

A background map of the Gulf of St. Lawrence and surrounding regions. The map shows the St. Lawrence River flowing into the Gulf, with labels for 'St. Lawrence River', 'Gulf of St. Lawrence', 'Newfoundland', 'Cabot Strait', 'St. Pierre Bank', 'Banquereau', 'Sable Island Bank', and 'Bay of Fundy'. The number '544' is visible near Cabot Strait, and '292' is in the bottom left corner. The map uses a light blue color for water and light green/yellow for land.

Introduction to WhaleMap

SMM22 Shiny Workshop

Overview

WhaleMap is a software tool designed to:

- Incorporate whale detection and survey effort from all survey methods in near real-time
- Allow survey teams to easily contribute and retain complete control over their data
- Provide the latest data in an accurate, user-friendly, and publicly accessible format
- Operate transparently using open-source tools and with limited supervision

WhaleMap does **NOT**:

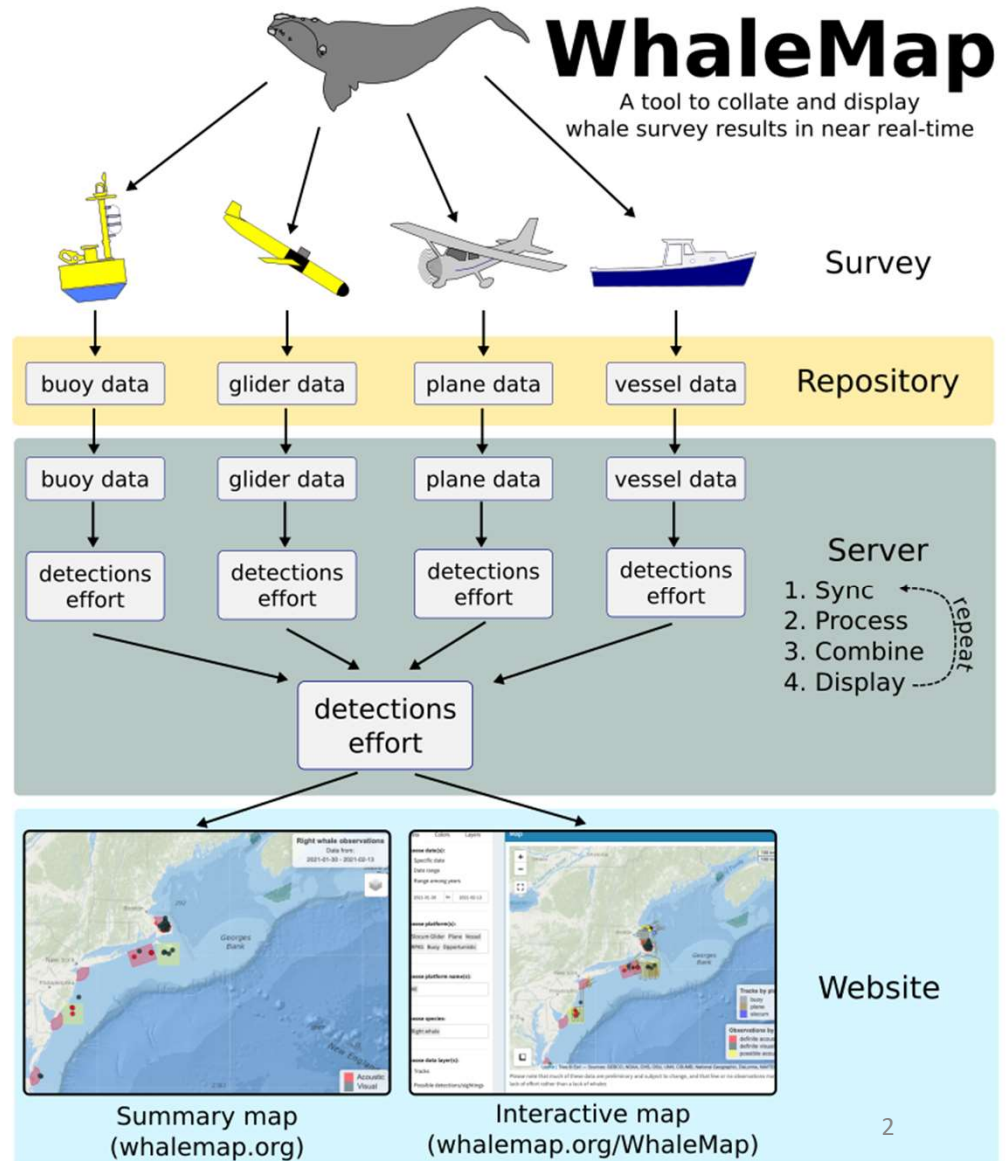
- Perform any quality-control, or take responsibility for the veracity of contributions
- Provide a long-term database for survey results
- Allow access to raw or processed data without approval from the data originator

