



BlueNavi: A Microservices Architecture-Styled Platform Providing Maritime Information

Liu, Hongze
Jurdana, Irena
Lopac, Nikola
Wakabayashi, Nobukazu

(Citation)

Sustainability, 14(4):2173

(Issue Date)

2022-02

(Resource Type)

journal article

(Version)

Version of Record

(Rights)

© 2022 by the authors. Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

(URL)

<https://hdl.handle.net/20.500.14094/0100477360>



Article

BlueNavi: A Microservices Architecture-Styled Platform Providing Maritime Information

Hongze Liu ^{1,*} , Irena Jurdana ² , Nikola Lopac ^{2,3}  and Nobukazu Wakabayashi ¹ 

¹ Graduate School of Maritime Sciences, Kobe University, Kobe 658-0022, Japan; waka@kobe-u.ac.jp

² Faculty of Maritime Studies, University of Rijeka, 51000 Rijeka, Croatia; irena.jurdana@pfri.uniri.hr (I.J.); nikola.lopac@pfri.uniri.hr (N.L.)

³ Center for Artificial Intelligence and Cybersecurity, University of Rijeka, 51000 Rijeka, Croatia

* Correspondence: i@liu.am

Abstract: Traditional methods of marine navigation are undergoing a revolution brought about by the almost universal adoption of the Automatic Identification System (AIS). AIS exchanges a wealth of navigational information among vessels and between ships to shore through Very High Frequency (VHF). With AIS data integrated into the Electronic Chart Display and Information System (ECDIS), the identification and navigational information of surrounding vessels as well as aids to navigation can be reflected on the electronic charts in real time, despite some problems such as the low AIS carriage rate on small vessels where it is not mandatory and the high cost of ECDIS preventing such vessels from installing it. In this paper, we introduce BlueNavi, a lower cost but sustainable maritime information providing platform built with microservices architecture allowing flexible on-demand scalability and cross-platform adaptability. Applications served by BlueNavi can provide users with data either stored in a remote data center through the internet or received locally by devices connected to the station without the need for the internet. From our land test, we show that users with only an internet connection but without any AIS equipment can also obtain live AIS data collected by other stations. Conversely, with access to the internet, BlueNavi can also send data back to the land stations, enabling other ships to identify non-AIS ships as well. Through the live-ship test, we demonstrate that BlueNavi works well offline in cooperation with shipborne AIS equipment. We also look at some possible application scenarios for BlueNavi with other data sources and means of communication other than AIS and VHF that can be expanded to the platform. BlueNavi will enable inexpensive ship identification for small vessels and provide an extension of functionality to ECDIS for large ships.

Keywords: e-navigation; information platform; design; microservices architecture; maritime communication system; automatic identification system; maritime transport



Citation: Liu, H.; Jurdana, I.; Lopac, N.; Wakabayashi, N. BlueNavi: A Microservices Architecture-Styled Platform Providing Maritime Information. *Sustainability* **2022**, *14*, 2173. <https://doi.org/10.3390/su14042173>

Academic Editor: Stratos Papadimitriou

Received: 15 December 2021

Accepted: 7 February 2022

Published: 14 February 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Automatic Identification System (AIS) is a maritime communication system using marine Very High Frequency (VHF) radio and Self Organizing Time Division Multiple Access (SOTDMA) transmission protocol for the exchange of ship identification and other navigational information [1–3]. AIS can transmit information autonomously and continuously to other ships and facilities equipped with receivers within an acceptable communication range, which is considered to be a major break from the traditional passive systems such as radar [4]. According to the guidelines given by the International Maritime Organization (IMO), the purposes of AIS include helping to identify ships, assisting in target tracking, assisting in search and rescue operations, simplifying information exchange, and providing additional information to assist situation awareness [1]. AIS was first designed in the 1990s [4] and has been in continuous development since then.

For shipborne AIS, transmitted information may include dynamic information, providing live information about the vessel, such as a ship's geographical coordinates, course

over ground (COG), speed over ground (SOG), etc., as well as static information that is relatively permanent to a particular vessel, including ship's name, Maritime Mobile Service Identity (MMSI), dimensions, etc., along with voyage-related information specific to each voyage, such as the ship's navigational status, destination, estimated time of arrival (ETA), etc., [1,2,5]. Providing this dynamic, static, voyage-related, and other additional information, AIS has improved both the safety of navigation on water as well as an enhanced level of surveillance from land [6], but it still faces some difficulties.

