AIS Indexer Development Report

Dan Radulescu

Prepared by:

OODA Technologies Inc.
4891 Av. Grosvenor, Montreal Qc, H3W 2M2

Project Manager: Anthony W. Isenor
Contract Number: W7707-115137, Call Up 6, 4500959431
Contract Scientific Authority: Anna-Liesa S. Lapinski

Defence R&D Canada – Atlantic

Contract Report

DRDC-RDDC-2014-C27
March 2014

Approved by

_____

Approved for release by

_____

Abstract

A Coverage Index Generator Java application was developed in the summer of 2011, based on the work of Lapinski and Isenor (Estimating Reception Coverage Characteristics of AIS, Lapinski and Isenor, Journal of Navigation, October 2011). This application gathers Automatic Identification System (AIS) messages and produces a coverage map, graphically representing the quality and fidelity of signals received by coastal sensors. The application gathers its inputs either from static data files or a streaming input source. During initial development, problems were identified with the streaming option that could not be fixed at the time. OODA Technologies improved the application in 2012 under call-up 5 [1] to the standing offer W7707-115137. Several bugs were fixed and performance was improved with the streaming option.

In this work, the Coverage Index Generator was made more user-friendly and a ship monitoring feature as well as a publishing feature interfacing with the GeoNetwork framework were added. The resulting application, called the AIS Indexer, is a flexible tool that visualizes spatial-temporal gaps of received AIS messages and exploits that information in a defined Area Of Interest (AOI).

This page intentionally left blank.

Table of contents

List of figures

1 Introduction

The goal of this project was to build on the Coverage Index Generator started in the summer of 2011 [2, 3], making it more user-friendly and adding a ship monitoring feature as well as a publishing feature interfacing with the GeoNetwork framework.

The AIS Indexer is the result of this effort, a tool that visualizes spatial-temporal gaps of received AIS messages and exploits that information in a defined AOI. This application allows the user to generate maps visualizing signal fidelity using live AIS messages or local pre-parsed reports. All generated maps are automatically saved and can be reloaded at any time.

The following sections detail the application structure and its features.

2 New Application Structure

To facilitate the redesign effort, it became apparent during the planning phase that some amount of restructuring would be required both in terms of the way the user interacts with the application (the front-end) as well as what happens behind the scenes (back-end).

2.1 Front-end

The user interface (UI) has been completely redesigned. A single window frame greets the user at the start of the application which is referred to as *Main Panel*. It offers only two alternatives, to either open an existing map or generate a new map. From then on, options become visible based on the user's choice. The goal of this frame is to provide the user with a single, intuitive way to start using the maps while maintaining the simplicity of navigation. The specifics of every available choice are covered in detail in the AIS Indexer User Guide [4] and will not be repeated here.

Whenever a map is generated or an existing map opened, a new frame appears showing the details and interaction options specific to that map. This frame is referred to as the *Map Viewer*. Several map viewers can be active in the same time, each one containing information pertaining to a single map. In other words, it is a core functionality that the user be able to operate on several maps at once as shown in Figure 1.
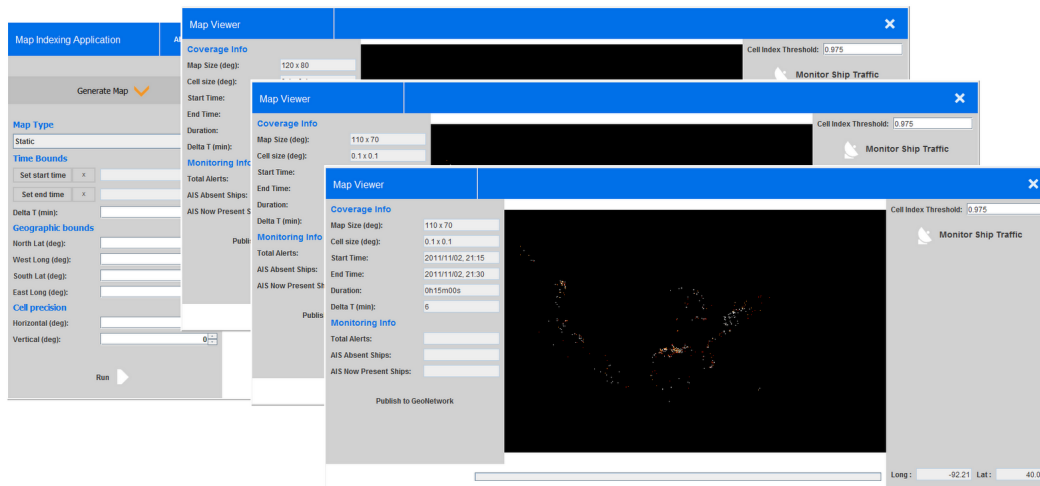