How it works

1. Raw survey data are uploaded to a remote repository (e.g., Google Drive) shared with WhaleMap
2. Data are copied to the WhaleMap server
3. Custom code extracts detections and effort from each survey
4. Data are combined and displayed on summary and interactive maps

Johnson HD, Morrison D, Taggart CT. (2021). WhaleMap: a tool to collate and display whale survey results in near real-time. *Journal of Open Source Software*, 6(62), 3094, doi: 10.21105/joss.03094

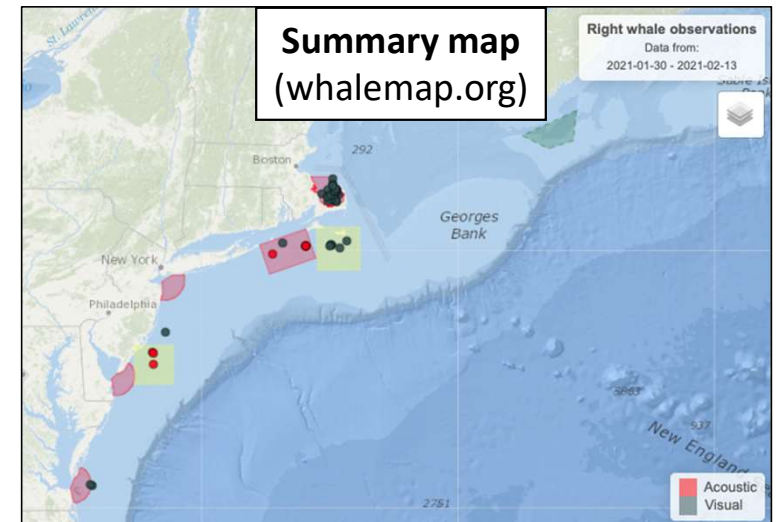
Source code: <https://github.com/hansenjohnson/WhaleMap>



Displays

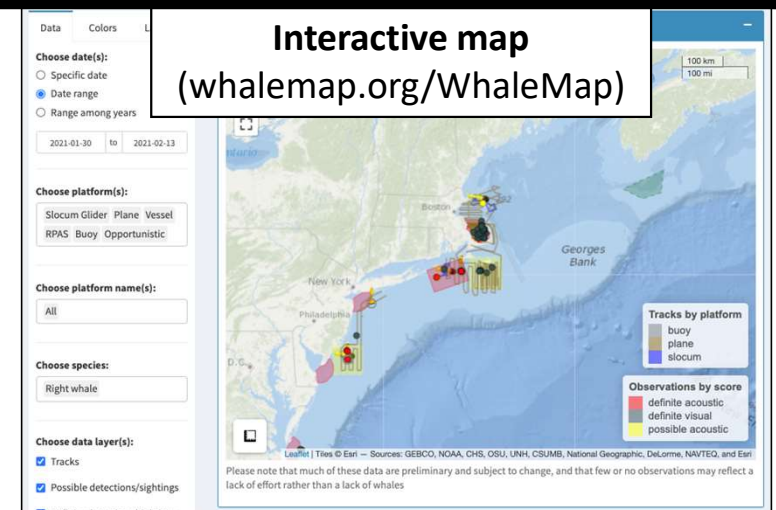
Summary map

- Available at the **whalemap.org** homepage and embedded on several other websites
- Provides a snapshot of last 14-days of right whale detections
- Easy to interpret, but limited functionality



