enabled = TRUE) %>%  
 visNetwork::visInteraction(navigationButtons = TRUE) %>%  
 visNetwork::visEvents(stabilizationIterationsDone = "function () {this.setOptions({physics:false});}")  
}  
  
# Build and show visNetwork plots  
build\_visnetwork\_plot(edges\_sent, "Nadia Sent Ego Network")



### 9.2.2.2 Received Communications Network



build\_visnetwork\_plot(edges\_received, "Nadia Received Ego Network")



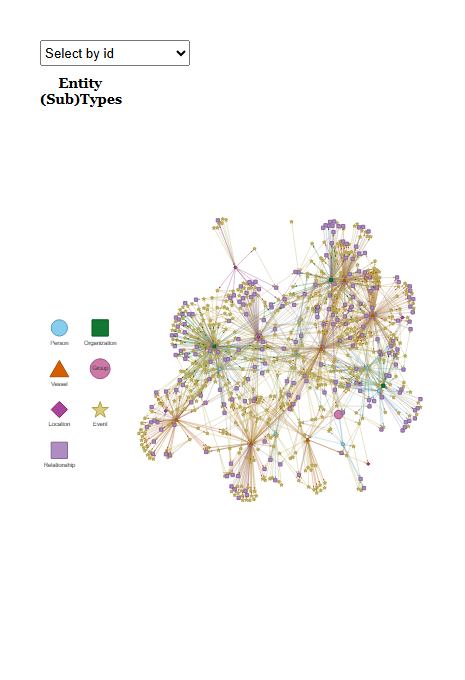
### **9.2.3 Findings**:

We then wanted to find more about the correspondences. The first graph illustrated Nadia’s sent communications and the second graph were her received communications. Hover over nodes to see the sent or received count from the perspective of the entity. Hover over arrowheads to see the number, direction, count, first and last correspondence time, type of sent or received, and content of one communication. Nadia sent only 8 but received 18 communications (possibly indicative of using a pseudonym):

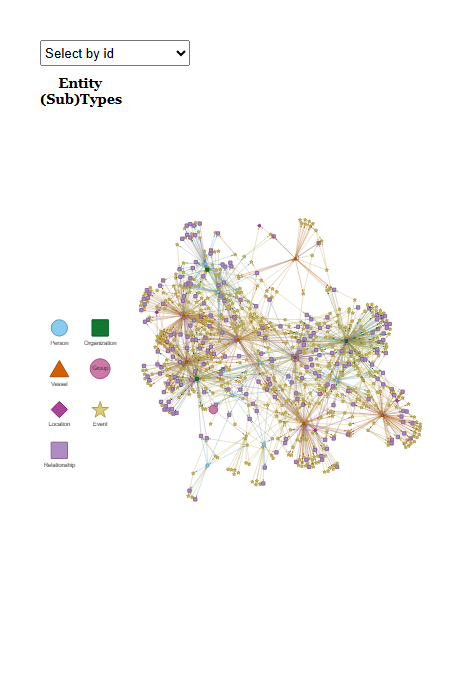
* Nodes involved: Nadia, Davis, Elise, Haacklee Habor, Liam, Marlin, Neptune, Oceanus City Council, Remora, Rodriguez, Sailor Shifts Team, Sentinel, V. Miesel Shipping.
* Sent more direct communications to Liam (2) and Neptune (2), relative to others.
* Received more from Davis (5), Elise (3), and Liam (2), relative to others.
* The next step would be to find out the suspicious characters/ activities.

## 9.3 Nadia’s Ego Network: 3-hops degree centrality

To uncover more details, we designed n-hops degree centrality here. We input 3 hops to hover over the nodes and look into the communications in a wider network. 3-hop exposes overlapping communites (Nadia communicating with someone who coordinates between groups). The “bridges” may show their value in 3-hops. It also increases the change of detecting complete or near complete cliques. We understand that should there be requirements for details, we could look into n-hops and more.



# --- Define event and relationship subtypes (optional, for filtering or labeling) ---  
event\_subtypes <- c(  
 "Communication", "Monitoring", "VesselMovement", "Assessment",  
 "Collaborate", "Endorsement", "TourActivity", "TransponderPing",  
 "Harbor Report", "Fishing", "Criticize"  
)  
  
relationship\_subtypes <- c(  
 "Coordinates", "AccessPermission", "Operates", "Colleagues",  
 "Suspicious", "Reports", "Jurisdiction", "Unfriendly", "Friends"  
)  
  
# --- Define node colors and shapes by type or key sub-type ---  
node\_legend\_colors\_plot <- c(  
 "Person" = "#88CCEE",  
 "Vessel" = "#D55E00",  
 "Organization" = "#117733",  
 "Location" = "#AA4499",  
 "Group" = "#CC79A7",  
 "Event" = "#DDCC77",  
 "Relationship" = "#AF8DC3"  
)  
  
node\_legend\_shapes\_plot <- c(  
 "Person" = "dot",  
 "Vessel" = "triangle",  
 "Organization" = "square",  
 "Location" = "diamond",  
 "Group" = "circle",  
 "Event" = "star",  
 "Relationship" = "square"  
)  
  
STYLES <- list(  
 node\_label\_dark = "black",  
 font\_family = "Roboto Condensed"  
)  
  
# ---- 2. Define function to extract ego subgraph ----  
extract\_ego\_subgraph <- function(center\_node = "Nadia Conti", hops = 3) {  
 # Construct igraph object  
 g <- graph\_from\_data\_frame(  
 d = mc3\_edges\_final %>% select(from = from\_id, to = to\_id),  
 vertices = mc3\_nodes\_final %>% select(id) %>% distinct() %>% rename(name = id),  
 directed = TRUE  
 )  
  
 # Extract ego network  
 ego\_graph <- make\_ego\_graph(g, order = hops, nodes = center\_node, mode = "all")[[1]]  
 node\_ids <- V(ego\_graph)$name  
 edge\_df <- as\_data\_frame(ego\_graph, what = "edges")  
  
