

Baltic Marine Environment Protection Commission

How we made shipping density maps



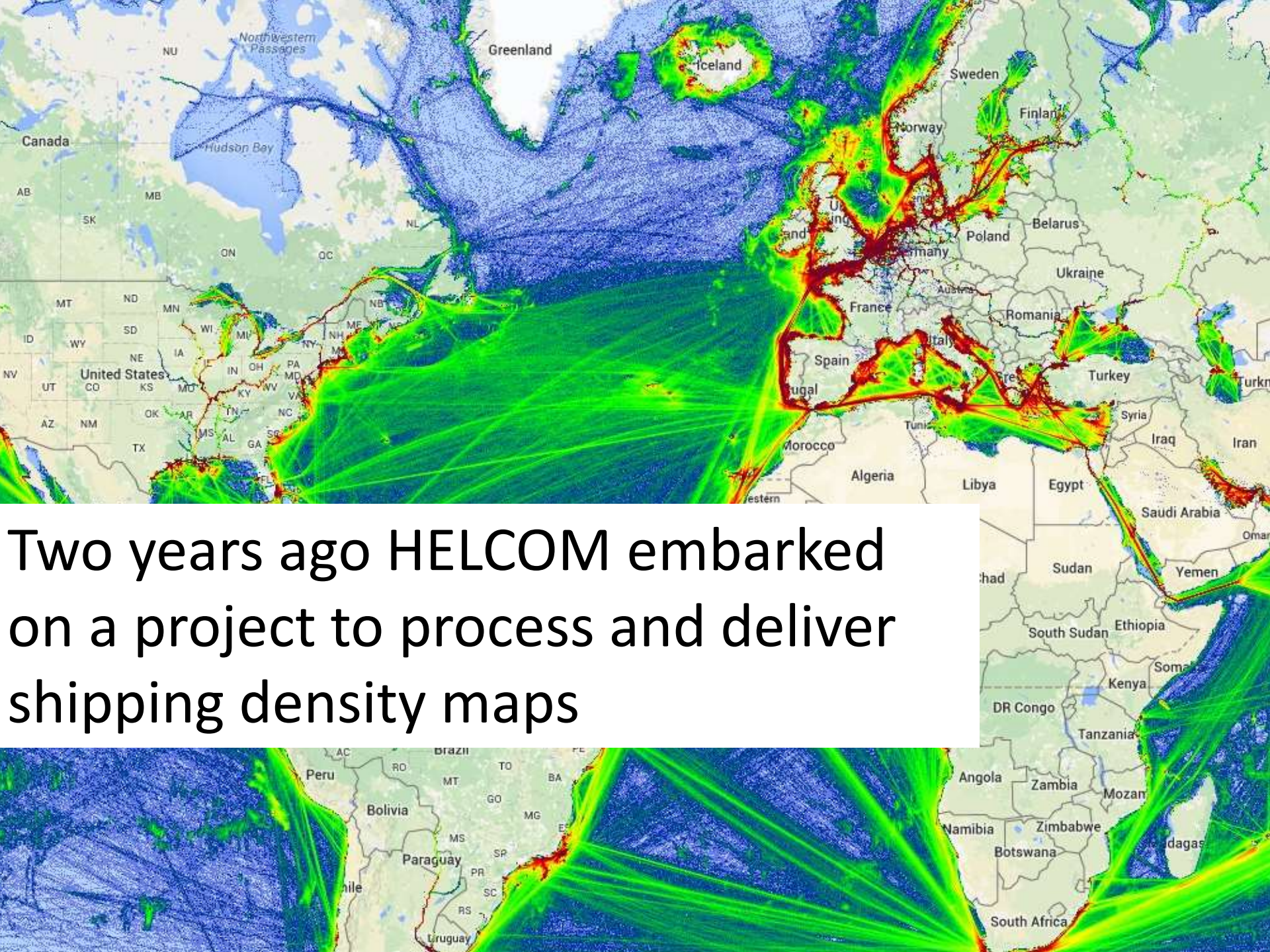


Manuel Frias
AIS Seminar 9 May 2017
HELCOM Secretariat, Helsinki, Finland

Let’s play!

First, let’s have a look at the maps we produced

https://goo.gl/qlst9R



Two years ago HELCOM embarked on a project to process and deliver shipping density maps

We started reserching and came to

two conclusions

1 There are different methods

$$\hat{f}(x) = \frac{1}{2hn}\ [\text{no. of } X_1, \ldots, X_n \text{ falling in } (x-h, x+h)];$$

$$P(a < X < b) = \int_a^b f(d)dx \quad \text{for all } a < b.$$

Grid cell

radius

L2

$$\hat{f}(x) = \frac{1}{n}\sum_{i+1}^{n} \frac{1}{h} w\left(\frac{x - X_i}{h}\right).$$

$$f(x) = \lim_{h \to 0} \frac{1}{2h} P(x-h < X < x+h).$$

Density = (... / (area_of_circle)

$$\hat{f}(x) = \frac{1}{nh}(\text{no. of } X_i \text{ in the same bin as } x).$$

$$\hat{f}(x) = \frac{1}{n}\sum_{i+1}^{n} \frac{1}{h} w\left(\frac{x - X_i}{h}\right).$$

line

$$SearchRadius = 0.9 * \min\left(SD, \sqrt{\frac{1}{\ln(2)}} * D_m\right) * n^{-0.2}$$