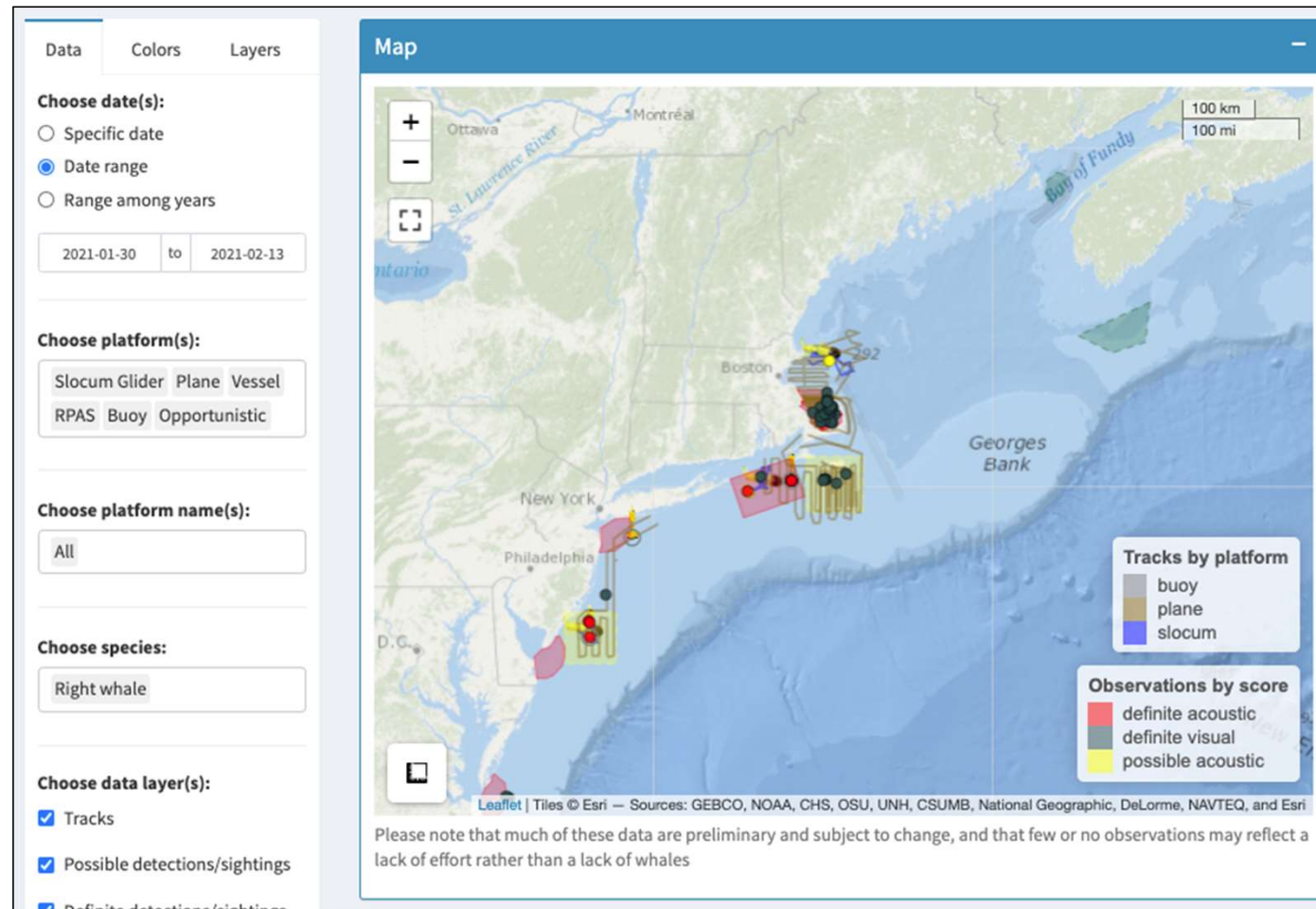
Interactive map (using Shiny!)

- Available at **whalemap.org/WhaleMap** (Or click "Interactive Map" on the WhaleMap homepage)
- Several data displays (map, timeseries, status)
- Numerous filters and layers to choose date range, species, platform, management layers, etc.



WhaleMap and Shiny

- **Shiny** allows users to interact with WhaleMap data
- App is currently live at: **whalemap.org/WhaleMap**
- Overview of:
 - Input data
 - Data filters / displays
 - Code
 - Workflow



Input data

- Observations (time, location, species, score, platform)
- Effort (time, location, platform)
- Management areas (time, location, definition)
- Platform status (time)