 # Prepare node metadata for visNetwork  
 ego\_nodes <- mc3\_nodes\_final %>%  
 filter(id %in% node\_ids) %>%  
 mutate(  
 label = ifelse(is.na(name), id, name),  
 tooltip\_extra = case\_when(  
 type == "Event" & sub\_type == "Communication" ~ content,  
 type == "Event" & sub\_type == "Monitoring" ~ findings,  
 type == "Event" & sub\_type == "VesselMovement" ~ destination,  
 type == "Event" & sub\_type == "Assessment" ~ results,  
 type == "Relationship" & sub\_type == "Coordinates" ~ coordination\_type,  
 type == "Relationship" & sub\_type == "Operates" ~ operational\_role,  
 type == "Relationship" & sub\_type == "Jurisdiction" ~ jurisdiction\_type,  
 TRUE ~ NA\_character\_  
 ),  
 title = paste0(  
 "<b>", label, "</b><br>",  
 "Type: ", type, "<br>",  
 "Sub-type: ", sub\_type, "<br>",  
 ifelse(!is.na(tooltip\_extra), paste0("<br><b>Details:</b> ", tooltip\_extra), "")  
 ),  
 group = case\_when(  
 sub\_type %in% names(node\_legend\_colors\_plot) ~ sub\_type,  
 type %in% names(node\_legend\_colors\_plot) ~ type,  
 TRUE ~ "Other"  
 )  
 ) %>%  
 select(id, label, group, title) %>%  
 distinct(id, .keep\_all = TRUE)  
  
 list(nodes = ego\_nodes, edges = edge\_df)  
}  
  
# ---- 3. Extract Nadia Conti's 3-hop ego network ----  
ego\_data <- extract\_ego\_subgraph("Nadia Conti", hops = 3)  
nodes <- ego\_data$nodes  
edges <- ego\_data$edges  
  
# (Optional) Save output  
nadia\_ego\_nodes\_2hop <- nodes  
nadia\_ego\_edges\_2hop <- edges  
  
# ---- 4. Build visNetwork plot ----  
net <- visNetwork(nodes, edges, width = "100%", height = "600px") %>%  
 visEdges(arrows = list(to = list(enabled = TRUE, scaleFactor = 1.5))) %>%  
 visOptions(highlightNearest = TRUE, nodesIdSelection = TRUE) %>%  
 visIgraphLayout(layout = "layout\_with\_fr") %>%  
 visNodes(font = list(  
 size = 14,  
 color = STYLES$node\_label\_dark,  
 face = STYLES$font\_family,  
 vadjust = -15  
 ))  
  
# ---- 5. Apply color/shape for known groups ----  
used\_groups <- intersect(unique(nodes$group), names(node\_legend\_colors\_plot))  
  
for (group\_name in used\_groups) {  
 net <- net %>% visGroups(  
 groupname = group\_name,  
 color = node\_legend\_colors\_plot[[group\_name]],  
 shape = node\_legend\_shapes\_plot[[group\_name]]  
 )  
}  
  
# ---- 6. Add legend dynamically ----  
legend\_df <- tibble::tibble(  
 label = used\_groups,  
 shape = node\_legend\_shapes\_plot[used\_groups],  
 color = node\_legend\_colors\_plot[used\_groups]  
) %>%  
 distinct(label, .keep\_all = TRUE)  
  
net <- net %>% visLegend(  
 addNodes = legend\_df,  
 ncol = 2,  
 position = "left",  
 main = "Entity (Sub)Types",  
 useGroups = FALSE  
)  
  
# ---- 7. Render the network ----  
net



#### **9.3.1 Findings**:

* Nodes from 1-hop: Nadia, Davis, Elise, Haacklee Habor, Liam, Marlin, Neptune, Oceanus City Council, Remora, Rodriguez, Sailor Shifts Team, Sentinel, V. Miesel Shipping.
* Additional nodes from 3-hop: Conservation vessels, EcoVigil, Sam, The Accountant, Mako, Nemo Reef.
* We would then use these 3-hop nodes to drill in deeper to find suspicious relationships between nodes.

# Question 4b)

## **9.4 Summary of Nadia’s Actions**

### 9.4.1 Sequential Timeline in Table (3-hop)

## The Suspicious Entities

[1] "Sam" "Elise"   
 [3] "Liam Thorne" "Davis"   
 [5] "Rodriguez" "The Accountant"   
 [7] "V. Miesel Shipping" "EcoVigil"   
 [9] "Sentinel" "Remora"   
[11] "Event\_Communication\_222" "Event\_Communication\_270"   
[13] "Event\_Communication\_272" "Event\_Communication\_370"   
[15] "Event\_Communication\_485" "Event\_Communication\_535"   
[17] "Event\_Communication\_565" "Event\_Communication\_574"   
[19] "Event\_Communication\_601" "Event\_Communication\_612"   
[21] "Event\_Communication\_639" "Event\_Communication\_677"   
[23] "Event\_Communication\_724" "Event\_Communication\_726"   
[25] "Event\_Communication\_753" "Event\_Communication\_761"   
[27] "Event\_Communication\_833" "Event\_Communication\_916"   
[29] "Event\_Communication\_951" "Relationship\_Suspicious\_71"   
[31] "Relationship\_Suspicious\_74" "Relationship\_Suspicious\_106"  
[33] "Relationship\_Suspicious\_107" "Relationship\_Suspicious\_110"  
[35] "Relationship\_Suspicious\_112" "Relationship\_Suspicious\_123"  
[37] "Relationship\_Suspicious\_145" "Relationship\_Suspicious\_209"  
[39] "Relationship\_Suspicious\_217" "Relationship\_Suspicious\_293"  
[41] "Relationship\_Suspicious\_294" "Relationship\_Suspicious\_317"  
[43] "Relationship\_Suspicious\_341" "Relationship\_Suspicious\_360"  
[45] "Relationship\_Suspicious\_368" "Relationship\_Suspicious\_410"  
[47] "Relationship\_Suspicious\_436" "Relationship\_Suspicious\_457"  
[49] "Relationship\_Suspicious\_483" "Relationship\_Suspicious\_600"  
[51] "Relationship\_Suspicious\_577" "Relationship\_Suspicious\_113"  
[53] "Relationship\_Suspicious\_470" "Sailor Shifts Team"   
[55] "Oceanus City Council" "Mako"   
[57] "Neptune"

## The Code

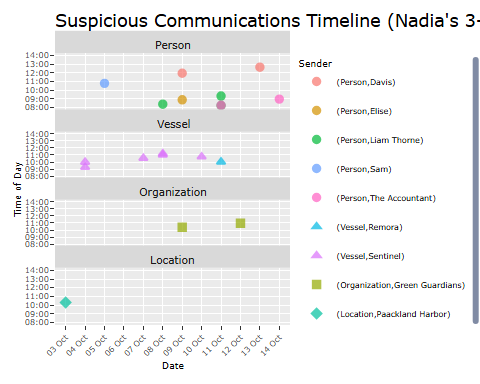
# Extract Nadia Conti's 3-hop ego network  
#| code-fold: true  
#| code-summary: "Show the code"  
  
# ---- Step 1: Create igraph from edge list ----  
g <- graph\_from\_data\_frame(  
 d = mc3\_edges\_final %>% select(from = from\_id, to = to\_id),  
 vertices = mc3\_nodes\_final %>% select(id) %>% distinct() %>% rename(name = id),  
 directed = TRUE  
)  
  