2 The result of many methods is based on complicated formulas—very difficult to understand!



So we embraced the

KISS principle

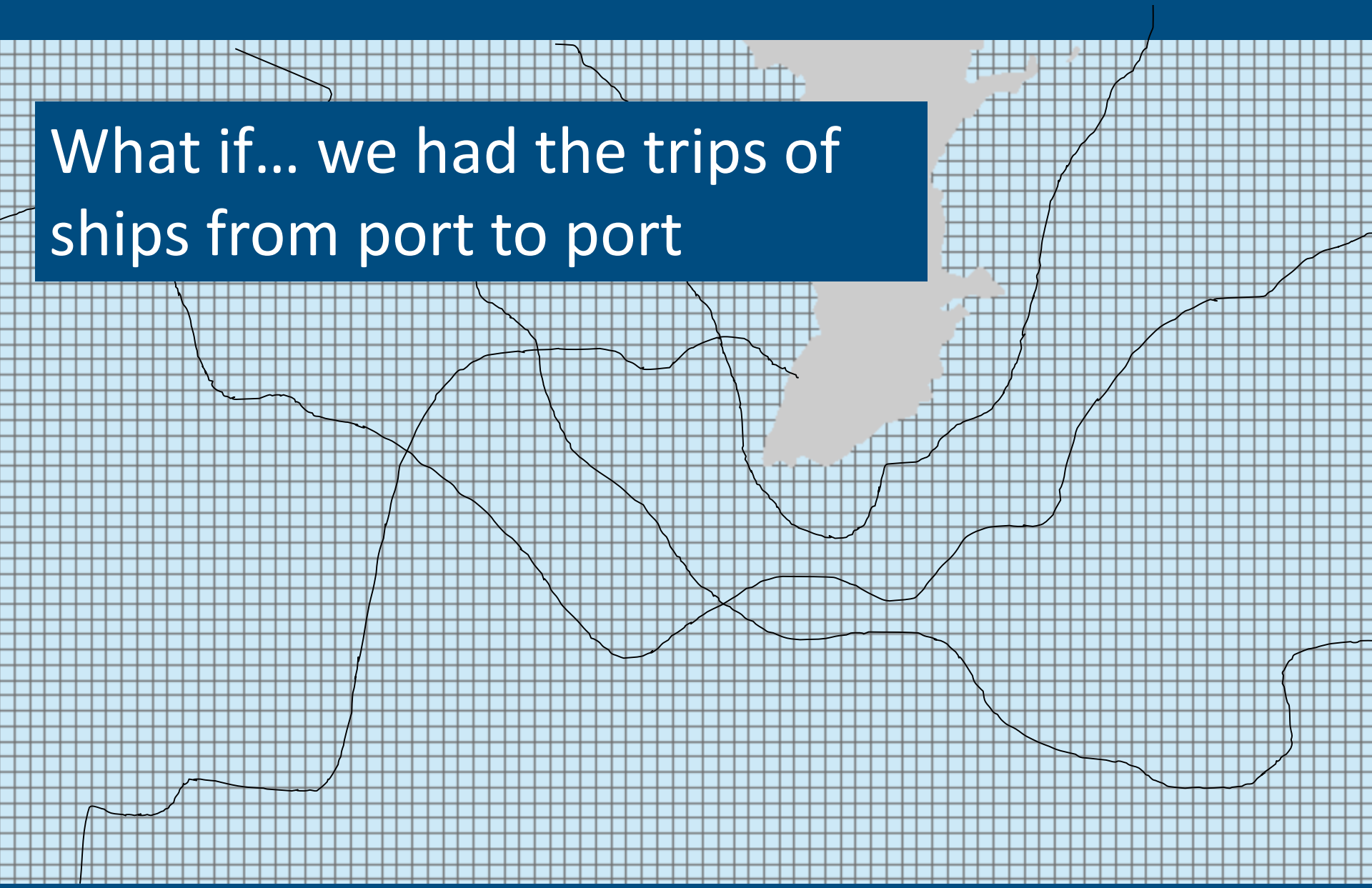
KISS principle

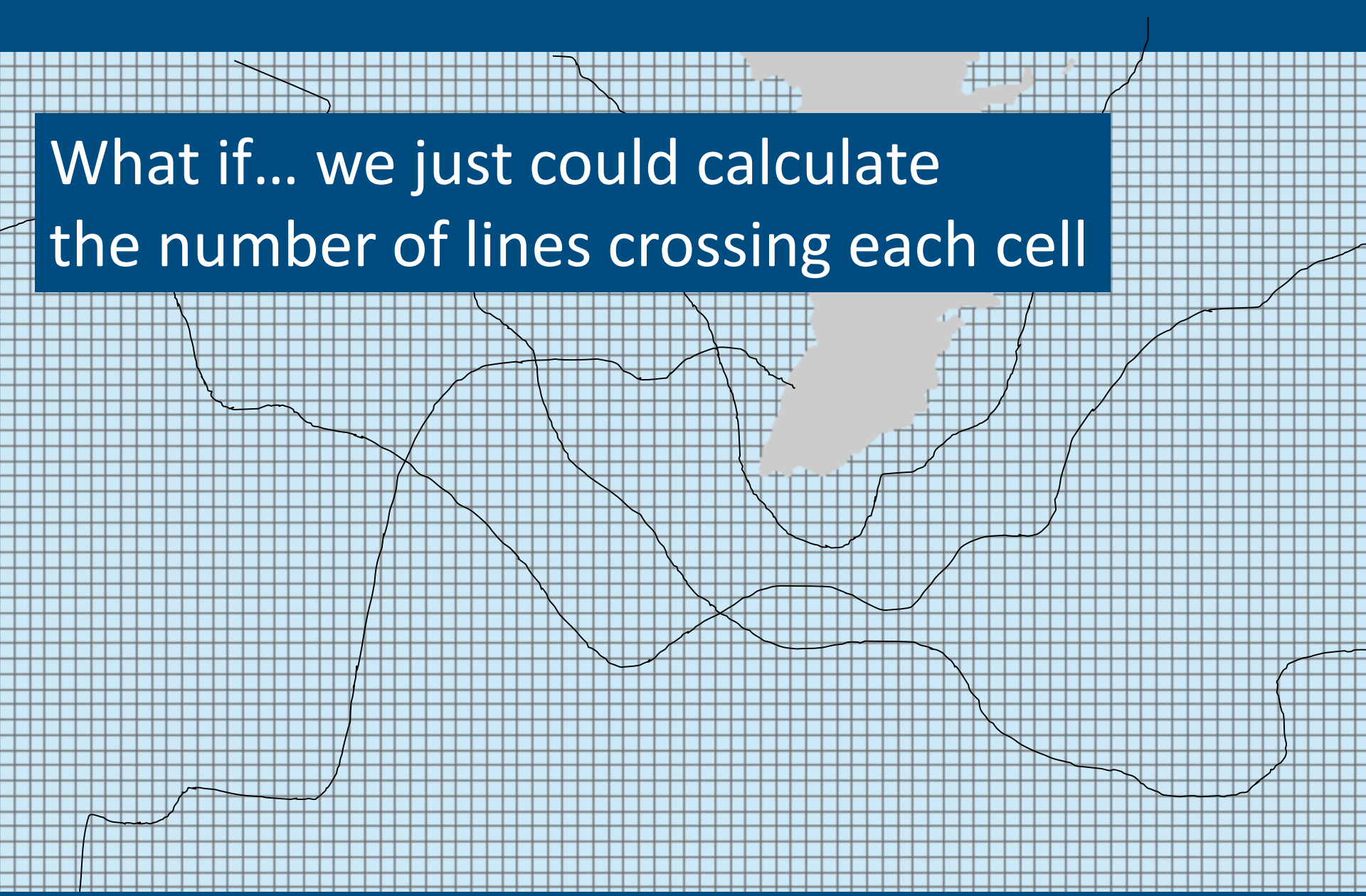
Keep It Simple Stupid

What if… we had a grid

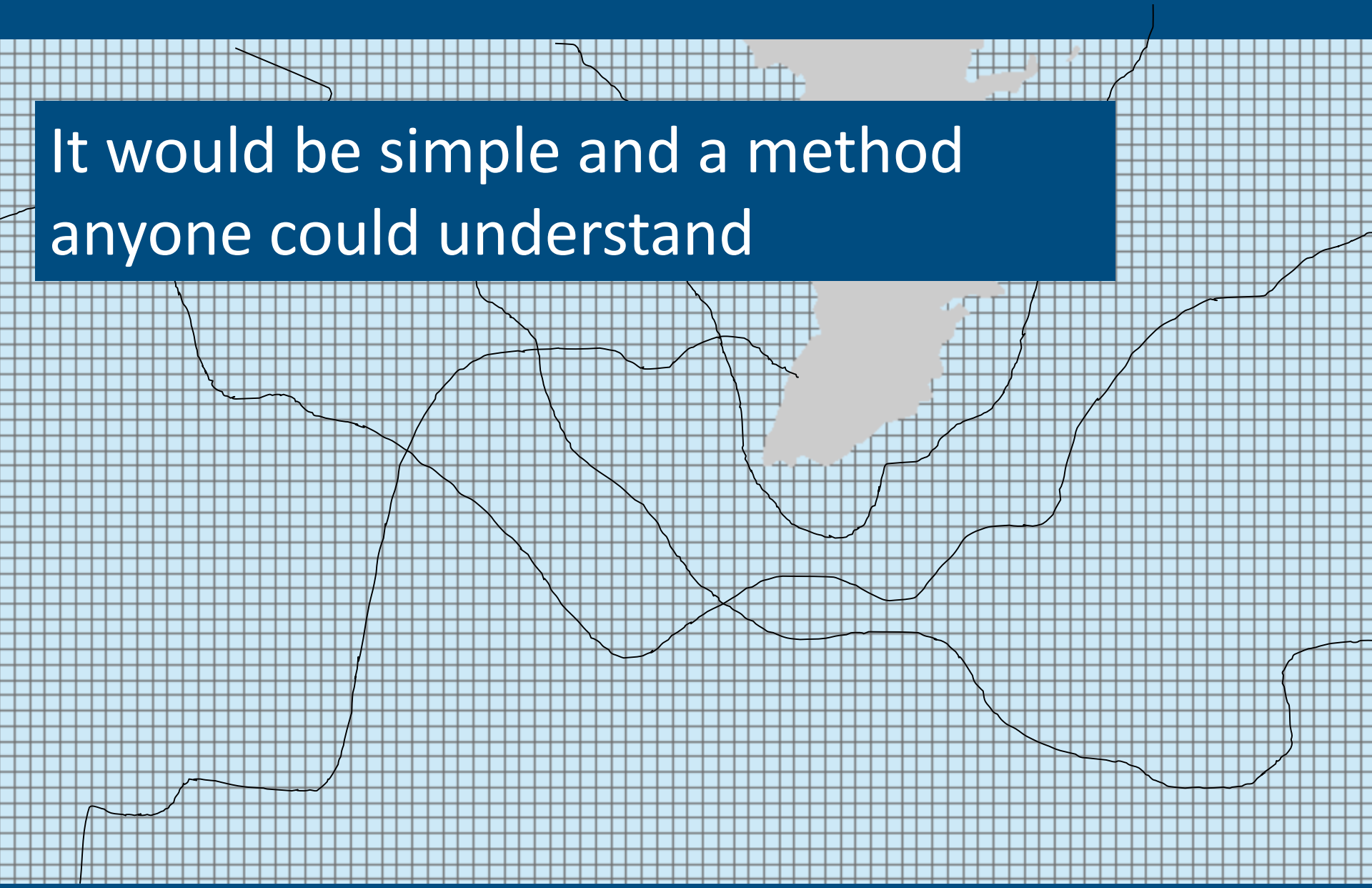


What if… we had the trips of ships from port to port

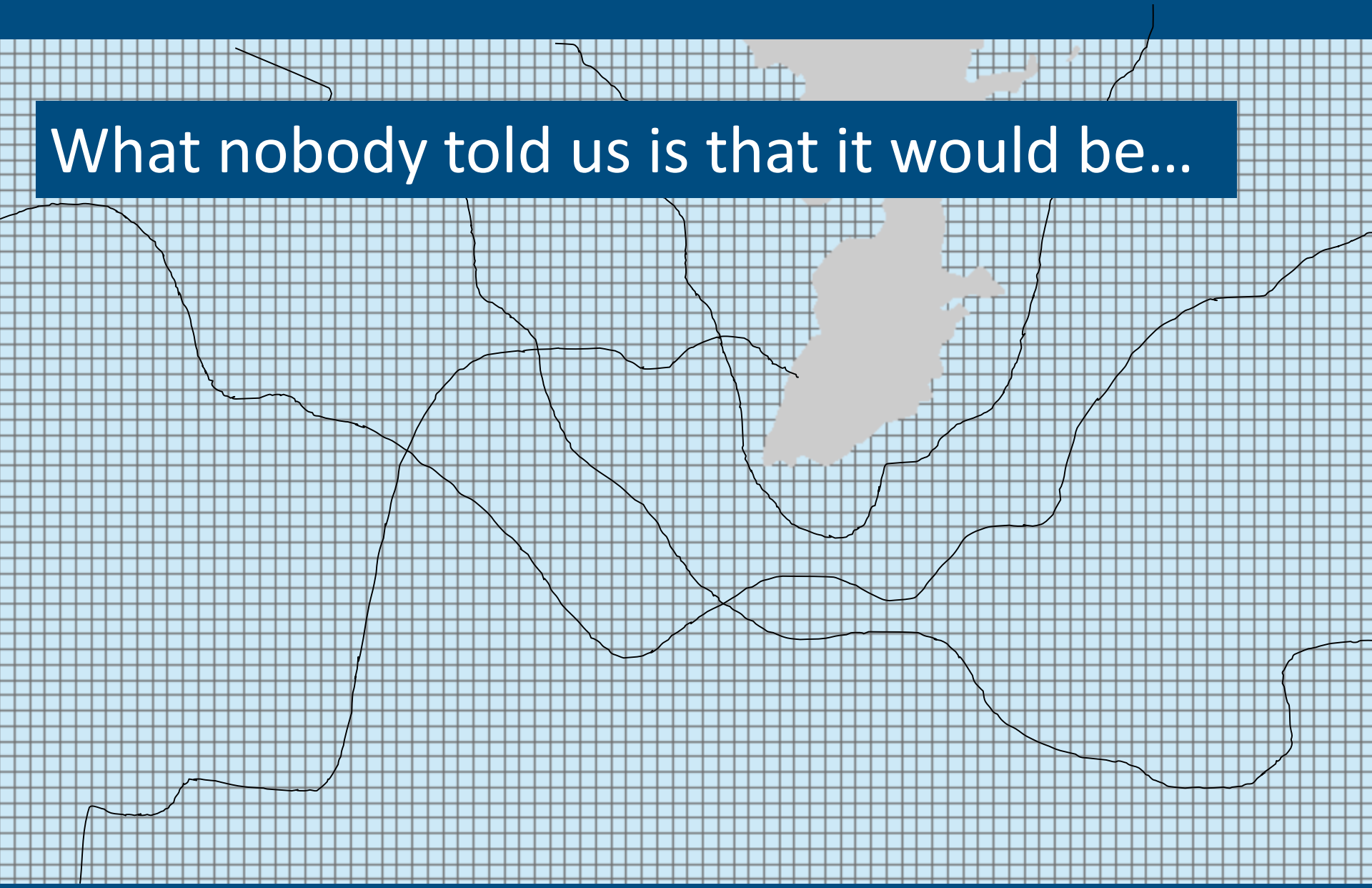



What if… we just could calculate