Figure 1: *This screenshot illustrates three different Map Viewers active at the same time.*

2.2 Back-end

The application structure has been modified to fit into the Model View Presenter (MVP) design pattern but not in a very strict sense. Since a good portion of the application relies

on computationally heavy algorithms that didn't fit well with the MVP pattern, this section of the code was decoupled from the design pattern.
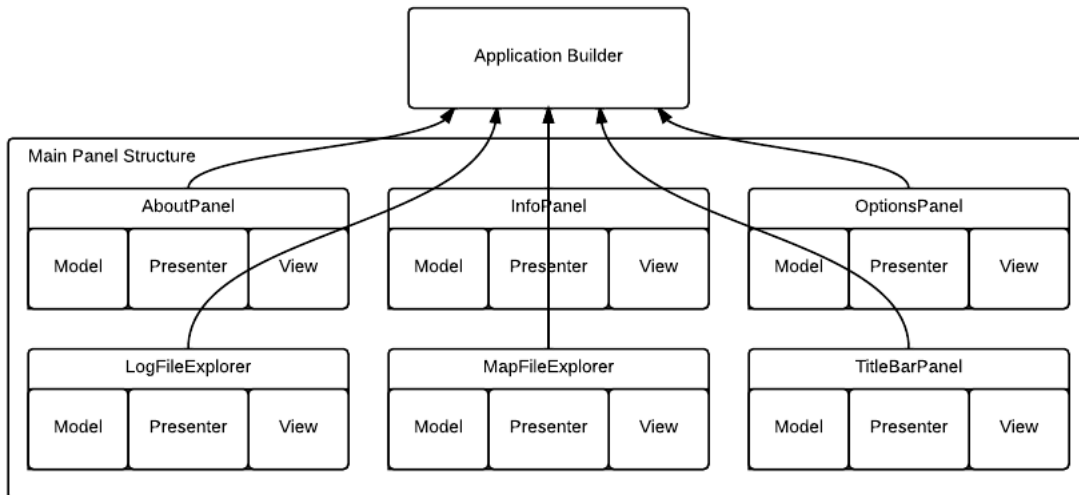

Figure 2: *This figure shows the object structure of the Main Panel.*

As shown in Figure 2, all components that are part of the Main Panel are connected to a single class entity called the Application Builder which serves as a routing point for sharing information between entities within the Main Panel. From the Main Panel, the user can generate maps. Every time a new map is generated, the group of objects associated to that map are completely self sufficient and isolated from the rest of the application. The main components of each map are the raw data, settings, an analysis structure and a ship monitor. All these components are managed by a single controller. These relationships are shown in Figure 3.