# ---- Step 2: Extract 3-hop ego network for Nadia Conti ----  
ego\_graph <- make\_ego\_graph(g, order = 3, nodes = "Nadia Conti", mode = "all")[[1]]  
node\_ids <- V(ego\_graph)$name  
edge\_df <- as\_data\_frame(ego\_graph, what = "edges")  
  
# ---- Step 3: Enrich ego nodes with labels and metadata ----  
nadia\_ego\_nodes\_3hop <- mc3\_nodes\_final %>%  
 filter(id %in% node\_ids) %>%  
 mutate(  
 label = ifelse(is.na(name), id, name),  
 tooltip\_extra = case\_when(  
 type == "Event" & sub\_type == "Communication" ~ content,  
 type == "Event" & sub\_type == "Monitoring" ~ findings,  
 type == "Event" & sub\_type == "VesselMovement" ~ destination,  
 type == "Event" & sub\_type == "Assessment" ~ results,  
 type == "Relationship" & sub\_type == "Coordinates" ~ coordination\_type,  
 type == "Relationship" & sub\_type == "Operates" ~ operational\_role,  
 type == "Relationship" & sub\_type == "Jurisdiction" ~ jurisdiction\_type,  
 TRUE ~ NA\_character\_  
 )  
 )  
  
# ---- Step 4: Enrich edges with sender/recipient names and types ----  
nadia\_ego\_edges\_3hop <- edge\_df %>%  
 left\_join(nadia\_ego\_nodes\_3hop %>% select(id, sender\_name = label, sender\_sub\_type = sub\_type), by = c("from" = "id")) %>%  
 left\_join(nadia\_ego\_nodes\_3hop %>% select(id, recipient\_name = label, recipient\_sub\_type = sub\_type), by = c("to" = "id"))  
  
# ---- Step 5: Create detailed communication dataframe ----  
nadia\_ego\_comm\_df\_3hop <- edge\_df %>%  
 left\_join(  
 nadia\_ego\_nodes\_3hop %>%  
 select(id, name, sub\_type),  
 by = c("from" = "id")  
 ) %>%  
 rename(  
 sender\_id = from,  
 sender\_name = name,  
 sender\_type = sub\_type  
 ) %>%  
 left\_join(  
 nadia\_ego\_nodes\_3hop %>%  
 select(id, name, sub\_type, content, findings, destination, results,  
 coordination\_type, operational\_role, jurisdiction\_type, timestamp),  
 by = c("to" = "id")  
 ) %>%  
 rename(  
 recipient\_id = to,  
 recipient\_name = name,  
 recipient\_type = sub\_type  
 ) %>%  
 mutate(  
 details = coalesce(content, findings, destination, results,  
 coordination\_type, operational\_role, jurisdiction\_type),  
 timestamp = suppressWarnings(ymd\_hms(timestamp))  
 ) %>%  
 select(  
 sender\_id, sender\_name, sender\_type,  
 recipient\_id, recipient\_name, recipient\_type,  
 timestamp, details  
 )  
  
# ---- Step 6: Filter for 'suspicious' in any relevant field ----  
suspicious\_links <- nadia\_ego\_comm\_df\_3hop %>%  
 filter(  
 grepl("suspicious", sender\_id, ignore.case = TRUE) |  
 grepl("suspicious", sender\_type, ignore.case = TRUE) |  
 grepl("suspicious", recipient\_id, ignore.case = TRUE) |  
 grepl("suspicious", recipient\_type, ignore.case = TRUE)  
 ) %>%  
 select(  
 sender\_id, sender\_name, sender\_type,  
 recipient\_id, recipient\_name, recipient\_type  
 )  
  
  
# ---- Step 7: View suspicious interactions ----  
unique\_ids <- unique(c(suspicious\_links$sender\_id, suspicious\_links$recipient\_id))  
print(unique\_ids)

* Elise, EcoVigil, Elise, Davis, Liam, Mako, Neptune, Sam, Remora, Rodriguez, Sailor Shifts Team, Sentinel, The Accountant, Oceanus City Council, V. Miesel Shipping possibly found something suspicious or were involved in something suspicious.

### 9.4.2 Plotted Timeline (3-hop)- Only suspicous communications

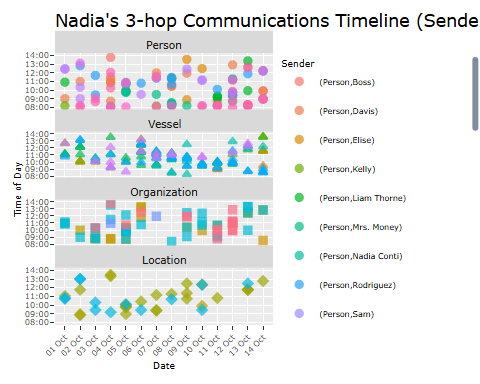
These are the suspicuous communications extracted from 9.4.1.

# ---- Step 1: Define fixed suspicious Event\_Communication IDs ----  
event\_ids\_to\_extract <- c(  
 "Event\_Communication\_222", "Event\_Communication\_270",  
 "Event\_Communication\_272", "Event\_Communication\_370",  
 "Event\_Communication\_485", "Event\_Communication\_535",  
 "Event\_Communication\_565", "Event\_Communication\_574",  
 "Event\_Communication\_601", "Event\_Communication\_612",  
 "Event\_Communication\_639", "Event\_Communication\_677",  
 "Event\_Communication\_724", "Event\_Communication\_726",  
 "Event\_Communication\_753", "Event\_Communication\_761",  
 "Event\_Communication\_833", "Event\_Communication\_916",  
 "Event\_Communication\_951"  
)  
  
# ---- Step 2: Match to master communication events ----  
matched\_comms\_df <- other\_communications\_df %>%  
 filter(event\_id %in% event\_ids\_to\_extract)  
  
# ---- Step 3: Prepare data for plotting ----  
matched\_comms\_df\_for\_plot <- matched\_comms\_df %>%  
 mutate(  
 timestamp = as.POSIXct(timestamp),  
 comm\_date = as.Date(timestamp),  
 comm\_time\_of\_day = hms::as\_hms(format(timestamp, "%H:%M:%S")),  
 sender\_sub\_type = factor(sender\_sub\_type, levels = c("Person", "Vessel", "Organization", "Location")),  
 communicating\_pair\_sorted = paste(pmin(sender\_name, recipient\_name), pmax(sender\_name, recipient\_name), sep = " & ")  
 )  
  
# ---- Step 4: Add tooltip labels ----  
plot\_data <- matched\_comms\_df\_for\_plot %>%  
 mutate(  
 wrapped\_content = str\_wrap(content, width = 50),  
 tooltip\_text = paste0(  
 "<b>Date:</b> ", comm\_date, "<br>",  
 "<b>Time:</b> ", format(comm\_time\_of\_day, "%H:%M:%S"), "<br>",  
 "<b>Event ID:</b> ", event\_id, "<br><br>",  
 "<b>From:</b> ", sender\_name, "<br>",  
 "<b>To:</b> ", recipient\_name, "<br><br>",  
 "<b>Content:</b><br>", wrapped\_content  
 )  
 )  
  