For example, not all ships carry AIS; therefore, even though it was stipulated in the International Convention for the Safety of Life at Sea (SOLAS) [7] since 2008 for all ships of 300 gross tonnage (GT) or more engaged in international voyages, all ships of 500 GT or more not engaged in international voyages, and all passenger ships regardless of size have become mandatory for navigational safety, there are still many vessels not included by the regulation. This includes vessels such as leisure crafts, fishing boats, and ships between 300 and 500 GT not engaged on international voyages [1,8], which means those vessels are still unable to obtain other ships' navigational information through AIS as well as making it difficult for other ships to detect and identify them by passive systems such as radar, due to their lesser size [9].

To solve this problem, we developed BlueNavi, a sustainable platform solution to bridge the gap between the widespread use of AIS in field of navigation safety and the low carriage rate of AIS on small vessels.

BlueNavi consists of ship stations, land stations, and data centers. It gives users the flexibility to customize on-demand, based on practical application scenarios.

BlueNavi ship stations can work independently with existing or extra Class-A AIS, Class-B AIS, or just simple lower-cost AIS receivers that do not have a data transmission function. Data received by those on-board AIS devices are parsed, preprocessed, stored, and then provided to users through the application (hereinafter, called *BlueNavi App*).

BlueNavi can also work without the need for any AIS devices installed when users are able to connect to land stations via the internet. Data, in this instance, are collected from other ship stations or land stations equipped with AIS devices and stored in data centers. Land stations provide data to users on request.

BlueNavi ship stations working with onboard AIS devices and land stations through the internet can function as a complete system. Consequently, users can obtain AIS information collected not only by shipborne AIS but also by land stations and/or other ship stations regardless of the limitations of the VHF propagation distance. Furthermore, due to the substantial computing power of land stations, services requiring big data, heavy computing, or high loads, such as statistics-based or artificial intelligence-based data analysis, can also be provided to users on board. Moreover, in addition to the AIS information, land stations can also provide other maritime information, including map data, satellite chart data, and meteorological and sea state information, to support the officer of the watch (OOV) and improve the safety and convenience of navigation.

This paper is organized as follows: Section 2 briefly introduces the background and related work of our research. Section 3 explains the specific design of the BlueNavi platform. Section 4 details the land and live-ship tests. Section 5 discusses the test results. Section 6 summarizes and presents the conclusions and outlines the main directions for future research.

2. Background and Related Work

In this section, we discuss the problem in terms of AIS in safe navigation and technologies chosen to help developing BlueNavi.

2.1. AIS in Safe Navigation

AIS is considered to play an important role in safe navigation because of its ability to provide accurate dynamic information coming directly from various sensors on board as well as other static and voyage-related data [1–3,10]. Specifically, AIS data are significant

in collision avoidance decision [11,12], collision risk detection and analysis [13–16], path planning [17,18], etc., [19]. For these reasons, the mandatory equipment of AIS for passenger ships and large cargo ships has been officially included in SOLAS. In addition, on the basis of SOLAS, countries around the world have also made more stringent regulations on the mandatory carriage of AIS equipment on their vessels [20].

However, despite this, there is still a significant number of small vessels that are not yet equipped with AIS. According to a survey conducted by Serra-Sogas [21], an estimated 70% of the total number of vessels do not transmit AIS in target areas of the Salish Sea, Canada alone, and a significant portion of these non-AIS vessels are recreational boats. In addition, during Tanaka's research voyage [22], it was found that in various sea areas around the world, including Panama, Suez, Seto Inland Sea of Japan, etc., the AIS carriage rate is less than 50%. The absence of AIS not only brings a risk to the navigation of the ownship but also poses a risk to other ships and traffic surveillance because of the AIS-unrecognizability.

To solve this problem, two approaches have been put forward. One is to equip these vessels with Class-B AIS equipment [23] which, although not in full accordance with the related requirements of IMO [1], has an advantage of cost over fully functional Class-A devices [23]. The other approach is for the vessels to use the internet. In this scenario, they do not need AIS but can still obtain the AIS information collected by third-parties while still inshore when internet connections are available.

However, the first solution still requires additional software because AIS device itself either has only a minimum keyboard and display (MKD) or does not even have display or input capabilities (on some Class-B or non-standard AIS devices). Otherwise, without software to help visualize the data and provide various functions, it is of limited help to the navigation of the ownship. The second solution is problematic since they are still unable to transmit information to other ships for identification, and when sailing in waters with no internet, they are unable to receive information.

In light of these approaches, we iteratively designed BlueNavi.