Updates every 15 min

Observations

| time | lat | lon | species | score | number | calves | platform | name | id | source |
|---------------------|---------|----------|----------|-----------------|--------|--------|----------|-----------------|----------------------------------|----------|
| 2022-07-26 18:44:21 | 42.2743 | -70.4776 | humpback | definite visual | 4 | NA | plane | noaa_twin_otter | 2022-07-26_plane_noaa_twin_otter | WhaleMap |
| 2022-07-26 18:37:40 | 42.4562 | -70.4643 | humpback | definite visual | 2 | NA | plane | noaa_twin_otter | 2022-07-26_plane_noaa_twin_otter | WhaleMap |
| 2022-07-26 18:36:17 | 42.4933 | -70.4609 | humpback | definite visual | 1 | NA | plane | noaa_twin_otter | 2022-07-26_plane_noaa_twin_otter | WhaleMap |
| 2022-07-26 18:31:21 | 42.6276 | -70.4544 | humpback | definite visual | 1 | NA | plane | noaa_twin_otter | 2022-07-26_plane_noaa_twin_otter | WhaleMap |
| 2022-07-26 18:30:30 | 42.6329 | -70.4522 | humpback | definite visual | 1 | NA | plane | noaa_twin_otter | 2022-07-26_plane_noaa_twin_otter | WhaleMap |
| 2022-07-26 17:10:13 | 42.9007 | -70.5931 | humpback | definite visual | 1 | NA | plane | noaa_twin_otter | 2022-07-26_plane_noaa_twin_otter | WhaleMap |
| 2022-07-26 15:51:07 | 43.1006 | -70.3770 | fin | definite visual | 1 | NA | plane | noaa_twin_otter | 2022-07-26_plane_noaa_twin_otter | WhaleMap |
| 2022-07-26 15:17:18 | 43.2005 | -70.3737 | fin | definite visual | 1 | NA | plane | noaa_twin_otter | 2022-07-26_plane_noaa_twin_otter | WhaleMap |
| 2022-07-26 14:17:56 | 42.6431 | -70.5270 | humpback | definite visual | 2 | 1 | plane | noaa_twin_otter | 2022-07-26_plane_noaa_twin_otter | WhaleMap |
| 2022-07-20 19:06:25 | 42.6005 | -70.2850 | humpback | definite visual | 1 | NA | plane | noaa_twin_otter | 2022-07-20_plane_noaa_twin_otter | WhaleMap |

Filters: data

- Date range
- Platforms
- Species
- Data sources
- Data layers
- Password protected data

Filters: data

Data Colors Layers

Choose date(s):

☐ Specific date

☒ Date range

☐ Range among years

2022-07-13 to 2022-07-27

Choose platform(s):

Slocum Glider Plane Vessel RPAS Buoy

Opportunistic

Choose species:

Right whale

Choose data source(s):

NARWC WhaleMap WhaleInsight RWSAS

Choose data layer(s):

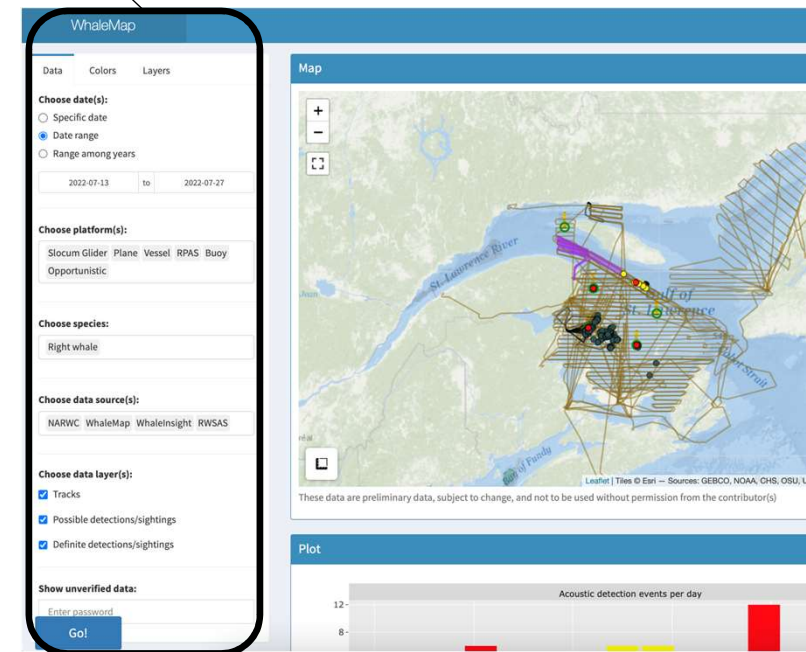
☒ Tracks

☒ Possible detections/sightings

☒ Definite detections/sightings

Show unverified data:

Enter password



Filters: colors

- Basemap
- Observation variable
- Observation palette
- Effort variable
- Effort palette

Data Colors Layers

Choose basemap:

ESRI Ocean

Color observations by:

Score

Choose color palette:

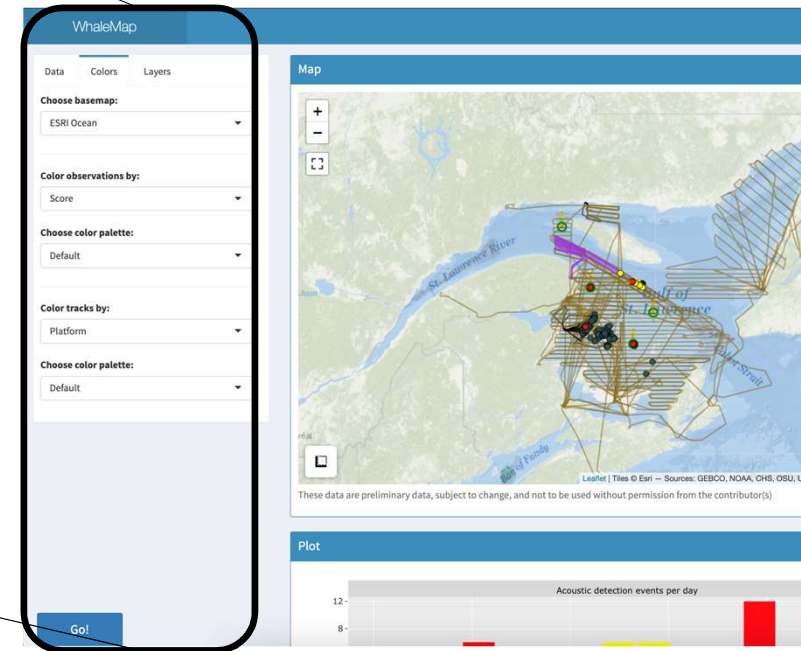
Default

Color tracks by:

Platform

Choose color palette:

Default



Filters: layers

- Misc map layers
- Canadian management layers
- US management layers

DataColorsLayers

Map layers

☐ Graticules

☐ NOAA charts

☒ Latest robot positions

☒ Shipping lanes

☒ Legends

Canadian management areas (2021)

☒ Critical habitat areas

☐ Management Grid

☐ Fishing management areas i

☐ Fishing depth contours (10 and 20 fathom) i

☐ Shipping management areas i

☐ Transport Canada restricted area i

US management areas

☐ Lobster management areas i

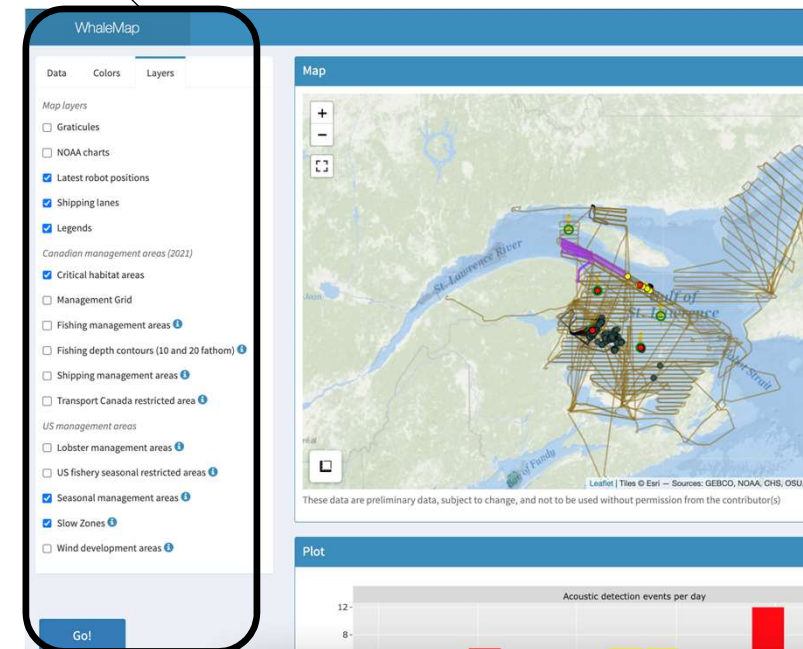
☐ US fishery seasonal restricted areas i