# ---- Step 5: Generate timeline plot ----  
p <- ggplot(plot\_data, aes(x = comm\_date, y = comm\_time\_of\_day)) +  
 geom\_point(aes(  
 color = sender\_name,  
 shape = sender\_sub\_type,  
 text = tooltip\_text  
 ), show.legend = c(color = TRUE, shape = FALSE), size = 2, alpha = 0.7) +  
  
 scale\_shape\_manual(values = c(  
 "Person" = 16,  
 "Vessel" = 17,  
 "Organization" = 15,  
 "Location" = 18  
 )) +  
  
 scale\_y\_time(  
 limits = hms::as\_hms(c("08:00:00", "14:00:00")),  
 breaks = hms::as\_hms(c("08:00:00", "09:00:00", "10:00:00", "11:00:00", "12:00:00", "13:00:00", "14:00:00")),  
 labels = c("08:00", "09:00", "10:00", "11:00", "12:00", "13:00", "14:00")  
 ) +  
  
 scale\_x\_date(  
 date\_breaks = "1 day",  
 date\_labels = "%d %b"  
 ) +  
  
 facet\_wrap(~ sender\_sub\_type, ncol = 1, scales = "fixed") +  
  
 labs(  
 title = "Suspicious Communications Timeline (Nadia's 3-hop Ego Network)",  
 x = "Date",  
 y = "Time of Day",  
 color = "Sender"  
 ) +  
  
 theme\_grey() +  
 theme(  
 axis.text.y = element\_text(size = 6),  
 axis.title.y = element\_text(size = 7),  
 axis.text.x = element\_text(size = 6, angle = 45, hjust = 1),  
 axis.title.x = element\_text(margin = margin(t = 10), size = 7),  
 panel.spacing = unit(0.5, "lines"),  
 strip.text = element\_text(size = 8, face = "bold"),  
 legend.position = "bottom",  
 legend.text = element\_text(size = 6),  
 legend.title = element\_text(size = 7)  
 )  
  
# ---- Step 6: Convert to interactive plot ----  
ggplotly(p, tooltip = "text")



### Plotted Timeline (3-hop)- Full communications

# ---- Step 1: Extract Communication Event IDs from Nadia's 3-hop ego network ----  
event\_ids\_to\_extract <- nadia\_ego\_comm\_df\_3hop %>%  
 filter(str\_starts(sender\_id, "Event\_Communication\_")) %>%  
 pull(sender\_id) %>%  
 unique()  
  
# ---- Step 2: Match to master communication events ----  
matched\_comms\_df <- other\_communications\_df %>%  
 filter(event\_id %in% event\_ids\_to\_extract)  
  
# ---- Step 3: Prepare data for plotting ----  
matched\_comms\_df\_for\_plot <- matched\_comms\_df %>%  
 mutate(  
 timestamp = as.POSIXct(timestamp),  
 comm\_date = as.Date(timestamp),  
 comm\_time\_of\_day = hms::as\_hms(format(timestamp, "%H:%M:%S")),  
 sender\_sub\_type = factor(sender\_sub\_type, levels = c("Person", "Vessel", "Organization", "Location")),  
 communicating\_pair\_sorted = paste(pmin(sender\_name, recipient\_name), pmax(sender\_name, recipient\_name), sep = " & ")  
 )  
  
# ---- Step 4: Add tooltip labels ----  
plot\_data <- matched\_comms\_df\_for\_plot %>%  
 mutate(  
 wrapped\_content = str\_wrap(content, width = 50),  
 tooltip\_text = paste0(  
 "<b>Date:</b> ", comm\_date, "<br>",  
 "<b>Time:</b> ", format(comm\_time\_of\_day, "%H:%M:%S"), "<br>",  
 "<b>Event ID:</b> ", event\_id, "<br><br>",  
 "<b>From:</b> ", sender\_name, "<br>",  
 "<b>To:</b> ", recipient\_name, "<br><br>",  
 "<b>Content:</b><br>", wrapped\_content  
 )  
 )  
  
# ---- Step 5: Generate timeline plot ----  
p <- ggplot(plot\_data, aes(x = comm\_date, y = comm\_time\_of\_day)) +  
 geom\_point(aes(  
 color = sender\_name,  
 shape = sender\_sub\_type,  
 text = tooltip\_text  
 ), show.legend = c(color = TRUE, shape = FALSE), size = 2, alpha = 0.7) +  
  
 scale\_shape\_manual(values = c(  
 "Person" = 16,  
 "Vessel" = 17,  
 "Organization" = 15,  
 "Location" = 18  
 )) +  
  
 scale\_y\_time(  
 limits = hms::as\_hms(c("08:00:00", "14:00:00")),  
 breaks = hms::as\_hms(c("08:00:00", "09:00:00", "10:00:00", "11:00:00", "12:00:00", "13:00:00", "14:00:00")),  
 labels = c("08:00", "09:00", "10:00", "11:00", "12:00", "13:00", "14:00")  
 ) +  
  
 scale\_x\_date(  
 date\_breaks = "1 day",  
 date\_labels = "%d %b"  
 ) +  
  
 facet\_wrap(~ sender\_sub\_type, ncol = 1, scales = "fixed") +  
  
 labs(  
 title = "Nadia's 3-hop Communications Timeline (Sender Perspective)",  
 x = "Date",  
 y = "Time of Day",  
 color = "Sender"  
 ) +  
  
 theme\_grey() +  
 theme(  
 axis.text.y = element\_text(size = 6),  
 axis.title.y = element\_text(size = 7),  
 axis.text.x = element\_text(size = 6, angle = 45, hjust = 1),  
 axis.title.x = element\_text(margin = margin(t = 10), size = 7),  
 panel.spacing = unit(0.5, "lines"),  
 strip.text = element\_text(size = 8, face = "bold"),  
 legend.position = "bottom",  
 legend.text = element\_text(size = 6),  
 legend.title = element\_text(size = 7)  
 )  
  
# ---- Step 6: Convert to interactive plot ----  
ggplotly(p, tooltip = "text")



## **9.5 Discussion/Interpretation (1 & 3 hop data):**

There were certain questions we posted to ourselves and came out with the answers.