The Coverage Builder is responsible for defining the map structure. It can create three different types of map structures based on the user's request. The map types are:

– static (created from pre-parsed AIS messages stored on the local machine)
– live (created from a live feed of AIS data streamed into the application)
– loaded (from map data stored as a file on the local file system)

All these map types share most of the same internal structure with slight differences in their controller and analysis. There is no imposed limit to the number of maps that can be simultaneously opened/created, however hardware limitations will cause a lack of responsiveness when memory runs low. The user is encouraged to monitor processor and memory load when running multiple maps.
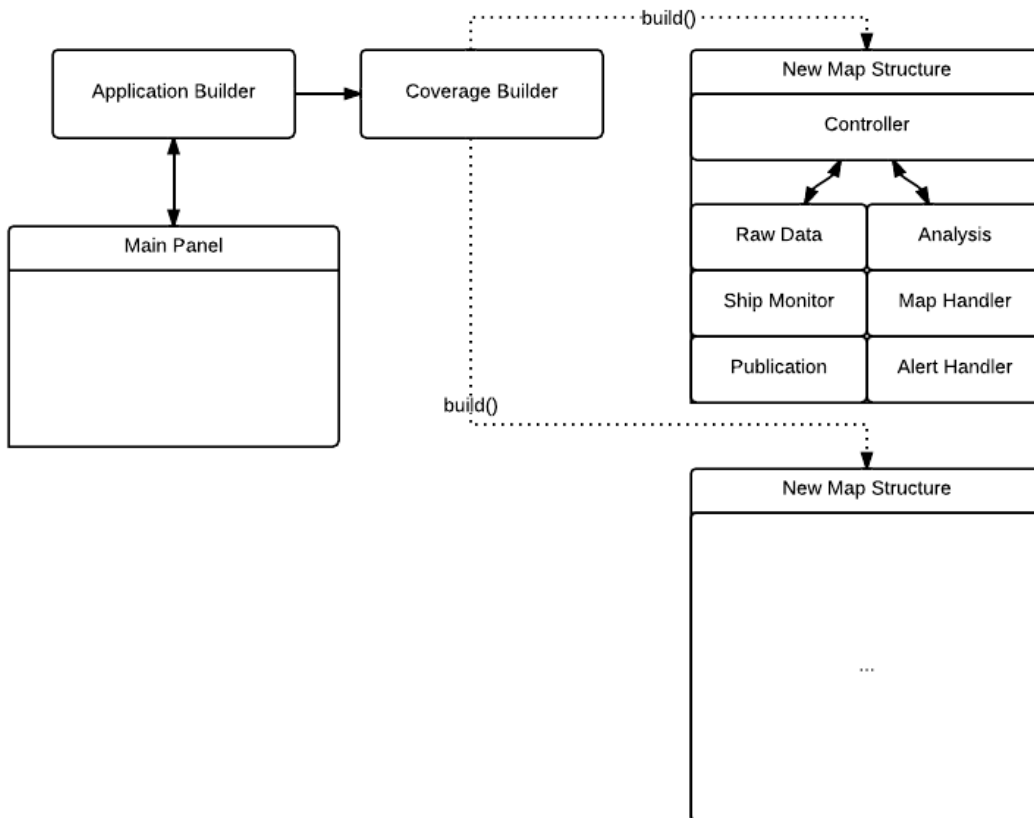
Figure 3: *This figure shows the internal structure of map generation as well as the information flow.*

3   New Features

The following sections detail key features that were added since the previous version of the application.

3.1   Fully Customizable Map Generation

Although the previous version of the application included customizable map generation, it was cumbersome to use, and in the case of static maps, not fully working. Generating new maps has been streamlined and simplified. No longer does the user have to navigate multiple panels in order generate a map. Geographic, time bounds and cell size parameters all appear in the same panel and have been tailored for ease of use.

3.2   Map Navigation

New map navigation tools allow the user to zoom in and out of a map and drag the mouse around to move the map. This control scheme should be very familiar to anyone that has used Google Maps or any other similar tool.

The Map Viewer also features a location indicator in the bottom right corner of the panel. As seen in the screenshot below (Figure 4), the indicator displays the coordinates in longitude and latitude of the cursor's position on the map.