2.2. *Microservices Architecture in Software Development*

Microservices architecture is a software architecture that approaches the modularization concept and contrasts to the traditional monolithic architecture that encapsulates all the business logics in one piece and therefore makes the software system can only be deployed as a single unit [24].

Figure 1 gives a brief illustration of a simple monolithic architecture, where everything is integrated together, packaged as a monolithic deployment bundle and communicates internally through methods such as method calls [25,26].

The power of a single computing component is, after all, limited. In order to solve the problem of high concurrency and high availability in the architecture, redundancy is introduced. Deploying monoliths into multiple server components with load balancing instead of just a single component spreads a large number of jobs rationally across multiple operating units for execution and, therefore, can improve the reliability of the whole system. Figure 2 shows the illustration of a monolithic architecture-styled software with load balancing introduced.

Although software systems with monolithic architectures can also follow the modular mindset, which means that inside the monoliths, each business logic can also be structured, developed, and tested separately, they have to be simultaneously deployed to the same infrastructure, sharing the very same resources, such as central processing unit (CPU), memories, etc., Therefore, they are disadvantaged by maintainability and scalability problems [24,27].

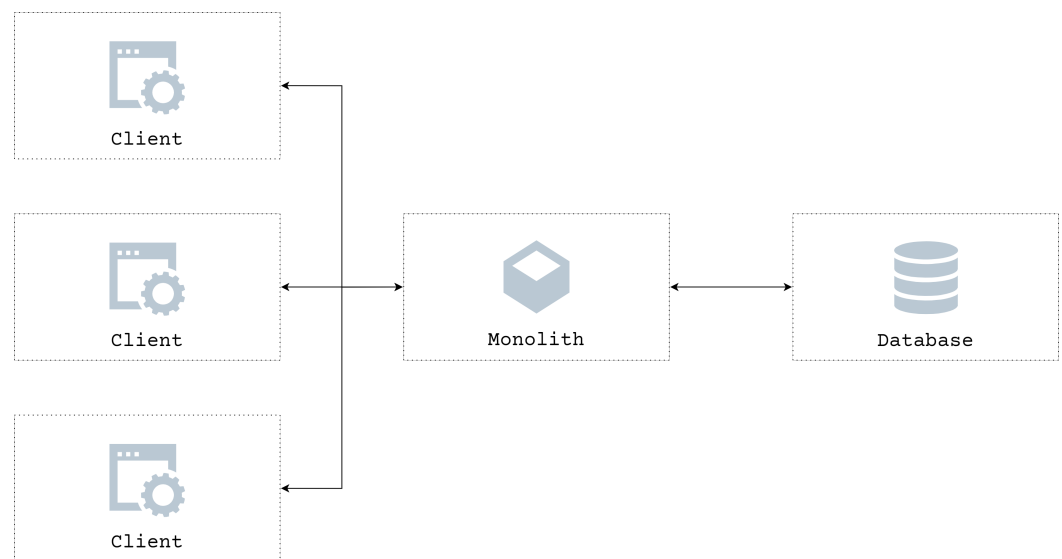


Figure 1. Example of monolithic architecture.

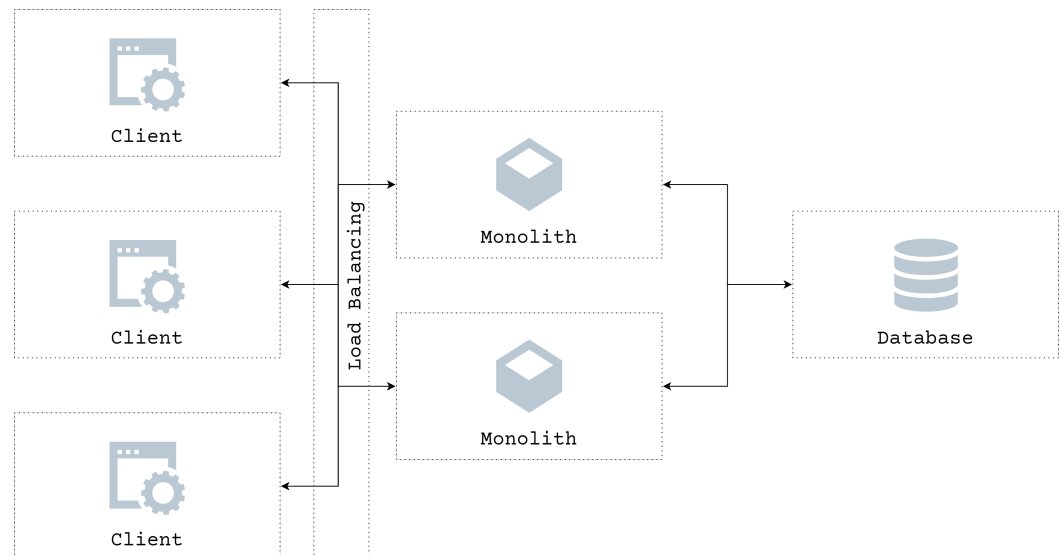


Figure 2. Example of monolithic architecture with load balancing.