### 9.5.1 The Community and Ego Network:

#### Question and Answer Analysis

| Question | Answer |
| --- | --- |
| Who were Nadia’s direct communication contacts (1-hop degree centrality)? Are any of them known to be suspicious or involved in illicit activities? | From the thicker width in the Ego network, it appeared that Nadia often communicated with **Liam**, **Elise**, and **Davis**.  Liam appeared to be The **Middleman** within Nadia’s direct community from the Louvain Community Network.  From the filtered table on suspicious relationships, **Elise**, **Liam**, **EcoVigil**, **Sentinel**, **Oceanus City Council**, and **V. Miesel Shipping** were noted to have suspicious relationships.  Their conversations were the area of focus to uncover their roles, relationships and identities. |
| Were there any other ‘Event’ or ‘Relationship’ nodes directly connected to Nadia in this communication network that hinted at suspicious people/ activities? | Based on information from question 2, **Rodriguez** was previously involved in **mining activities** that affected the environment. ‘**Mining**’ as a topic and his conversations would be tracked. |

### 9.5.2 The Communication Timeline and Content:

In the nadia\_full\_communications\_timeline table, the actual content of her direct two-way communications were investigated over the course of 9 days. There were certain suspicious entities, keywords, coded language, or unusual topics detected that were suspicious. We have tabled out the segmented suspicious and non-suspicious entities for investigation and elimination.

### Not in the Network but Mentioned in the Content

| Entity | Form of Subject Matter | Rationale | Event ID |
| --- | --- | --- | --- |
| Nemo Reef | Location | Likely conservation area which was picked by the characters for illicit activities. | 331, 943 |
| Permit #CR-7844 | Item | Likely a permit to show tourism activity as a cover for suspicious activities. Rodriguez is likely linked to vessels Mako, Neptune, and Remora operating under this permit with a tourism facade. | 582, 847, 805 |
| EcoVigil | Vessel | EcoVigil will likely affect Nadia’s operations when they use their ROV. Nadia recommended to V. Miesel to accelerate the planned operation. They were likely working for different sides. | 753, 847 |

### Suspicious Entity

| Suspicious Entity | Sub Type | Community | Rationale | Event ID |
| --- | --- | --- | --- | --- |
| Haacklee Harbor | Location | 2 | Nadia’s communication to Haacklee Harbor is suspicious when she wanted the documents destroyed and the special corridor to Nemo Reef cancelled. | 331 |
| Liam aka The Middleman | Person | 2 | Nadia wanted him to double his usual fee to ensure Harbor Master remains cooperative. Identity revealed through: | 529, 795, 847 |
| Davis | Person | 1 | Nadia told him to create a clean paper trail. She will provide permits. | 521 |
| Elise aka Mrs. Money | Person | 1 | Nadia warned Elise that conservation vessels might complicate their operation. Identity revealed through: | 708, 528, 538, 677 |
| Rodriguez aka Small Fry | Person | 1 | Linked to Mako, which is operating under a permit with a tourism facade. | 805 |
| Remora | Vessel | 1 | Remora reported a tourism facade to Nadia and planned underwater lighting placements in Nemo Reef while monitoring conservation vessels. | 943 |
| Neptune | Vessel | 1 | Nadia told Neptune to stay under the radar. | 538 |
| V. Miesel HQ | Organisation | 1 | Organisation was aware of the suspicious permit and The Middleman. | 846, 847 |
| Sailor Shifts Team aka Glitters Team | Organisation | 1 | Nadia provided crew members for the setup related to the permit. | 520 |

### Non Suspicious Entity

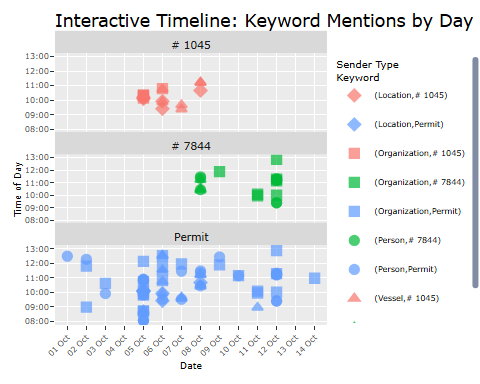
| Non Suspicious Entity | Sub Type | Community | Rationale | Event ID |
| --- | --- | --- | --- | --- |
| Oceanus City Council | Organisation | 2 | Oceanus City Council as a whole is not suspicious as an organisation as Liam stated this organisation suspected nothing. | 535 |
| Sentinel | Vessel | 2 | Sentinel Vessel suspected ulterior motives as the water quality was fine. | 677 |
| Marlin | Vessel | 2 | Marlin inquired about unusual vessel routes near eastern shoals, Nadia and Davis might need to address this. | 584 |

## **9.6 Querying Keywords**

Based on our knowledge, we decided to connect to external information. We used knowledge from Nadia’s current network and communication with keyword search of our queries.

### 9.6.1 Plotted Timeline (Word Query: Permit related)

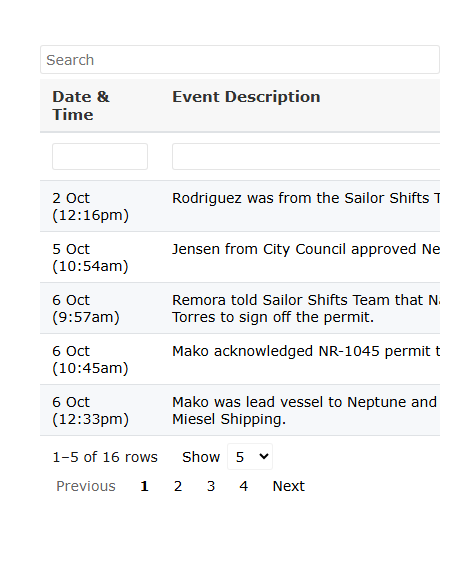
# -- Step 1: Define keywords  
keywords <- c("permit", "1045", "7844")  
pattern <- paste0("\\b(", paste(keywords, collapse = "|"), ")\\b")  
  
# -- Step 2: Filter messages with keyword match (case-insensitive)  
keyword\_matches\_df <- other\_communications\_df %>%  
 mutate(content\_lower = tolower(content)) %>%  
 filter(str\_detect(content\_lower, pattern))  
  
# -- Step 3: Extract and annotate keyword matches  
plot\_data <- keyword\_matches\_df %>%  
 mutate(  
 matched\_keywords = str\_extract\_all(content\_lower, pattern),  
 timestamp = suppressWarnings(as.POSIXct(timestamp)),  
 comm\_date = as.Date(timestamp),  
 comm\_time\_of\_day = suppressWarnings(hms::as\_hms(format(timestamp, "%H:%M:%S"))),  
 wrapped\_content = str\_wrap(content, width = 50)  
 ) %>%  
 unnest(matched\_keywords) %>%  
 mutate(  
 matched\_keywords = recode(matched\_keywords,  
 "permit" = "Permit",  
 "1045" = "# 1045",  
 "7844" = "# 7844"  
 ),  
 tooltip\_text = paste0(  
 "<b>Date:</b> ", comm\_date,  
 "<br><b>Time:</b> ", format(comm\_time\_of\_day, "%H:%M:%S"),  
 "<br><b>Event ID:</b> ", event\_id,  
 "<br><b>Content:</b><br>", wrapped\_content  
 )  
 )  
  