☒ Seasonal management areas i

☒ Slow Zones i

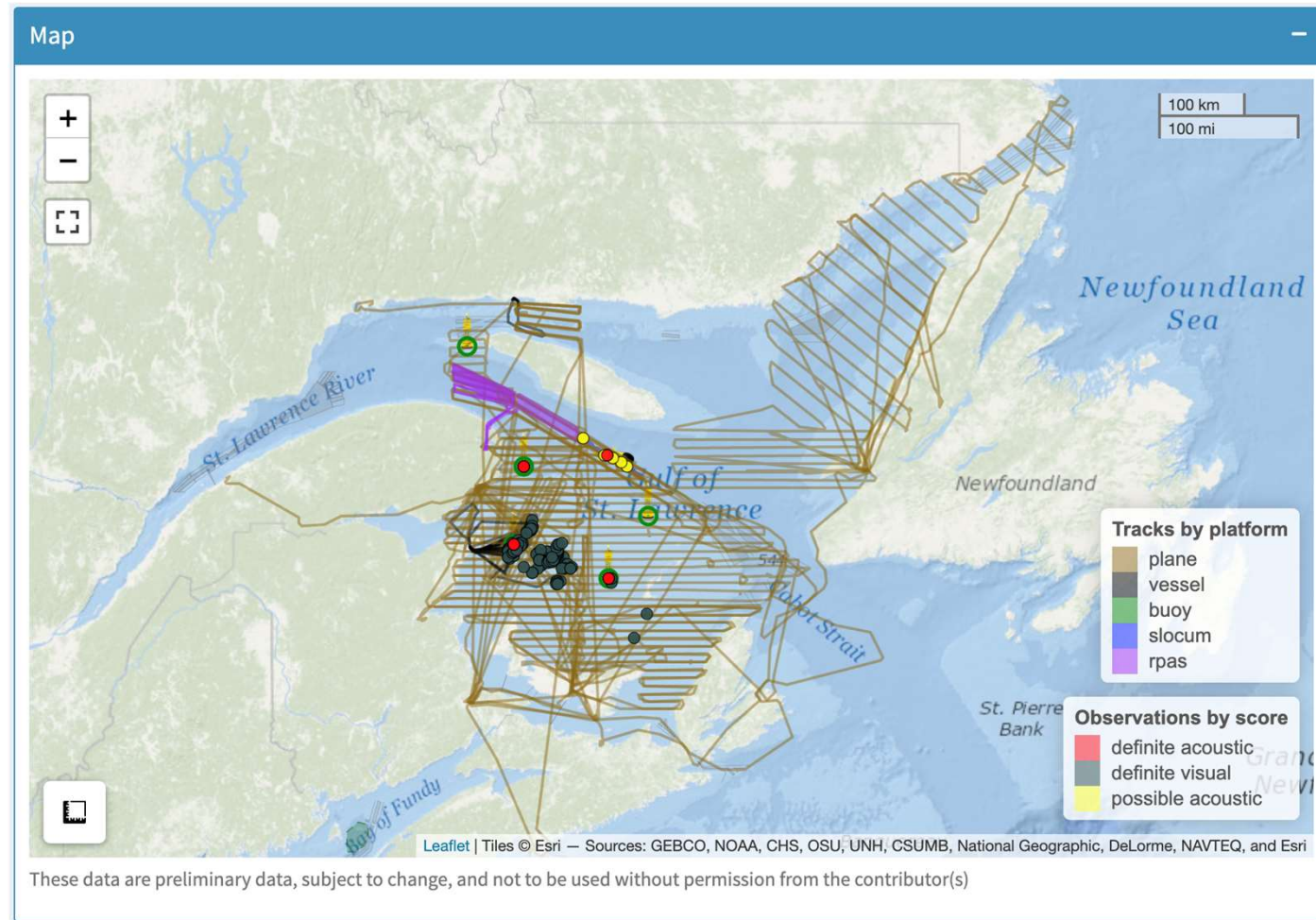
☐ Wind development areas i

Go!