the number of lines crossing each cell

It would be simple and a method anyone could understand



What nobody told us is that it would be…



…so difficult!!



After many, many, many trials we got it.



And we found out we needed three things

Ports



Ports



Stops



Ports



Stops



Trips



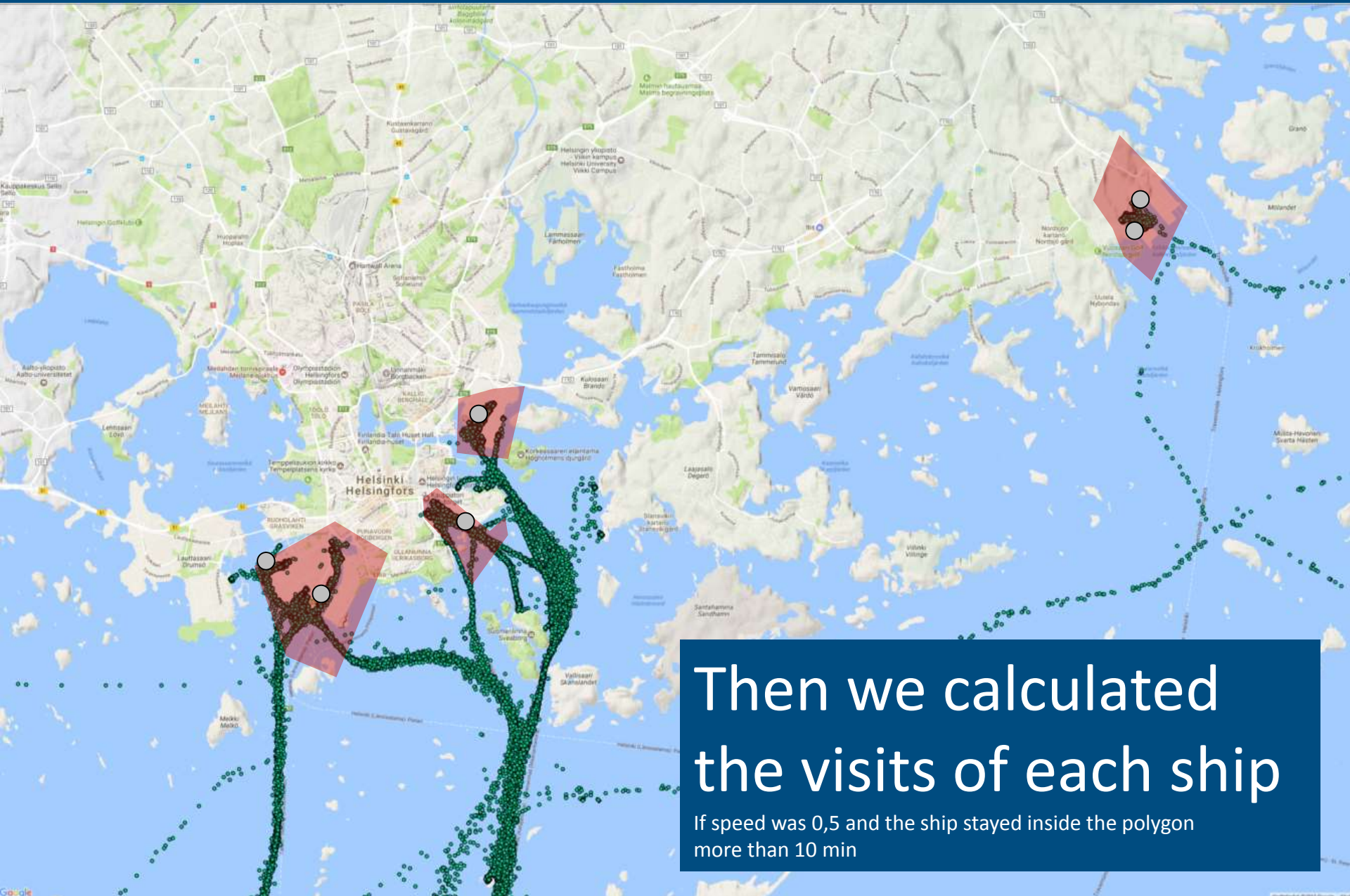
Ports Stops Trips



We digitized all ports in the Baltic Sea

Thanks Nazmus, Thaysa, Fana and Princess, outstanding Erasmus Mundus MSP trainees