Figure 4: *A close-up screenshot of the location indicator.*

3.3   Save and Open Maps

The AIS Indexer automatically saves any generated map to the local file system in the `Saved Maps` folder. The maps are stored both as a text and png, although only text files can be read by the AIS Indexer. The header of every map text file contains information on its characteristics. The exact format of saved maps is presented in the annex along with an example.

Opening a map is as simple as navigating to the desired map text file and clicking a button.

3.4 Ship Monitoring

Ship monitoring is a process that can be assigned to any map, whatever the type (loaded, static or live), the goal of which is to supervise how ships report their positions relative to a cell's signal quality. In simple terms, a ship that has reported from a cell with a strong index (representing good historic signal transmission) will be expected to report again. If it fails to do so in a specified time period, an alarm is triggered. The threshold of what is deemed to be a strong index can be set by the user but a default value is set to 0.975 (out of 1.0). Conversely, a ship that passes through a cell with a poor signal index will have a lower expectation of reporting again and will not be monitored.

The monitoring system is expected to identify erratic ship reporting behaviour, for example the case of a ship that fails to report from an area with a strong signal index, or the case of a missing ship that reappears.

The main component of the monitoring system is the list of monitored ships. This list contains all ships that are expected to send a report within a specific amount of time. The actual time to report is determined by the variable delta T which is also used by the indexing algorithm. More specifically, it is 2 x delta T with a default value for delta T of 6 min. An asynchronous process goes through each ship in the list and decrements their timer by one second, every second. If any ship's timer should countdown to zero, before the ship has had a chance to report, an alert will be recorded and the ship is marked as missing. In other words, a monitored ship should report within a period of 2 x delta T or it is marked as missing. The flowchart for this process is described in the Figure 5.

The algorithm in Figure 6 runs in parallel with the one mentioned in Figure 5 and determines when ships are added or removed from the monitoring list. Every AIS report received by the socket is processed by this algorithm. Looking at the terminal boxes marked 1 through 4 is the best way to understand the basics of the algorithm. The 5th box is a special case that deals with reappearing ships.

1. In order to add a ship to the monitoring list, the report has to originate within the geographic area of interest (AOI); the cell where the report originated must have an index above the specified threshold (0.975 by default); and the ship must not be monitored already.

2. In this case, the report originated within the AOI, from a cell with an index above the threshold but the ship is already being monitored. The countdown timer associated with this ship is reset in this case (set back to 2 x delta T).

3. There are two ways for a ship to be un-monitored. If a monitored ship sends a report originating within the AOI, but from a cell whose index is below the threshold, the ship will be removed from the monitoring list.

4. Another way to un-monitor a ship is the case where a monitored ship reports from outside the AOI. Any ship exiting the AOI is therefore ignored.

6
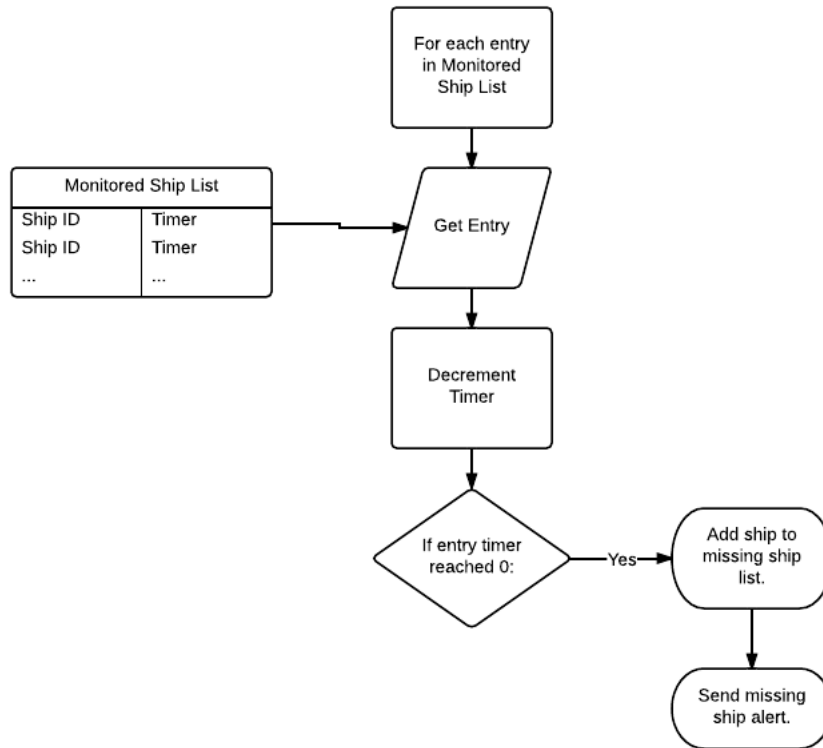
Figure 5: *Flowchart showing the handing of the countdown timer on each monitored ship.*