The emergence of microservices architecture has alleviated those issues. A microservices architecture-based software is actually a collection of different small, well-grained, single-purpose, and independently deployed services usually divided according to the functionality. Microservices communicate via the network, using application programming interface (API) or messages [24].

Figure 3 gives an example of the microservices architecture. In this example, the client applications communicate only with the API gateway responsible for distributing their requests to the specific microservice. Once the communication interfaces between them are well defined, each microservice can be designed, implemented, tested, and deployed independently.

The database, in this case, is deployed separately and not exposed to other microservices directly but only the responsible one. However, in some other cases, databases can also be deployed together with microservices as a part of them.

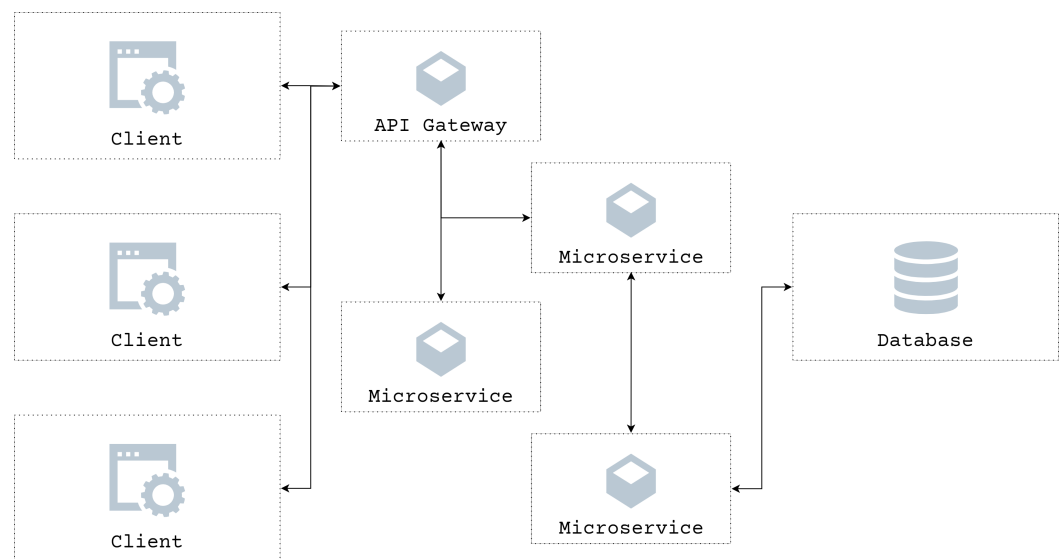


Figure 3. Example of microservices architecture.

Microservices architecture has demonstrated widespread popularity because of its significant advantages in technological heterogeneity, replaceability, maintainability, scalability, and also availability.

Following the idea of modularization, well-grained microservices can be designed, developed, and deployed independently, therefore giving the developer more freedom of choice in terms of technological implementation, including the code base, platform, and the database to go with, depending on the needs. Its highly modular nature makes it easy to replace single components, even with totally different technology unless the same interface. Maintenance, by the same token, becomes easier as well. Furthermore, unlike monoliths, which have to scale everything together when the performance bottleneck is reached, microservices, on the other hand, can be scaled independently in need or only those intensely used. Moreover, the failure of one microservice usually does not cascade. The failure may not influence other parts of the remaining system for the other functions, which also helps troubleshooting [24,28].

Since BlueNavi is designed to have such features, the microservices architecture was chosen.

3. BlueNavi: Platform Design

Based on the findings in Sections 1 and 2, we designed and developed BlueNavi, a microservices architecture-styled platform providing users various kinds of maritime information, currently led by AIS information, in order to ensure safe navigation as well as other purposes.

BlueNavi platform consists mainly of ship stations, land stations, and data centers. Ship stations are usually found on board with AIS devices and can work alone or be networked to collaborate with land stations. Land stations are responsible for processing and providing data or handling the data sent by the ship station and are connected to the data center. Both ship stations and land stations include microservices that can provide BlueNavi App to users, but the services provided by ship stations are mainly oriented toward the specific ship, while that of land stations can be provided to an unspecified group of users through the internet.