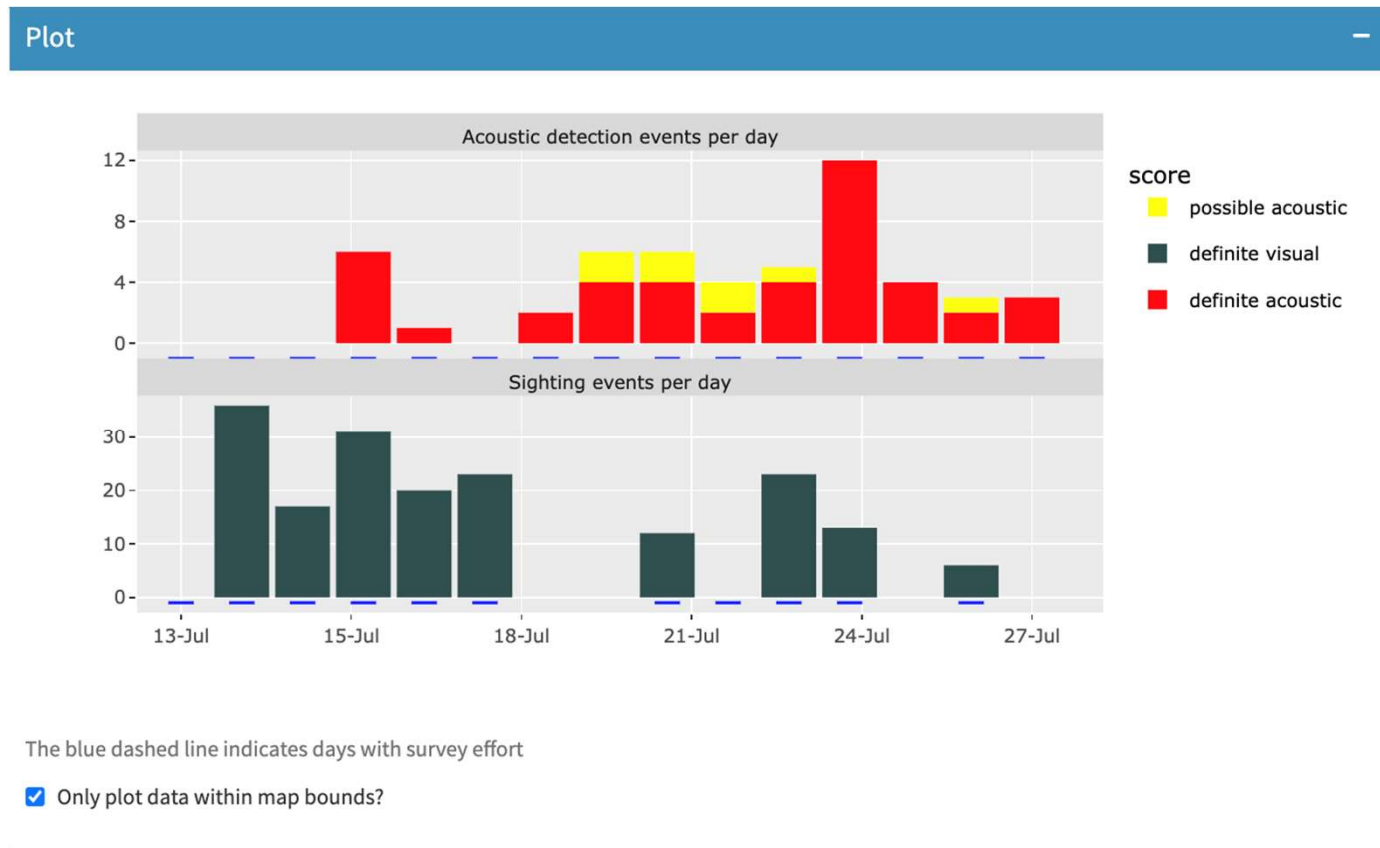
Displays: map

- Interactive map built using leaflet
- Displays observations, effort, and various spatial data layers
- Zoom in/out, measure distances, clickable layers, customize layer colouring



Displays: plot

- Interactive plot built using ggplot and plotly
- Displays daily timeseries of acoustic and visual detections
- Indicates daily presence of effort
- Select data in map bounds (or not)
- Numerous plotly widget features



Displays: summary

- Simple text output with summary statistics
- Restricted to viewing area in map

Currently viewing:

Species: right

Number of definite sighting events: 181

Number of whales sighted (includes duplicates): 207

Number of possible sighting events: 0

Number of whales possibly sighted: 0

Number of definite detections: 44

Number of possible detections: 8

Earliest observation: 2022-07-14

Most recent observation: 2022-07-27

Most recent position: 48.5833, -63.8833

Displays: status

- Provides a list of data providers and links for additional information
- Shows when the data from each platform were processed last
- Indicates if there's an error in processing

| Status: | |
|---|----------------------|
| Platform | Last processed [UTC] |
| Canadian observations/tracks | 2022-07-27 09:15:23 |
| WHOI acoustic detections | 2022-07-26 18:01:26 |
| NOAA NEFSC aerial survey sightings/tracks | 2022-07-26 17:00:37 |
| New England Aquarium vessel survey sightings/tracks | 2022-07-18 22:45:27 |
| New England Aquarium aerial survey sightings/tracks | 2022-07-16 12:45:30 |
| Center for Coastal Studies aerial surveys | 2022-07-12 09:13:50 |
| NOAA NEFSC vessel survey sightings/tracks | 2022-05-10 12:46:09 |
| New England Aquarium SNE vessel survey sightings/tracks | 2022-04-27 13:06:30 |
| Southeast US aerial surveys | 2022-04-18 08:03:44 |

This shows when data from a particular platform were last processed by the WhaleMap system. Errors in processing are indicated by an error message in place of a timestamp.

Code: ui.R

Defines the user interface for the app

<https://github.com/hansenjohnson/WhaleMap/blob/master/ui.R>

```
# ui.R
# WhaleMap - a Shiny app for visualizing whale survey data

# header -----

header <- dashboardHeader(title = 'WhaleMap',

  # data
  dropdownMenu(
    type = "notifications",
    icon = 'More Information',
    badgeStatus = NULL,
    headerText = "",
    notificationItem("Cite",
      icon = icon('education', lib = 'glyphicon'),
      href = "https://whalemap.org/#cite"),
    notificationItem("Contact",
      icon = icon('envelope', lib = 'glyphicon'),
      href = "https://whalemap.org/#contact"),
    notificationItem("Code",
      icon = icon('console', lib = 'glyphicon'),
      href = "https://github.com/hansenjohnson/WhaleMap"),
```