5. This case dictates that whenever a missing ship reappears, whether inside the AOI or not, the ship is marked as *AIS now present*. The algorithm then continues normally as with any report.

3.5 Publishing

Publishing a map refers to the transmission of a map's image and characteristics to the GeoNetwork website. Any map can be uploaded with the click of a button. Publication is described in further detail in [5].

7

Figure 6: *Flowchart of the process which determines which ships are monitored.*
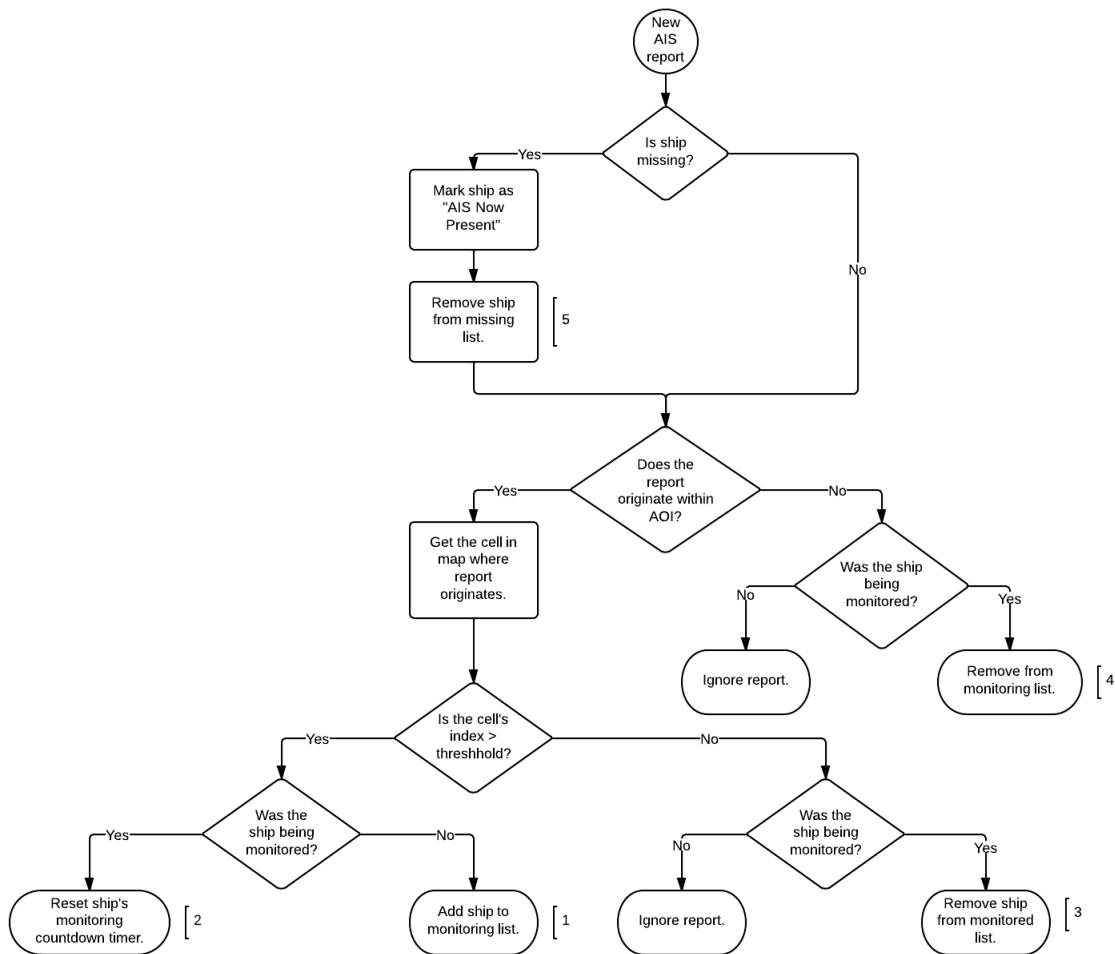
4   Way Ahead