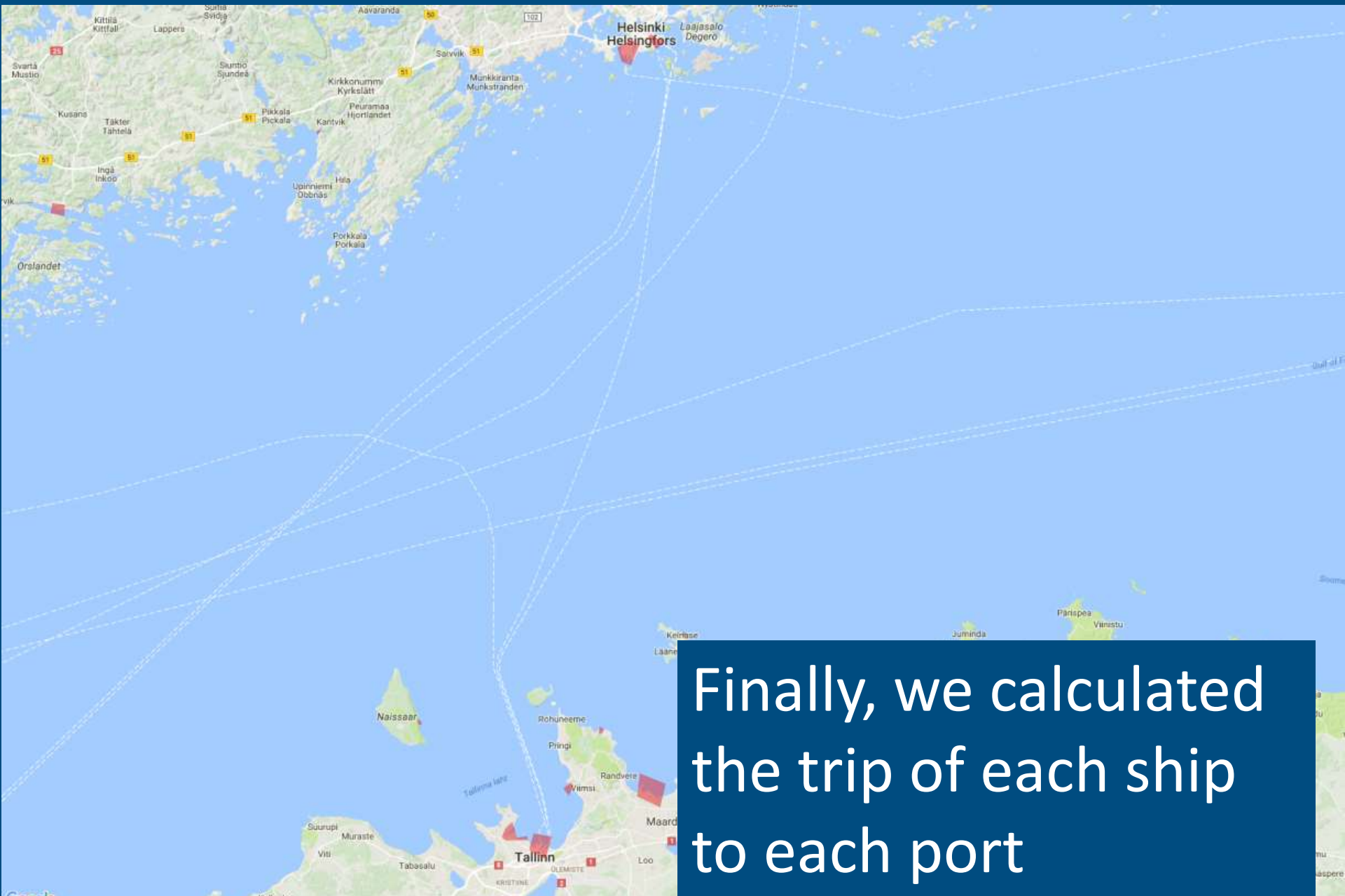
Ports Stops Trips



Then we calculated the visits of each ship

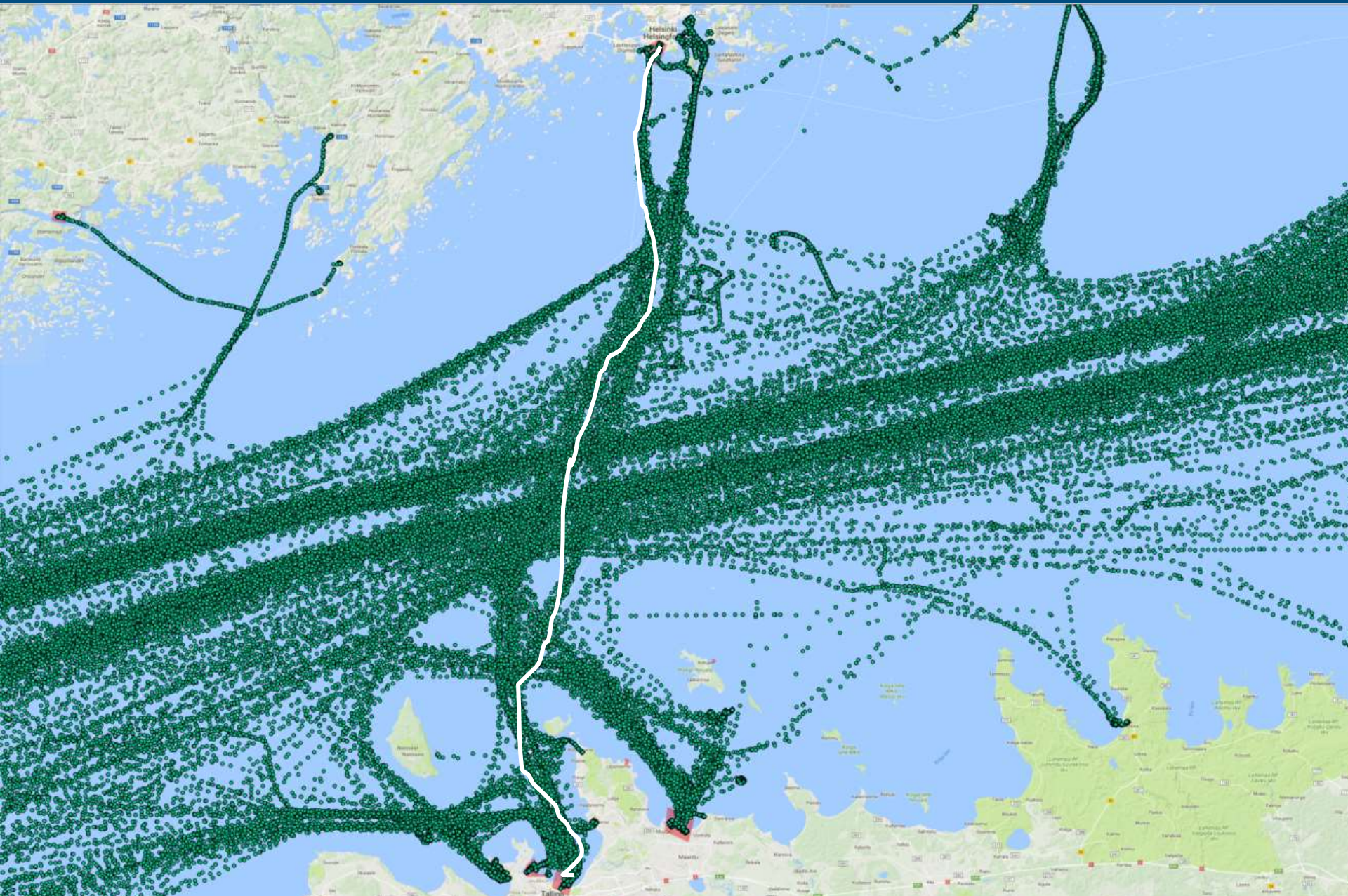
If speed was 0,5 and the ship stayed inside the polygon more than 10 min

Ports Stops Trips



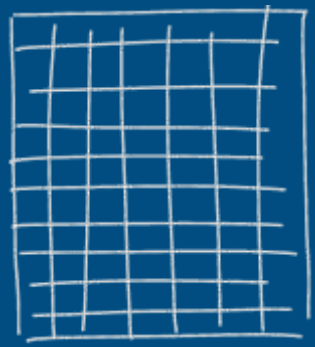
Finally, we calculated the trip of each ship to each port

Ports Stops Trips



But that’s not all! The final step involves one more thing…

Grid





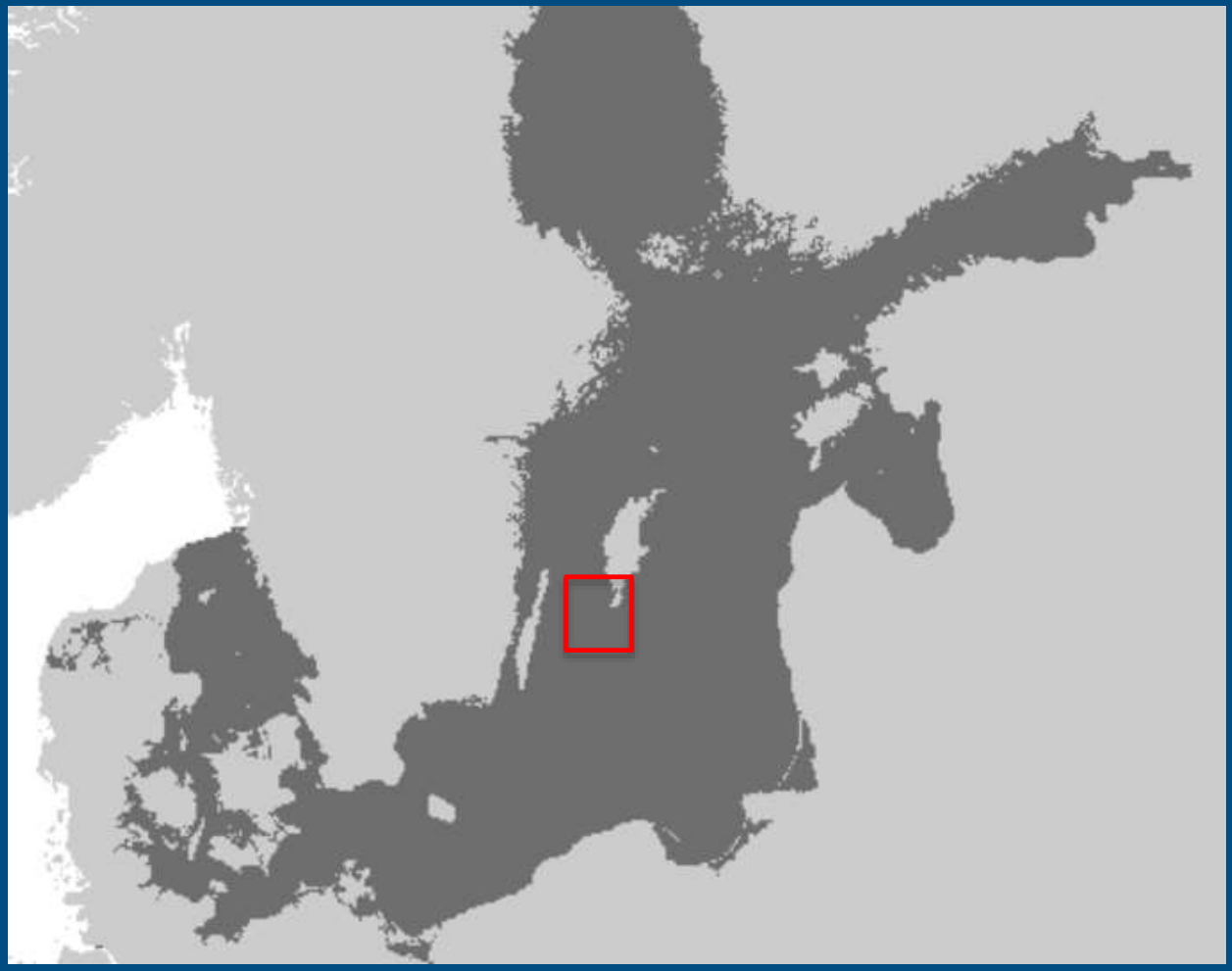
We want to overlap lines with the grid. We used a 1km x 1km grid