# -- Step 4: Plot timeline  
p <- ggplot(plot\_data, aes(x = comm\_date, y = comm\_time\_of\_day)) +  
 geom\_point(aes(  
 color = matched\_keywords,  
 shape = sender\_sub\_type,  
 text = tooltip\_text,  
 group = matched\_keywords  
 ), size = 2.5, alpha = 0.7, show.legend = TRUE) +  
 scale\_shape\_manual(values = c(  
 "Person" = 16,  
 "Vessel" = 17,  
 "Organization" = 15,  
 "Location" = 18  
 )) +  
 facet\_wrap(~ matched\_keywords, ncol = 1, scales = "fixed") +  
 scale\_y\_time(  
 limits = hms::as\_hms(c("08:00:00", "13:00:00")),  
 breaks = hms::as\_hms(c("08:00:00", "09:00:00", "10:00:00", "11:00:00", "12:00:00", "13:00:00")),  
 labels = c("08:00", "09:00", "10:00", "11:00", "12:00", "13:00")  
 ) +  
 scale\_x\_date(date\_breaks = "1 day", date\_labels = "%d %b") +  
 labs(  
 title = "Interactive Timeline: Keyword Mentions by Day and Time",  
 x = "Date",  
 y = "Time of Day",  
 shape = "Sender Type",  
 color = "Keyword"  
 ) +  
 theme\_grey() +  
 theme(  
 axis.text.y = element\_text(size = 6),  
 axis.title.y = element\_text(size = 7),  
 axis.ticks.y = element\_line(),  
 axis.text.x = element\_text(size = 6, angle = 45, hjust = 1),  
 axis.title.x = element\_text(margin = margin(t = 10), size = 7),  
 panel.spacing = unit(0.5, "lines"),  
 strip.text = element\_text(size = 8, face = "bold"),  
 legend.position = "bottom",  
 legend.text = element\_text(size = 6),  
 legend.title = element\_blank()  
 )  
  
# -- Step 5: Convert to plotly  
ggplotly(p, tooltip = "text")



### 9.6.1.1 Findings in sequential order:

* There were overlaps as the blue permit also included the red NR-1045 permit and green CR-7844 permit.

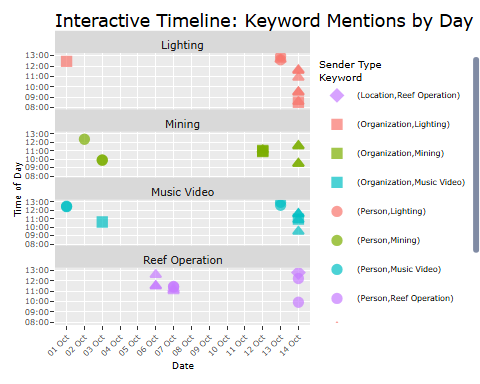
library(reactable)  
library(dplyr)  
library(stringr)  
library(readr) # Used for read\_lines  
  
# 1. Prepare data from the text provided  
data\_text <- "  
Date & Time | Event Description  
------------|---------------------------------------------------------------------------------------------------------------  
2 Oct (12:16pm) | Rodriguez was from the Sailor Shifts Team.  
5 Oct (10:54am) | Jensen from City Council approved Nemo Reef permit.  
6 Oct (9:57am) | Remora told Sailor Shifts Team that Nadia got Commissioner Torres to sign off the permit.  
6 Oct (10:45am) | Mako acknowledged NR-1045 permit to Nemo Reef.  
6 Oct (12:33pm) | Mako was lead vessel to Neptune and Remora as authorised by V. Miesel Shipping.  
7 Oct (9:40am) | Mako was operating under permit NR-1045 for conservation research.  
8 Oct (10:24am) | Mako was operating under V. Miesel's special marine research permit CR-7844 approved by Oceanus City Council.  
8 Oct (10:30am) | Mako informed Remora that both of them were operating under permit CR-7844 and have a 5 day deadline.  
8 Oct (10:40am) | Remora was approved by Paackland Harbor to operate with extended hours under permit NR-1045.  
9 Oct (11:53am) | Mako requesed for additional crew from v Miesel Shipping for the 24hr operations over next 5 days.  
11 Oct (6:00am) | Nemo Reef closure mandated by Oceanus City Council.  
11 Oct (8:57am) | All research permits must be submitted within 72 hours.  
11 Oct (10:05am) | V. Miesel Shipping informed Remora that 30% of her crew to be reassigned to Neptune.  
12 Oct (10:01am) | Davis as Captain oversaw crew reallocation.  
12 Oct (11:19am) | Nadia secured documentation for CR-7844.  
12 Oct (12:52pm) | Harbor closure for 3 days from 13 Oct 06:00.  
"  
  
# Parse the data into a data frame  
# Use read\_lines to handle the multi-line string  
data\_lines <- read\_lines(data\_text)  
  
# Remove the header separator line and any empty lines  
data\_lines <- data\_lines[!grepl("^---|===", data\_lines) & data\_lines != ""]  
  
# The first non-separator line is the header  
col\_names\_raw <- str\_split(data\_lines[1], "\\|")[[1]]  
col\_names <- str\_trim(col\_names\_raw)  
  
# The rest are data rows  
data\_content <- data\_lines[-1]  
  
# Create data frame by splitting lines and putting into a tibble  
df <- tibble(line = data\_content) %>%  
 mutate(  
 `Date & Time` = str\_trim(str\_extract(line, "^[^|]+")),  
 `Event Description` = str\_trim(str\_extract(line, "(?<=\\|).+$"))  
 ) %>%  
 select(`Date & Time`, `Event Description`) # Ensure correct column order and names  
  