3.1. User Scenarios

Here, we provide some scenarios where the BlueNavi can be employed, using three virtual ship examples, followed by the platform description.

3.1.1. From Zero to One

Sarobetsu is a yacht with no AIS equipment on board nor ECDIS. The owner wanted a retrofit with a BlueNavi ship station that would allow them to obtain real-time positions and various other navigational information of the AIS-equipped vessels around it and display them on a tablet computer while *Sarobetsu* was operating. At the same time, the owner also wanted to make the yacht's position and its name visible to other ships.

3.1.2. Minimization

Kamui is a small fishing boat that usually travels offshore and can easily pick up cellular network signals. The owner plans to use the BlueNavi App on a smartphone to obtain more information to help with navigation, replacing the currently used regular map application. However, for some reason, *Kamui*'s owner does not want to reveal its location to the other boats. Since the internet is accessible on *Kamui*, all services can be provided by the land station, and the ship station is not needed in this case.

3.1.3. The More, the Better

Okhotsk is a container ship already equipped with AIS and can access the internet through satellite broadband service. In addition to AIS data, the company also wanted to overlay some necessary weather, safety, and other maritime information onto the electronic chart. Furthermore, the company also wants to regularly receive, record, and make use of data sent from *Okhotsk* and other ships it owns to facilitate ship management. To this end, the company introduces the complete BlueNavi platform, including a number of ship stations, a land station, and a data center.

3.2. Platform Description

The design of the BlueNavi platform is described separately in terms of the ship station, land station, and data center.

3.2.1. Ship Station

Figure 4 gives an overview of a fully configured BlueNavi ship station, which includes four microservices, a data service (Receiver), a data service (Provider), an API service (Gateway), and an app service (Renderer), as well as a database. Arrows indicate the data flow. The blue arrows stand for the original message broadcast in NMEA 0183 format [29], a specification for communication between marine electronics, and green arrows for the parsed data in JSON data interchange format [30].

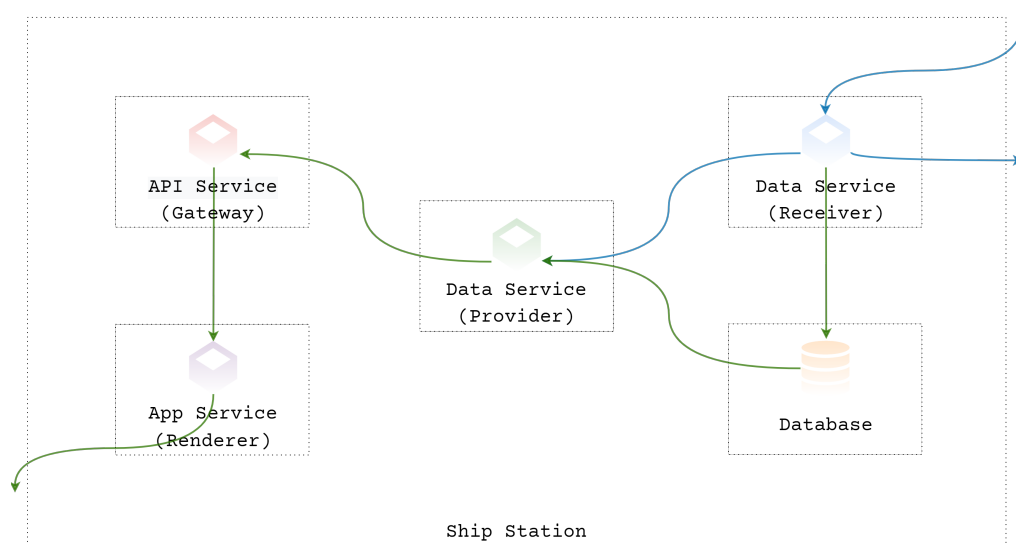


Figure 4. BlueNavi ship station.

The raw AIS data in NMEA 0183 format are transmitted to the Receiver via wired or wireless (depending on whether supported by the AIS equipment) network. When the Receiver obtains a message, it forwards it to the Provider through broadcast and starts parsing the message at the same time. If a land station is accessible, the Receiver there can also receive the broadcast messages. Parsed data are written to the database.

The Provider makes the navigational data available to the client when needed. There are two sources of data: broadcast raw messages received passively from the Receiver (indicated by the blue arrow) and parsed data obtained by active queries to the database (indicated by the green arrow). For the former, microservices communicate with each other via messages; for the latter, they communicate via APIs.

BlueNavi App is a web application, with a separate front-end and back-end design. The API service (Gateway) is responsible for handling requests from the front-end rendered by the app service (Renderer).