INSPIRE Directive compliant and dowloaded from EEA

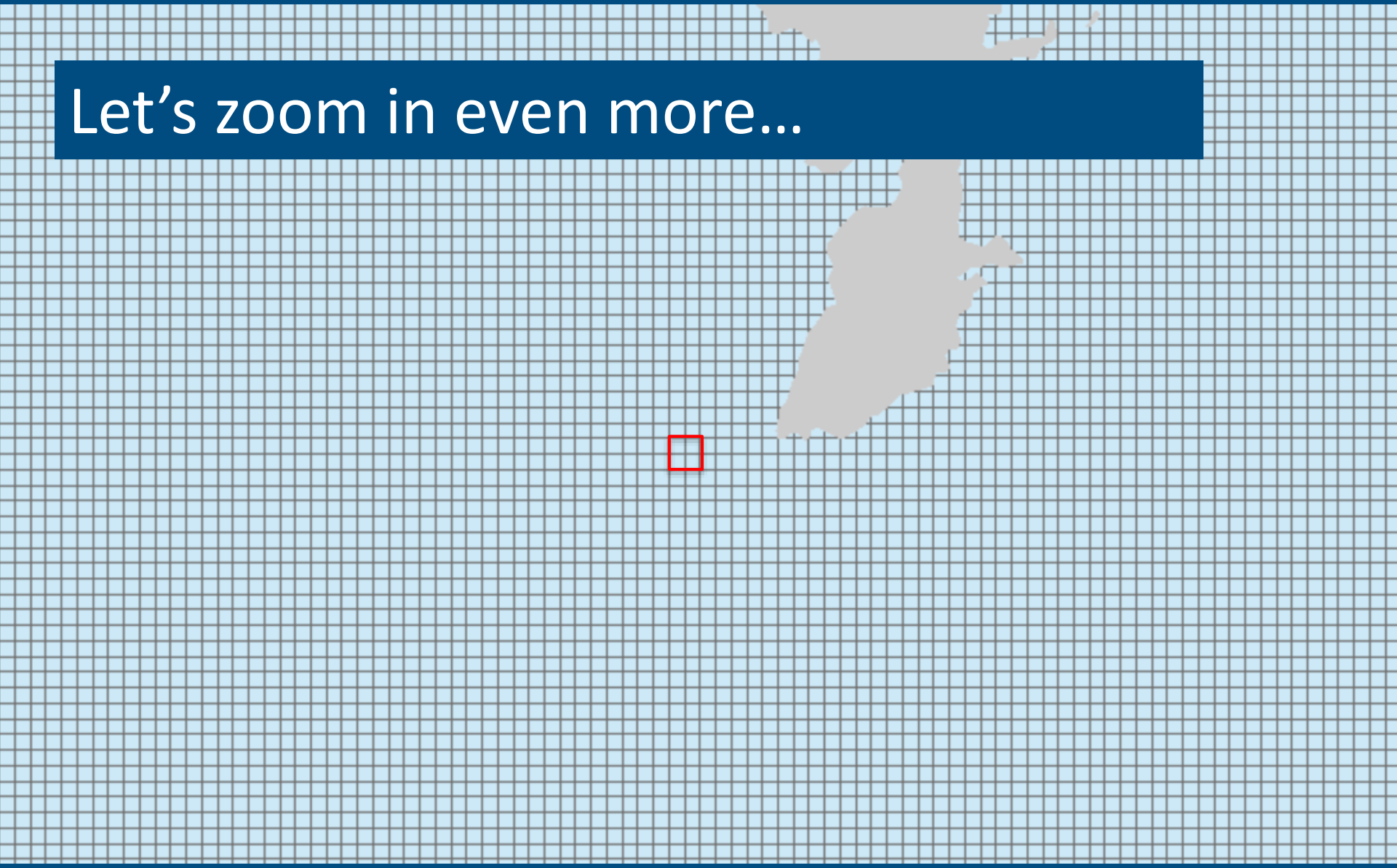


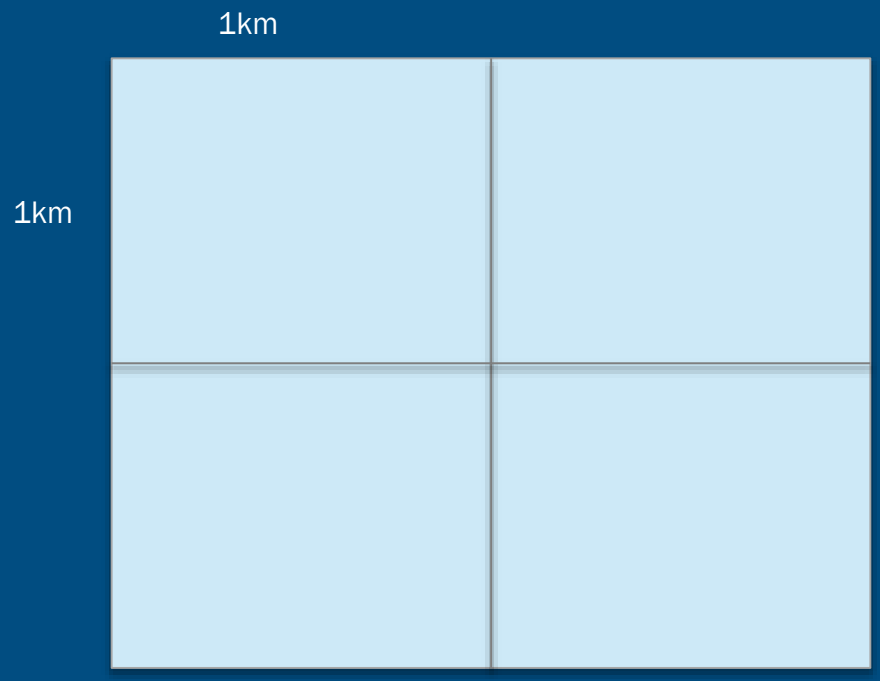


It’s so detailed that you can barely see the cells… Let’s zoom in!