# 2. Create the reactable table with desired features  
  
reactable(  
 df,  
 filterable = TRUE, # Enable column-specific filters (from the tutorial)  
 searchable = TRUE, # Enable global search box (from the tutorial)  
 paginationType = "numbers", # Display page numbers (corrected from "pages")  
 defaultPageSize = 5, # Show 5 rows per page  
 showPageSizeOptions = TRUE, # Allow users to change page size  
 pageSizeOptions = c(5, 10, 15, 20, 50), # Options for page sizes  
 striped = TRUE, # Add alternating row colors (from the tutorial)  
 highlight = TRUE, # Highlight row on hover (from the tutorial)  
 columns = list(  
 `Date & Time` = colDef(  
 name = "Date & Time",  
 minWidth = 120, # Adjust width to fit content  
 align = "left"  
 ),  
 `Event Description` = colDef(  
 name = "Event Description",  
 minWidth = 500, # Ensure enough width for event descriptions  
 align = "left"  
 )  
 ),  
 # Apply a custom theme for better aesthetics (inspired by the tutorial)  
 theme = reactableTheme(  
 borderColor = "#dfe2e5",  
 stripedColor = "#f6f8fa",  
 highlightColor = "#f0f5f9",  
 cellPadding = "8px 12px",  
 style = list(fontFamily = "Verdana, Geneva, sans-serif", fontSize = "14px"),  
 headerStyle = list(  
 "&.rt-th:hover" = list(backgroundColor = "#e0e6eb"),  
 fontSize = "15px",  
 fontWeight = 600,  
 color = "#333", # Darker header text for contrast  
 background = "#f7f7f7" # Slightly grey background for header  
 ),  
 rowSelectedStyle = list(backgroundColor = "#e6f2ff", "&:hover" = list(backgroundColor = "#e6f2ff")),  
 # Styles for search/filter inputs (from tutorial's theme example)  
 searchInputStyle = list(width = "100%", margin = "5px 0", padding = "5px"),  
 filterInputStyle = list(width = "100%", margin = "2px 0", padding = "4px")  
 )  
)



* Since there were little communications on 13 Oct, we looked into other word queries.
* Using information obtained from question 2 and 4, we had in mind certain keywords to query for.

### 9.6.2 Plotted Timeline (Word Query: Music Video Related)

library(dplyr)  
library(stringr)  
library(ggplot2)  
library(plotly)  
library(hms)  
library(tidyr)  
  
# -- Step 1: Define keywords  
keywords <- c("mine", "mining", "music video", "lighting", "reef operation")  
pattern <- paste0("\\b(", paste(keywords, collapse = "|"), ")\\b")  
  
# -- Step 2: Filter messages  
search\_target\_content <- tolower(other\_communications\_df$content)  
  
keyword\_matches\_df <- other\_communications\_df %>%  
 filter(str\_detect(search\_target\_content, pattern))  
  
# -- Step 3: Extract keywords  
plot\_data <- keyword\_matches\_df %>%  
 mutate(matched\_keywords = str\_extract\_all(tolower(content), pattern)) %>%  
 unnest(matched\_keywords) %>%  
 mutate(matched\_keywords = str\_to\_title(matched\_keywords)) %>%  
 arrange(matched\_keywords, timestamp) %>%  
 mutate(  
 timestamp = as.POSIXct(timestamp),  
 comm\_date = as.Date(timestamp),  
 comm\_time\_of\_day = hms::as\_hms(format(timestamp, "%H:%M:%S")),  
 wrapped\_content = str\_wrap(content, width = 50),  
 tooltip\_text = paste0(  
 "<b>Date:</b> ", comm\_date,  
 "<br><b>Time:</b> ", comm\_time\_of\_day,  
 "<br><b>Event ID:</b> ", event\_id,  
 "<br><b>From:</b> ", sender\_name,  
 "<br><b>To:</b> ", recipient\_name,  
 "<br><b>Content:</b><br>", wrapped\_content  
 )  
 )  
  
# -- Step 4: Plot  
p <- ggplot(plot\_data, aes(x = comm\_date, y = comm\_time\_of\_day)) +  
 geom\_point(aes(  
 color = matched\_keywords,  
 shape = sender\_sub\_type,  
 text = tooltip\_text,  
 group = matched\_keywords # ensures matched\_keywords is in layer  
 ), size = 2.5, alpha = 0.7, show.legend = TRUE) +  
 scale\_shape\_manual(values = c(  
 "Person" = 16,  
 "Vessel" = 17,  
 "Organization" = 15,  
 "Location" = 18  
 )) +  
 facet\_wrap(~ matched\_keywords, ncol = 1, scales = "fixed") +  
 scale\_y\_time(  
 limits = hms::as\_hms(c("08:00:00", "13:00:00")),  
 breaks = hms::as\_hms(c("08:00:00", "09:00:00", "10:00:00", "11:00:00", "12:00:00", "13:00:00")),  
 labels = c("08:00", "09:00", "10:00", "11:00", "12:00", "13:00")  
 ) +  
 scale\_x\_date(date\_breaks = "1 day", date\_labels = "%d %b") +  
 labs(  
 title = "Interactive Timeline: Keyword Mentions by Day and Time",  
 x = "Date",  
 y = "Time of Day",  
 shape = "Sender Type",  
 color = "Keyword"  
 ) +  
 theme\_grey() +  
 theme(  
 axis.text.y = element\_text(size = 6),  
 axis.title.y = element\_text(size = 7),  
 axis.ticks.y = element\_line(),  
 axis.text.x = element\_text(size = 6, angle = 45, hjust = 1),  
 axis.title.x = element\_text(margin = margin(t = 10), size = 7),  
 panel.spacing = unit(0.5, "lines"), # Applies to both x and y spacing  
 strip.text = element\_text(size = 8, face = "bold"),  
 legend.position = "bottom",  
 legend.text = element\_text(size = 6),  
 legend.title = element\_blank()  
 )  
  
# -- Step 5: Convert to plotly  
ggplotly(p, tooltip = "text")



### 9.6.2.1 Findings through Questions & Answers:

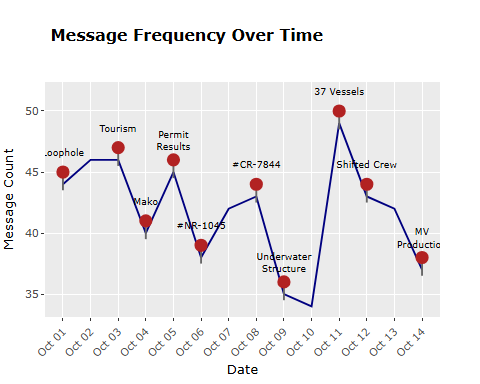
#### **Question and Answer Analysis**

| **Question** | **Answer** |
| --- | --- |
| Which vessel was the suspicious permit #CR-7844 prepared for?  (See 9.6.1.1) | From question 2 and 4, we know this is V. Miesel’s Marine Research Permit, and prepared for Mako (lead vessel), Neptune, and Remora. |
| What suspicious activity was performed at Nemo Reef? Which day was it?  (See 9.6.2) | 14 Oct 2040 for a music video production. |
| Why is underwater lighting placement needed at Nemo Reef?  (See 9.6.2) | For a music video production. |
| What were the expedited approvals and secretive logistics?  (See 8.6.1) | Permits for Nemo Reef through NR-1045 and CR-788 were expedited. The secretive logistics were the crates and equipment on the vessels for the music video production. |
| Who were the high-level Oceanus officials, Sailor Shift’s team, local influential families, and local conservationist group?  (Various Ids) | **Oceanus Officials**: Commissioner Blake, Commissioner Torres, Council Knowles, The Middleman, Jensen, Liam Thorne  **Sailor Shift’s Team**: Boss, Council Knowles, Davis, Glitters Team, Liam Thorne, Mako, Mrs. Money, Nadia, Neptune, Remora, Rodriguez, Sam, Samantha Blake, Small Fry, The Accountant, The Intern, The Middleman  **Local Influential Families**: Council Knowles, V. Miesel Shipping  **Local Conservationist Group**: Defender, EcoVigil, Green Guardians, Horizon, Kelly, Reef Guardians, Seawatch, Sentinel, The Lookout |
| Was the music video production activity legal?  (Id 979) | There was no environmental damage or mining involved in the music production. However, an environmental assessment was not conducted prior. Clepper may assess that his suspicions about Nadia Conti’s illicit activity may not be straightforward and could depend on whether an assessment was mandatory before commercial activities. |

# **10) Recreation of Story-Line**

Here we attempted to summarise the story-line.