Code: server.R

Defines the data processing/plotting for the app

<https://github.com/hansenjohnson/WhaleMap/blob/master/server.R>

```
# server.R
# WhaleMap - a Shiny app for visualizing whale survey data

function(input, output, session){

  # read in data -----

  # tracklines
  tracks = readRDS('data/processed/effort.rds')

  # latest dcs positions
  lfile = 'data/processed/dcs_live_latest_position.rds'
  if(file.exists(lfile)){
    latest = readRDS(lfile)
  }

  # sightings / detections
  observations = readRDS('data/processed/observations.rds')

  # dynamic map polygons
  load('data/processed/dma.rda')
  load('data/processed/sma.rda')
```

Code: global.R

Conveniently define objects that are used in both the UI and server scripts

<https://github.com/hansenjohnson/WhaleMap/blob/master/global.R>

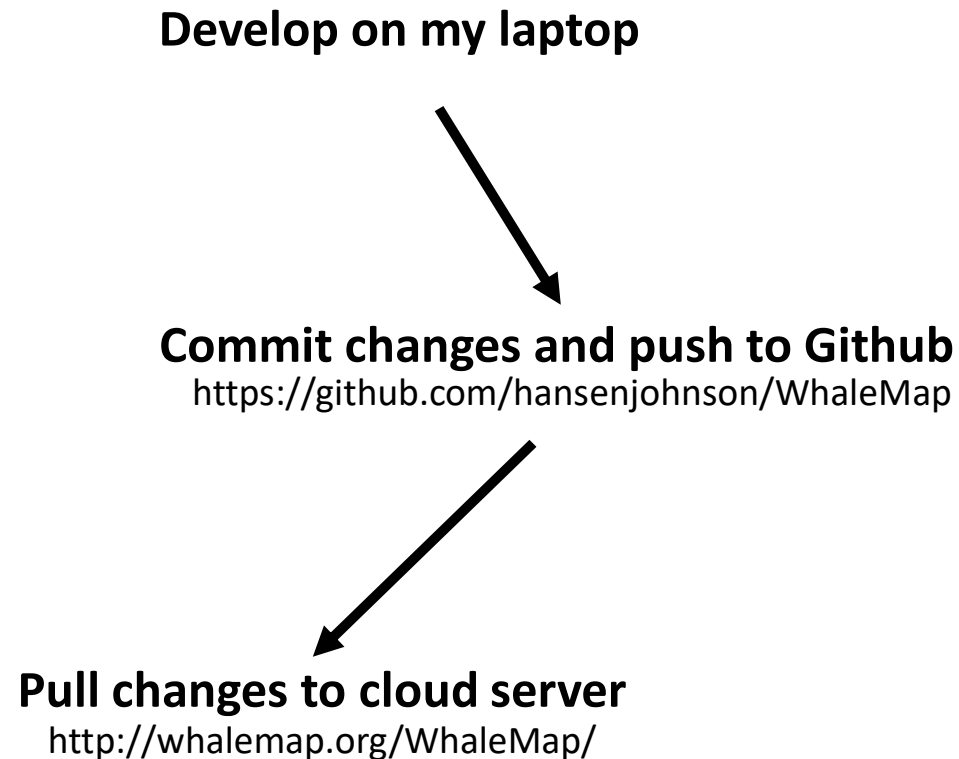
```
# global.R|
# WhaleMap - a Shiny app for visualizing whale survey data

# setup -----

suppressPackageStartupMessages(library(shiny))
suppressPackageStartupMessages(library(leaflet))
suppressPackageStartupMessages(library(rgdal))
suppressPackageStartupMessages(library(htmltools))
suppressPackageStartupMessages(library(htmlwidgets))
suppressPackageStartupMessages(library(maptools))
suppressPackageStartupMessages(library(lubridate))
suppressPackageStartupMessages(library(oce))
suppressPackageStartupMessages(library(shinydashboard))
suppressPackageStartupMessages(library(ggplot2))
suppressPackageStartupMessages(library(plotly))
suppressPackageStartupMessages(library(leaflet.extras))
suppressPackageStartupMessages(library(shinybusy))
source('R/functions.R')
```

Workflow

- Interactive development on my laptop using Rstudio
- All changes tracked using git and github
 - Review all changes ever (>1200!)
 - Easily revert if something breaks
 - Share code across machines
 - Collaborate with anyone
- Live app is hosted on a cloud server (Linux virtual machine on Azure cloud) running free Shiny server





Questions?

Contact me:

hansen.johnson@dal.ca

More information:

WhaleMap publication: <https://joss.theoj.org/papers/10.21105/joss.03094>

Thank you!