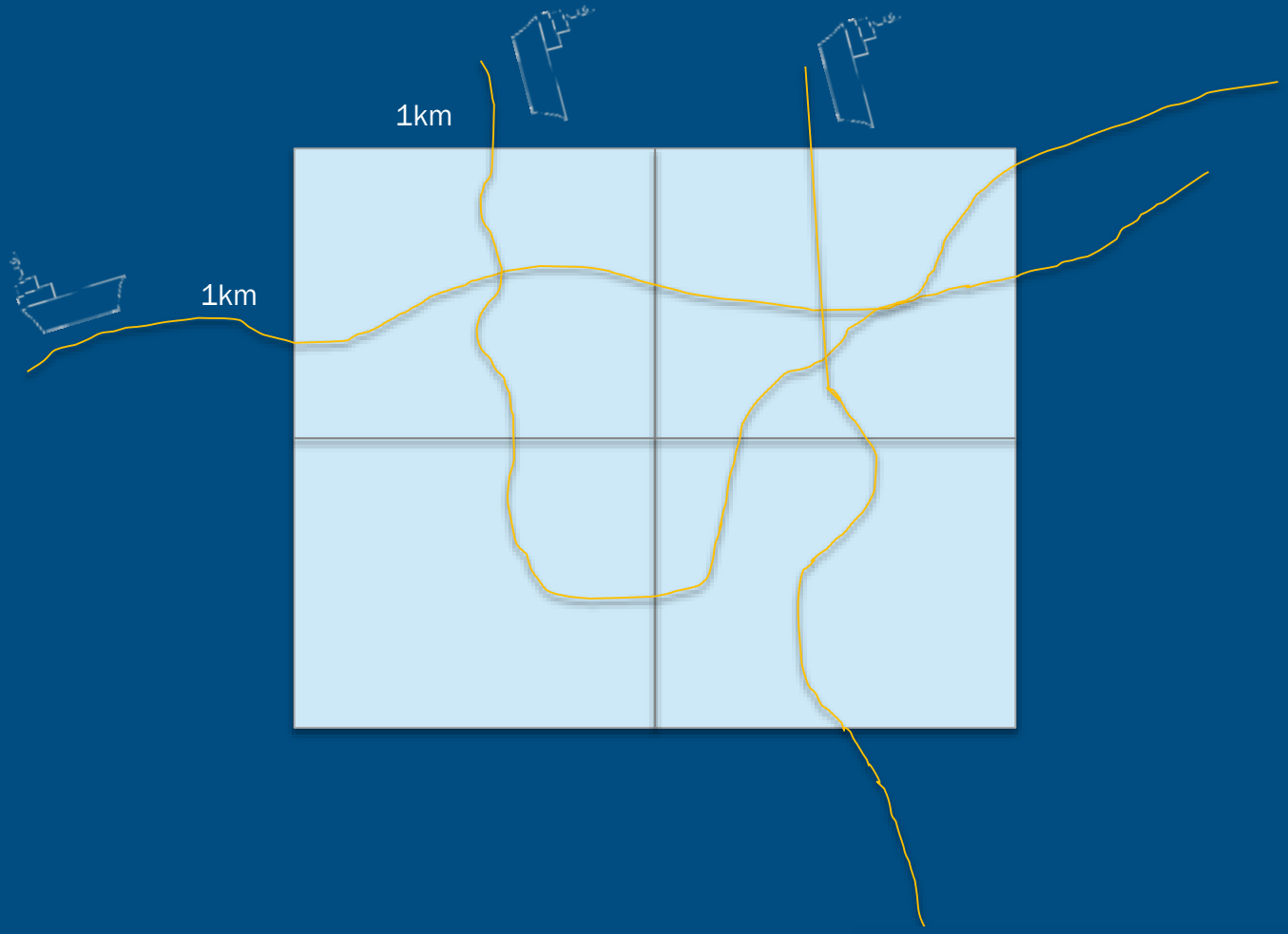


Let’s zoom in even more…

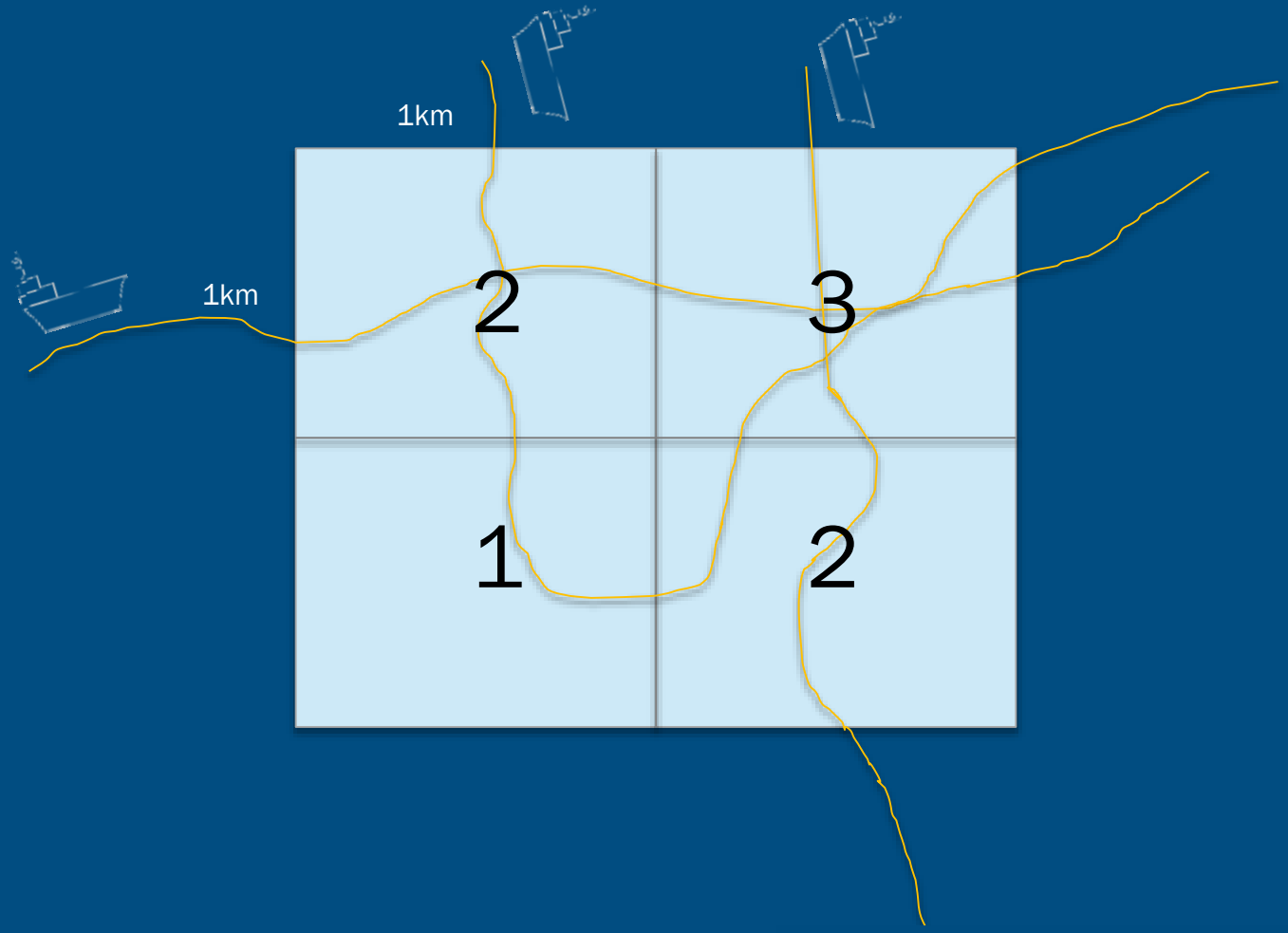




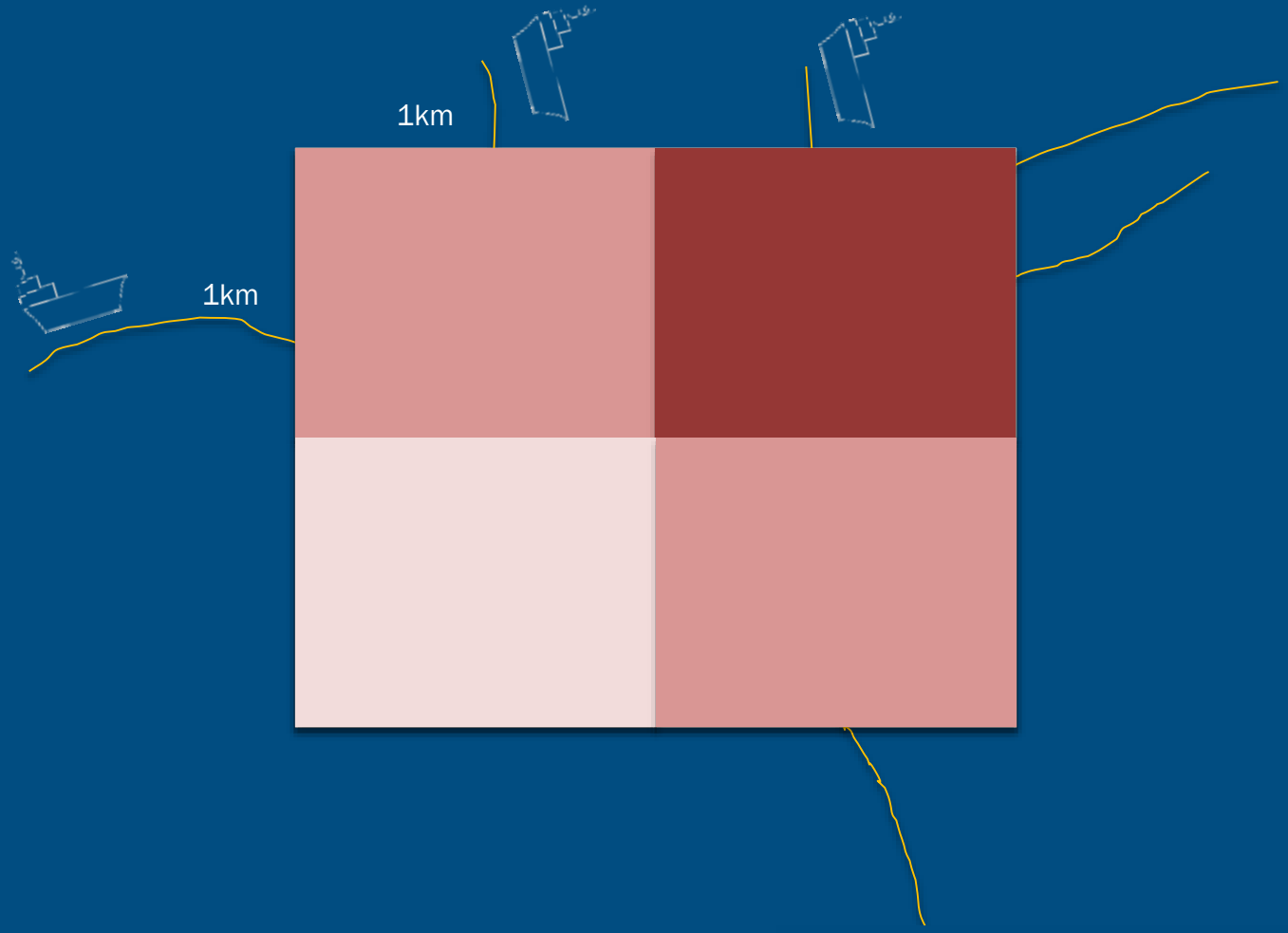
This is a sample of four cells



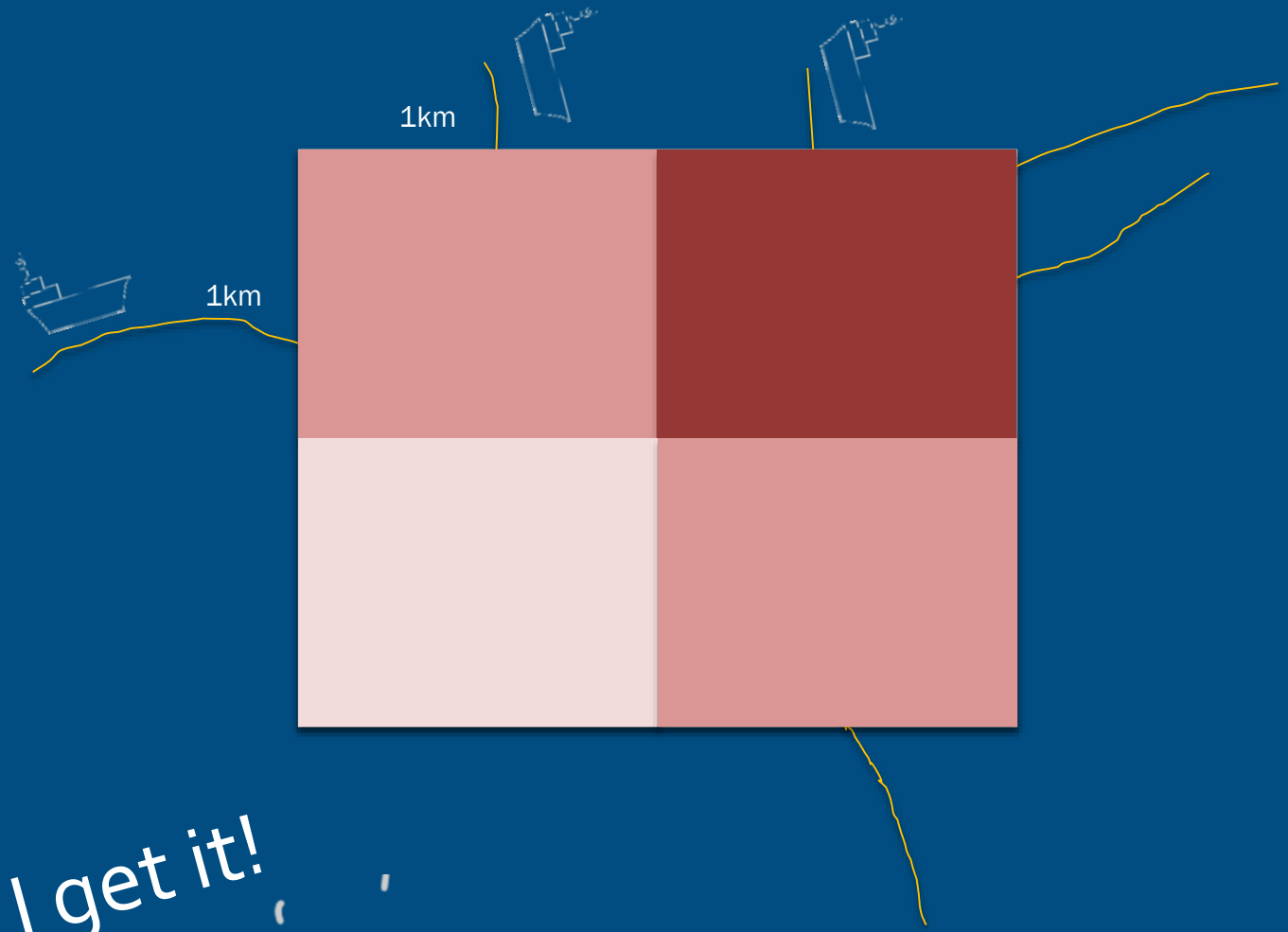
We put the lines on top



We count the lines crossing each cell