The database is optional. Data received by the Receiver are kept in the database for backup or further use, such as data analysis or route planning. The database can be omitted if the user does not need the functionality based on historical data. In this case, the Provider has only one data source, with the real-time data coming from the Receiver, as shown in Figure 5.

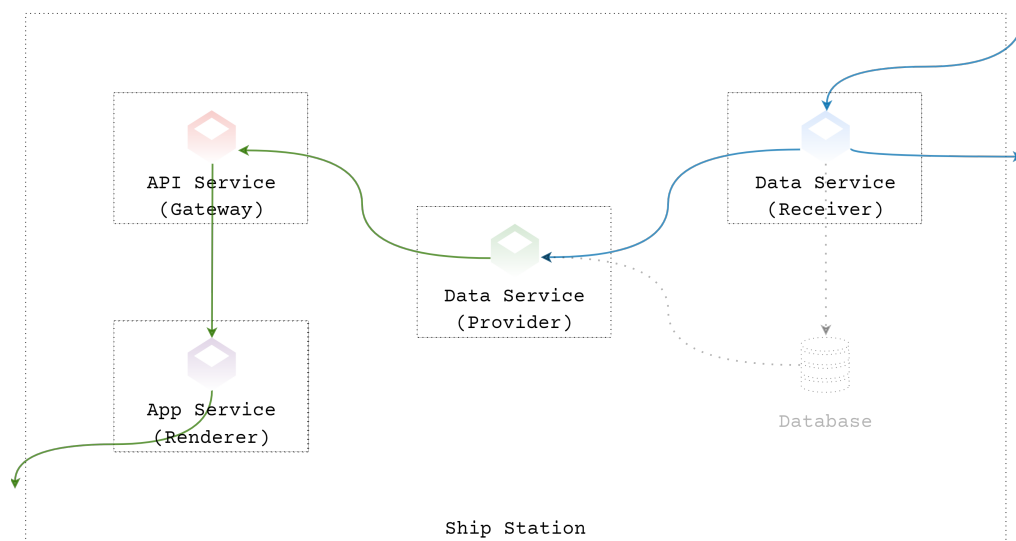


Figure 5. BlueNavi ship station (without database deployed).

The deployment of these components is flexible, as different microservices and the database of one ship station can be deployed on the same or different computers on board, as needed.

3.2.2. Land Station

On land, microservices and databases are deployed separately in the land stations and data centers. Generally, there are two kinds of land stations—the full-featured land station that has the ability to provide services to users via the internet and the minimized land station that is not user-oriented but is only used for collecting AIS data due to the limited transmission distance of VHF.

Due to the nature of the microservices architecture, microservices developed for ship stations can also be used in land stations, whether full-featured or minimized, without the need to write a separate set of software systems, as in the case of traditional monolithic architecture systems.

A simple example of a full-featured land station is provided in Figure 6.

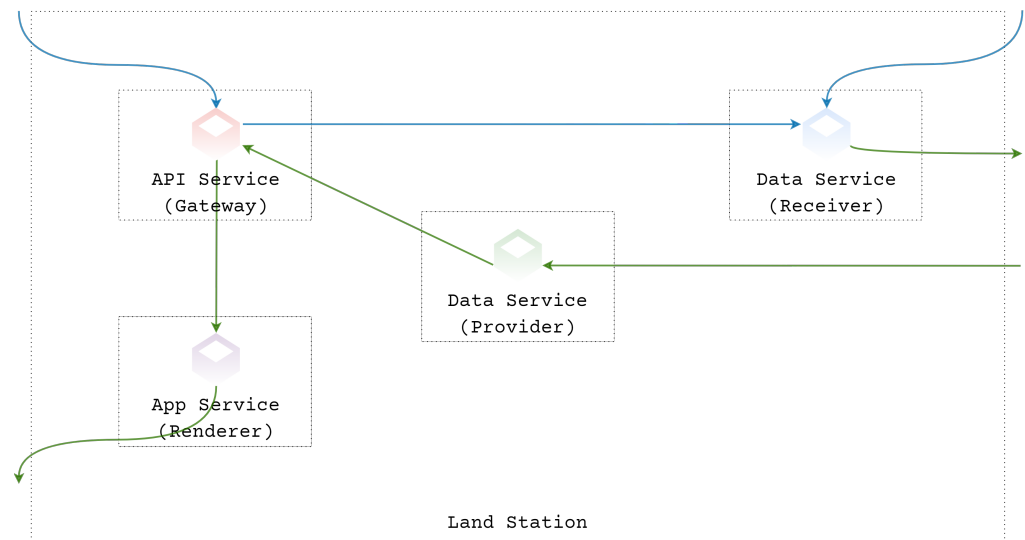


Figure 6. Full-featured BlueNavi land station.