This section describes possible additions to the application that could improve usability.

4.1   World Map Outline

Currently, when maps are generated, reports appear as colored rectangles or squares. If the map resolution is high enough and the reports come in dense enough clusters, it is possible to infer the shapes of shorelines.

It would be possible however to facilitate the visualization of maps by drawing continental shorelines as a separate layer on the map. This feature could take as much as 3 weeks of development.

4.2   Map Grid

Another improvement benefiting the visualization of maps would be to overlay a coordinate grid on top of maps. This feature would allow the user to visually approximate distances between objects on the map and would convey a more accurate sense of scale. This feature could be completed in 1-2 weeks.

4.3   Database for Ships and Alerts

A database structure would enable advanced analysis of reports and alerts. For example, ship tracking could be implemented by looking at previous reports and the distance between their emission. Such an analysis could also be used to predict a ship's path or reduce the amount of false missing ship alarms by determining when a ship will exit the AOI. Processes could also be implemented to mine alert data for pattern discovery. Implementing such a feature would require several months of development.

References

[1] D., Radulescu (2012), AIS Coverage Estimate - Completion of Java Program for Computing AIS Index.

[2] Lapinski, Anna-Liesa S. and (2011), Anthony W. Isenor (2011), Estimating Reception Coverage Characteristics of AIS, *Journal of Navigation*, pp. 609–623.

[3] Lapinski, A.-L. S. and Isenor, A.W (2010), Mapping AIS coverage for trusted surveillance, *In Proceedings of Unmanned/Unattended Sensors and Sensor Networks, Proc. SPIE 7833*.

[4] D., Radulescu (2012), AIS Indexer User Guide.

[5] M.-O., St-Hilaire and M., Mayrand (2012), Automatic Publication of a MIS Product to GeoNetwork - Case of the AIS Indexer.

10

Annex A: Log file structure

Log files contain a list of AIS reports. The generation of these files is not in the scope of this call-up. It is assumed that these files are generated through other applications and that the AIS Indexer should only be able to read them in order to generate maps. Static maps can only be generated with these files.

The entry structure is as follows: day/month/year hr:min:sec, MMSI, Latitude, Longitude

A log extract is presented below:
```
30/09/2010 21:00:00, 316005416, 50.086888333333334, -125.28631333333334
30/09/2010 21:00:00, 368293000, 47.63009666666667, -122.38216833333334
30/09/2010 21:00:00, 316011504, 49.19606, -122.90812666666666
30/09/2010 21:00:00, 366976870, 47.262605, -122.43240833333333
30/09/2010 21:00:00, 316003657, 48.429066666666664, -123.37276666666666
30/09/2010 21:00:01, 316001247, 49.37663333333333, -123.2708
30/09/2010 21:00:01, 367019340, 50.37305, -125.65856166666667
30/09/2010 21:00:01, 311013500, 49.13028333333333, -123.39851666666667
30/09/2010 21:00:01, 367082010, 48.06439, -125.73521833333334
30/09/2010 21:00:01, 316001246, 49.834133333333334, -124.53081666666667
```

This page intentionally left blank.