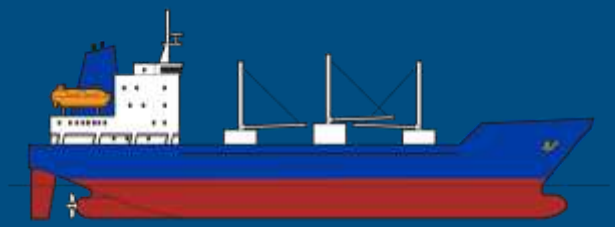
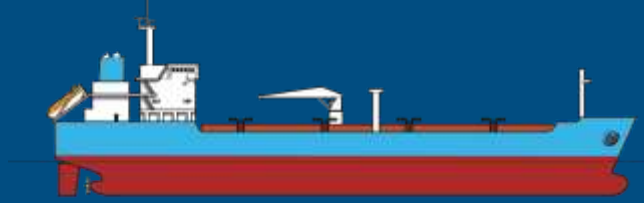
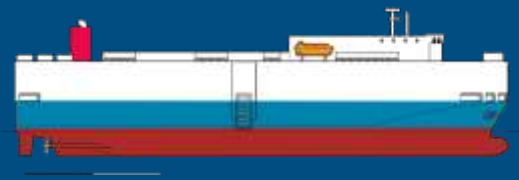
And finally we add the color gradation

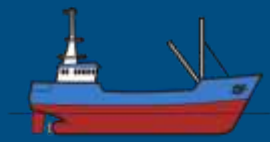
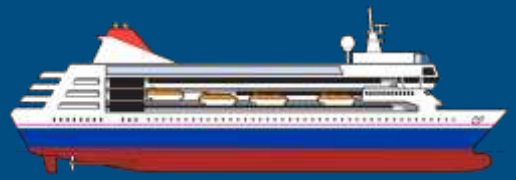
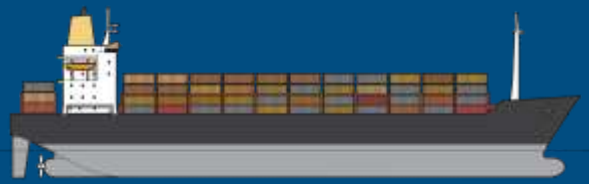


I get it!