# --- Step 0: Define Event Data for Annotations ---  
event\_dates <- as.Date(c("2040-10-01", "2040-10-03", "2040-10-04", "2040-10-05",  
 "2040-10-06", "2040-10-08", "2040-10-09", "2040-10-11",  
 "2040-10-12", "2040-10-14"))  
  
event\_labels\_full <- c(  
 "The Middleman told The Boss (Nadia) about a potential loophole in Nemo Reef protection zone",  
 "Tourism vessels were offering authorised access. Their violations were addressed at City Council meeting",  
 "Nadia told Mako to abort Nemo Reef mission as conservation vessels were close",  
 "News of Nadia assisting to get permit for Nemo Reef signed by Jensen from City Council on 4/10 spreaded",  
 "V. Miesel Shipping HQ told Neptune that Mako is lead vessel",  
 "Neptune told Mako they were under special marine research permit CR-7844",  
 "Neptune's underwater concrete forms in fragile crates discovered",  
 "37 'authorised maintenance vessels' documented despite Nemo Reef closure",  
 "Crew reallocated from Remona to Neptune overseen by Captain Davis",  
 "Music video production after Nemo Reef equipment transfer"  
)  
  
event\_labels\_short <- c(  
 "Loophole", "Tourism", "Mako", "Permit\nResults", "#NR-1045",  
 "#CR-7844", "Underwater\nStructure", "37 Vessels", "Shifted Crew", "MV\nProduction"  
)  
  
# --- Step 1: Calculate Daily Message Frequencies ---  
freq\_df <- other\_communications\_df %>%  
 mutate(comm\_date = as.Date(timestamp)) %>%  
 count(comm\_date, name = "message\_count") %>%  
 # Complete the date range to include all days, filling missing counts with 0  
 complete(comm\_date = seq(min(comm\_date), max(comm\_date), by = "day"),  
 fill = list(message\_count = 0))  
  
# --- Step 2: Build Event Annotation Table ---  
event\_df <- tibble(  
 date = event\_dates,  
 label = gsub("\n", "<br>", event\_labels\_short), # Replace \n with <br> for HTML tooltips  
 tooltip\_raw = str\_wrap(event\_labels\_full, width = 40) # Wrap long text for tooltips  
)  
  
core\_events <- event\_df %>%  
 left\_join(freq\_df, by = c("date" = "comm\_date")) %>%  
 group\_by(date) %>%  
 mutate(  
 offset = row\_number(), # For stacking multiple events on the same day  
 # Adjust value for arrow positioning relative to message count  
 value = as.numeric(message\_count) + 0.5 \* offset,  
 date\_jitter = date + (offset - 1) \* 0.1 # Slight horizontal jitter if multiple events  
 ) %>%  
 ungroup() %>%  
 mutate(  
 # Create rich HTML tooltip content for Plotly  
 tooltip = paste0(  
 "📅 <b>Date:</b> ", date,  
 "<br>✉️ <b>Messages:</b> ", message\_count,  
 "<br>📝 <b>Core Event:</b> ", tooltip\_raw  
 )  
 )  
  
# --- Step 3: Create the ggplot2 Plot ---  
p <- ggplot(freq\_df, aes(x = comm\_date, y = message\_count)) +  
 # Line plot for message frequency  
 geom\_line(color = "navyblue") +  
 # Segments (arrows) pointing from the line to the event annotations  
 geom\_segment(  
 data = core\_events,  
 aes(x = date, xend = date, y = value - 1, yend = value + 0.5), # Adjust yend for arrow tip  
 arrow = arrow(length = unit(0.15, "inches")), color = "grey40"  
 ) +  
 # Points for the events (these will have the interactive tooltips)  
 geom\_point(  
 data = core\_events,  
 aes(x = date, y = value + 0.5, text = tooltip), # 'text' aesthetic is key for Plotly tooltips  
 color = "firebrick", size = 3  
 ) +  
 # Text labels for the short event names  
 geom\_text(  
 data = core\_events,  
 aes(x = date, y = value + 2, label = label), # Position text above the point/arrow  
 color = "black", fontface = "bold", size = 2.5, vjust = 0  
 ) +  
 # X-axis scale for dates  
 scale\_x\_date(date\_breaks = "1 day", date\_labels = "%b %d") +  
 # Labels and titles  
 labs(  
 title = "Message Frequency Over Time",  
 x = "Date", y = "Message Count"  
 ) +  
 # Ensure annotations are not clipped by plot limits  
 coord\_cartesian(clip = "off") +  
 # Minimal theme for a clean look  
 theme\_grey(base\_size = 10) +  
 # Further theme adjustments  
 theme(  
 axis.text.x = element\_text(angle = 45, hjust = 1),  
 plot.title = element\_text(face="bold"),  
 plot.margin = margin(30, 30, 10, 10), # Adjust plot margins if needed  
 )  
  
# --- Step 4: Convert to Interactive Plotly Object ---  
# The 'tooltip = "text"' argument tells Plotly to use the 'text' aesthetic  
# for the hover information, which we created as 'tooltip' in core\_events.  
p\_interactive <- ggplotly(p, tooltip = "text")  
  
# --- Step 5: Display the Interactive Plot ---  
p\_interactive



Hover over the red points to see the details. Here, we provided a high level summary on the core event on the preparation and production of the music video Nadia was involved in for a period of 2 weeks, and also provided a snippet to the sub-event of the tourism vessels breaching protocols earlier in the week.

# **11) References**

* Datastorm (2021) [visNetwork](https://datastorm-open.github.io/visNetwork/)
* Dr. Kam Tin Seong (2025) [Lesson 9: Visualising and Analysis Network Data](https://isss608-ay2024-25apr.netlify.app/outline/lesson09_outline)
* R-Graph (2025) [Reactable](https://r-graph-gallery.com/package/reactable.html)