In Figure 6, data service Receiver is deployed in the full-featured land station to receive the messages from the AIS equipment and the messages broadcast by the ship station. The ship station broadcast messages arrive at Receiver via the API gateway over the internet and are written to the database after being parsed. The messages coming from AIS equipment go to Receiver that works the same as that of the ship station.

Compared with ship stations, full-featured land stations have higher computing power and can thus handle more complex functions. Figure 7 shows an example of introducing new functions (microservices) to an existing system.

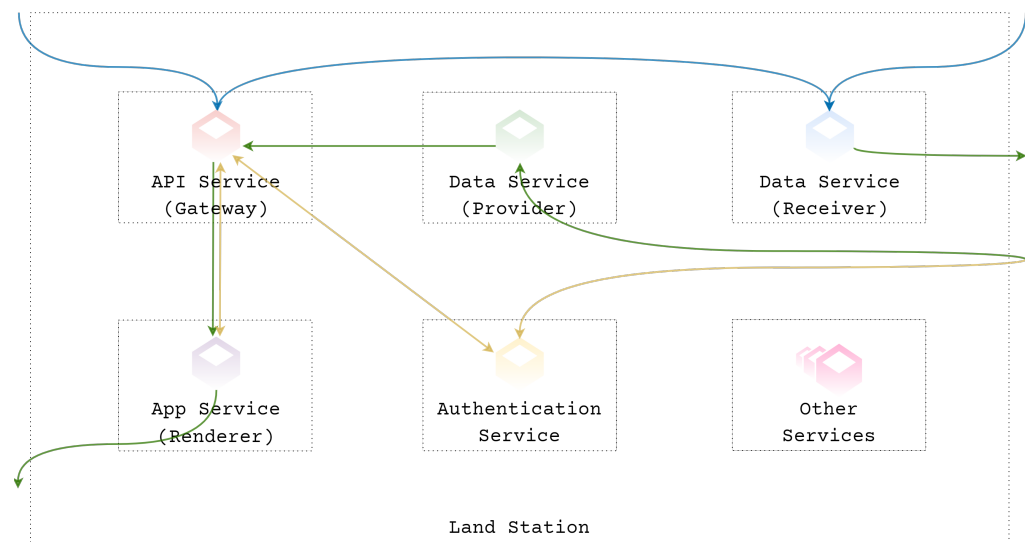


Figure 7. Full-featured BlueNavi land station (with new microservices introduced).

In Figure 7, a new authentication service is introduced to make certain core functions or those requiring large computational volumes accessible only to authorized users. Yellow arrows indicate the data flow of authorization information, such as valid tokens, that comes from the authentication database, separated from the AIS databases but also located in the data center.

Moreover, there are times when we simply need specific land stations to collect AIS data at certain locations and do not need them to provide heavy services to the user. In this case, a minimized land station can meet the demand. The minimal land station design is straightforward, as shown in Figure 8.

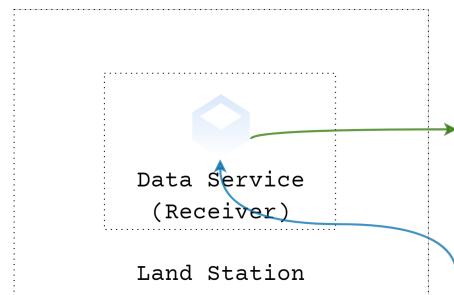


Figure 8. Minimized BlueNavi land station.

Here, only one data service Receiver needs to be deployed in a minimized land station connected with an AIS receiving device.

3.2.3. Data Center

The data center and land stations can be in a one-to-many relationship, i.e., multiple land stations deployed in different locations can share the same database.

Figure 9 illustrates the design of a data center. The user database stores all user information, and the authentication database is used to store temporary tokens assigned to authorized users to access some specific services. AIS data are stored separately in a real-time database and a historical database. The data service (Manipulator) is responsible for sampling the real-time data, filtering the duplicated data and rejecting the outdated data in the real-time database, then writing them to the historical database at regular intervals.