And finally we add the color gradation

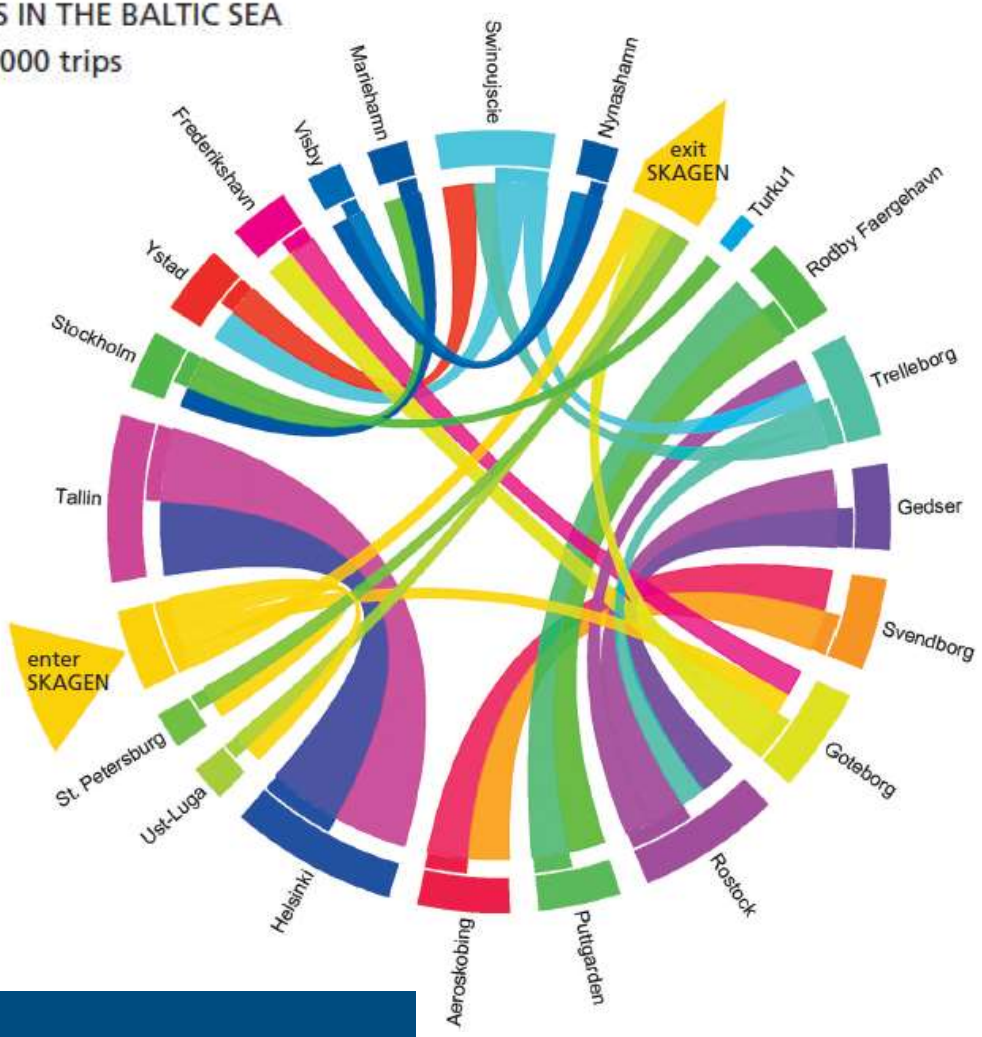






We did this for all ship types

INTERACTION BETWEEN PORTS, ARRIVALS AND DEPARTURES IN THE BALTIC SEA

All ship types traffic in 2016: 50 biggest ports, minimum of 1000 trips



With this method we can even calculate interesting statistics, for example visits to ports

```
#---------------------------------------------------------------
# Name:             Split tracks by ship type
# Purpose:
# Authors:          Manuel Frias, Andzej Milos
# Copyright:        (c) HELCOM Secretariat
# ArcGIS Version:   10.2
# Python Version:   2.7
#---------------------------------------------------------------

import arcpy
import time
from arcpy.sa import *
import os
import multiprocessing
from datetime import datetime

year = "2014"
# CARGO CONTAINER PASSENGER TANKER VEHICLECARRIERROROCARGO SERVICE FISHING OTHER
shipType = "VEHICLECARRIERROROCARGO"
grids_folder = r"E:\DensityMaps_V2\grid"
grid = r"E:\DensityMaps_V2\grid\Grid1km_BalticSea.shp"

### Smaller grid division in case you want to make a raster of a whole year
##grids_folder = r"E:\DensityMaps_V2\grid\small"
##grid = r"E:\DensityMaps_V2\grid\Grid1km_BalticSea.shp"

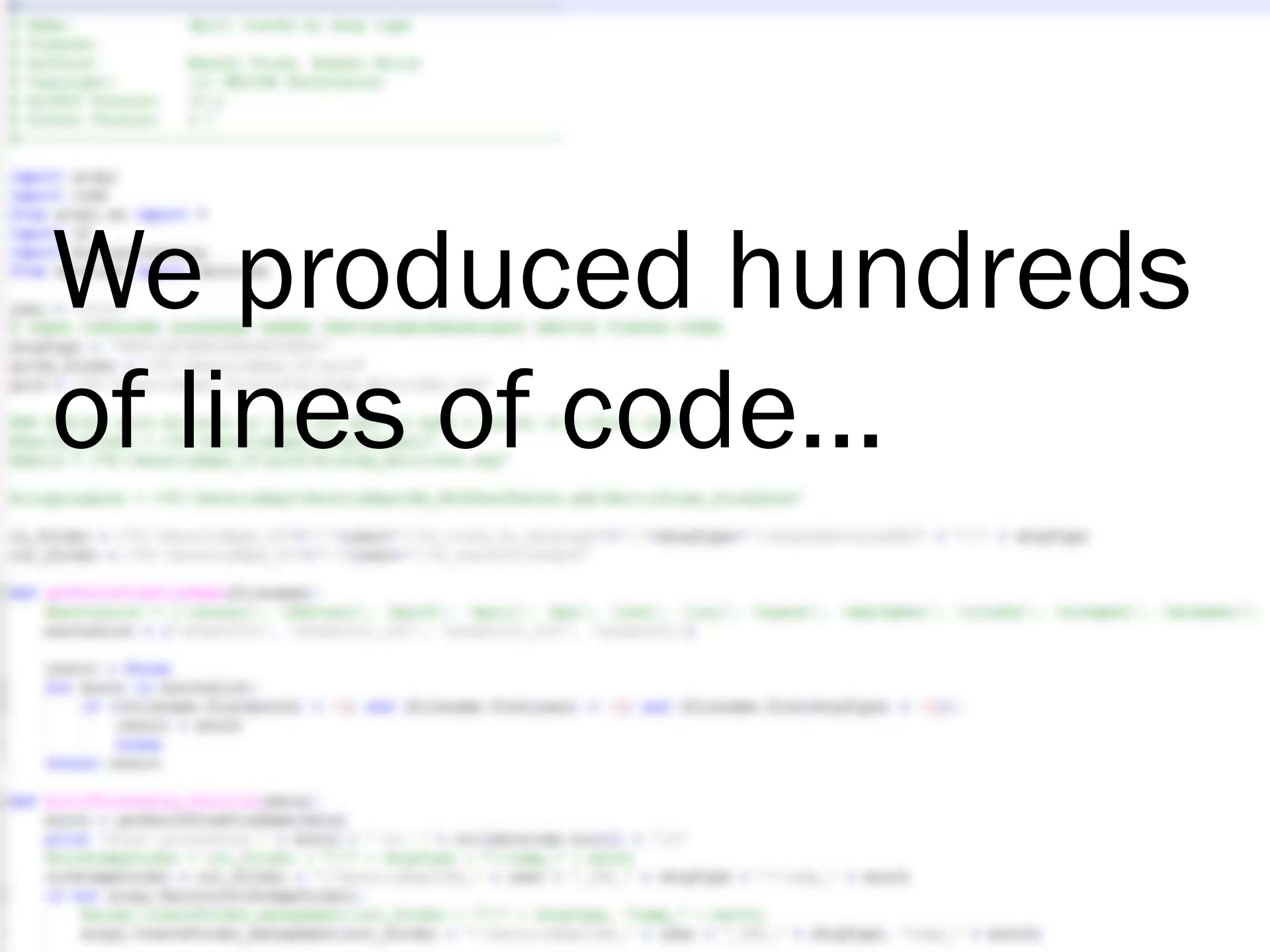
#clippingArea = r"E:\DensityMaps\DensityMaps1Km_GRIDSandTables.gdb\BalticScope_StudyArea"

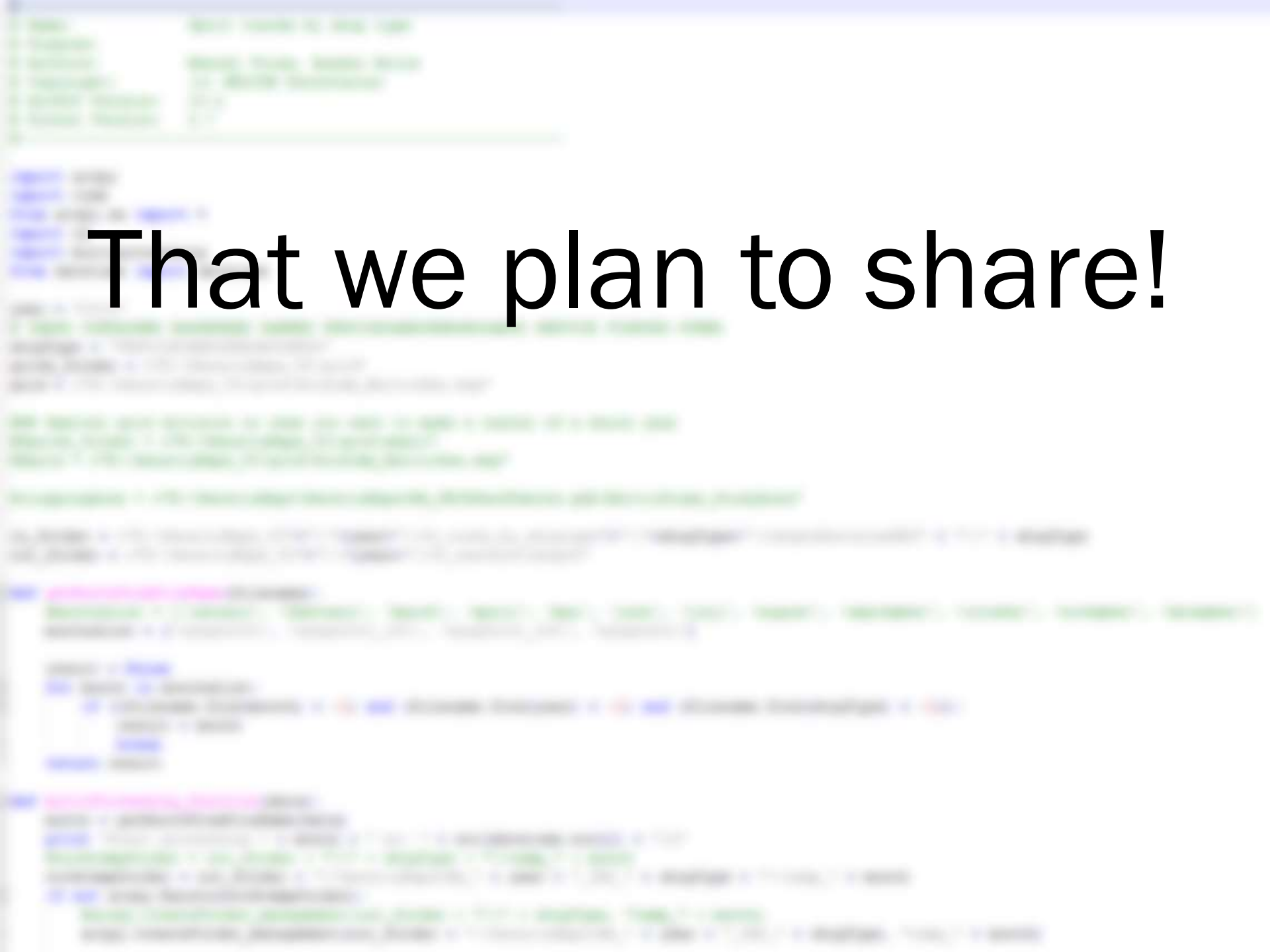
in_folder = r"E:\DensityMaps_V2"+"\\"+year+"\\03_lines_by_shiptype"+"\\"+shipType+"\\LengthEstonianEEZ" + "\\" + shipType
out_folder = r"E:\DensityMaps_V2"+"\\"+year+"\\04_rasters\Length"

def getMonthFromFileName(filename):
    #monthsList = ['january', 'february', 'march', 'april', 'may', 'june', 'july', 'august', 'september', 'october', 'november', 'december']
    monthsList = ['Length100', 'Length101_150', 'Length151_200', 'Length201']

    result = False
    for month in monthsList:
        if ((filename.find(month) > -1) and (filename.find(year) > -1) and (filename.find(shipType) > -1)):
            result = month
            break
    return result

def multiProcessing_function(data):
    month = getMonthFromFileName(data)
    print "Start processing " + month + " at: " + str(datetime.now()) + "\n"
    #worktempfolder = out_folder + "\\" + shipType + "\\temp_" + month
    worktempfolder = out_folder + "\\DensityMaps1Km_" + year + "_IMO_" + shipType + "\\temp_" + month
    if not arcpy.Exists(worktempfolder):
        #arcpy.CreateFolder_management(out_folder + "\\" + shipType, "temp_" + month)
        arcpy.CreateFolder_management(out_folder + "\\DensityMaps1Km_" + year + "_IMO_" + shipType, "temp_" + month)
```


We produced hundreds
of lines of code...


That we plan to share!



Baltic Marine Environment Protection Commission