Annex B: Saved maps structure

Saved maps can be loaded by the AIS Indexer with very little processing. Each map file contains two sections: the Map Settings and Map Values. Using the computer science paradigm, the Map Settings contain all data required to define the map while the Map Values contain all data required to instantiate the map. Both sections are rendered in CSV format. The Map Settings section takes the form of: Field,Value

Map Values are of form:
X Grid Position, Y Grid Position, Index Value

Only non-zero index values are recorded.

Below is an excerpt from a saved map file:
```
***EDITING THIS FILE MAY RENDER IT UNUSABLE***
<MapSettings>
StartTime,1320279600
EndTime,1320280800
Duration,1200
Frequency,600
Delta T,6
NorthLat,70.0
WestLong,-150.0
SouthLat,0.0
EastLong,-20.0
HorizontalCellSize,0.1
VerticalCellSize,0.1
NulberOfColumns,1300
NulberOfRows,700
</MapSettings>
<MapValues>
1272,126,1.0
1276,133,1.0
1279,136,0.75
1278,139,1.0
1296,139,1.0
1265,140,0.5
1266,140,1.0
560,145,1.0
561,145,1.0
575,146,1.0
575,147,1.0
1264,149,0.5
```

```
1265,149,0.65
1273,152,0.5
1276,153,1.0
1276,154,1.0
542,156,1.0
1263,156,1.0
1287,159,1.0
```

Annex C: Saved alerts structure

Alerts are saved to the local file system on a session basis in the `Monitored Alerts` folder. The session limits are defined by the monitor button being toggled on while the session end is triggered by the monitor button being toggled off or the map viewer being closed. In the case of a live map generation, several maps may be generated within a single session.

Alerts come in two flavors: *Missing* and *Now Present* (or reappearing). Since a ship can go missing, and then reappear only once before going missing again, every missing-reappearing duo is assigned a unique ID. This ID can be used to quickly identify the ships that never reappear or perhaps ships that don't report often enough.

Alert values are stored in CSV format. The structure of alert values is defined below: Missing-Reappearing ID, MMSI, Alert Type, Longitude, Latitude, Report Time, Cell Index Value

Below is an excerpt from a saved alerts file:

```
<AlertValues>
0,636011641,Missing,-75.58866666666667,39.58266666666667,1320322491,1.0
1,371559000,Missing,-104.11566666666667,18.0763,1320322696,1.0
2,235004249,Missing,-75.56034166666667,37.266895,1320322983,1.0
3,636091501,Missing,-76.13340666666667,34.298165,1320322944,1.0
4,338100376,Missing,-76.29135333333333,37.00978666666666,1320322944,1.0
3,636091501,AIS Now Present,-76.04184666666667,34.354643333333335,1320323978,0.0
5,309484000,Missing,-87.68725,18.731283333333334,1320322868,1.0
5,309484000,AIS Now Present,-87.68726666666667,18.731283333333334,1320324501,0.0
6,367084480,Missing,-94.76661333333334,28.419428333333332,1320322805,1.0
7,366766990,Missing,-80.20960666666667,28.52665,1320322479,1.0
8,237898000,Missing,-125.33415,38.33586666666667,1320322504,1.0
6,367084480,AIS Now Present,-94.79328666666666,28.456166666666668,1320324636,0.0
```

List of symbols/abbreviations/acronyms/initialisms

AIS Automatic Identification System

AOI Area Of Interest

FIFO First-In, First-Out

GB Gigabyte

hrs hours

min minutes

ms milliseconds

MSARI Maritime Situational Awareness Research Infrastructure

MSSIS Maritime Safety and Security Information System

MVP Model View Presenter

s seconds

UI user interface