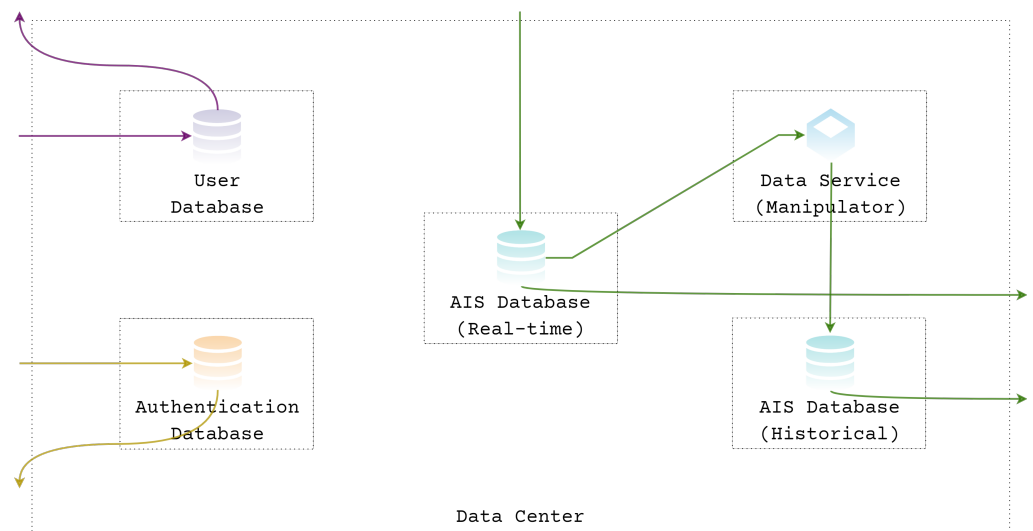


Figure 9. BlueNavi data center.

When the business volume reaches a certain level, beyond which one single data center may not be able to meet the demand, distributed data centers can also be used.

4. Implementation and Tests

We implemented BlueNavi guided by two considerations. The first one is the compatibility so that users can use existing equipment on board, including the AIS equipment and computers that can serve as the BlueNavi operating environment, as much as possible. The second consideration is the accessibility so that the BlueNavi App is available regardless of the device, including mobile phones, tablets, laptops or desktops, and the operating system, such as iOS, Android, Windows, Linux or Mac OS.

4.1. Implementation

To ensure compatibility, microservices are deployed using Docker, which enables them running in a loosely isolated environment, less dependent on the given host [31].

For the ship station, we deployed all the necessary microservices on a Raspberry Pi connected to AIS Class-B equipment via wireless network. Due to the cost advantage of Raspberry Pi compared to other computers, the solution can be implemented at a lower cost for small ships. In order to test whether the system can also work well on different platforms, in addition to the Raspberry Pi with its ARM architecture and UNIX-like operating system, we also deployed the same microservices on a Windows laptop with x86 architecture. Both of these ship station platforms were used in the live-ship tests.

For the land station, we deployed one data service (Receiver) in our laboratory located in Kobe, Japan, connected to the laboratory's AIS receiving equipment as the land station's data source. In addition, due to performance and network bandwidth reasons, we deployed the rest of the necessary microservices on a server located in Tokyo, Japan, to be able to serve users in unspecified geographic areas better. Here, different components of the land station are deployed in different places to meet different needs, which is one of the advantages of the microservices architecture.

The data center is also located in Tokyo and is deployed separately from the land station. The database is only accessible to related microservices on specific ports for security reasons.

The BlueNavi App provided by the app service (Renderer) uses web development technology, built with Angular, TypeScript, HTML, and CSS, and can run on most modern browsers, including Chrome, Safari, Firefox, and Edge. Most of the microservices were developed using Node.js and JavaScript, and some with Go as well. Due to AIS data characteristics [32], the MongoDB NoSQL database was chosen to store the parsed AIS data.

4.2. Land Test

In the land test, we simulated the situation where a ship is not equipped with AIS and can only access the data through the BlueNavi App provided by the land station via the internet, and for situations where the Vessel Traffic Services (VTS), ship company, port administration, and other general users obtain AIS data on land.

Figures 10 and 11 show the screenshots of the BlueNavi App running on both the laptop and the mobile phones during the land test. Chart data from the Hydrographic and Oceanographic Department, Japan Coast Guard, and map data provided by different online map providers were appropriately fused. Navigational information received by the AIS equipment in the land station is accurately parsed and represented on the map. Users can click on the small triangular icons in green (stands for data coming from Class-A AIS) or yellow (from Class-B AIS) representing vessels, and then more specific dynamic, static, and voyage-related information are displayed in the form of cards in the upper right corner. Users can also drag and drop the cards to any location within the map area, as shown in Figure 11.

In other words, for services provided by the BlueNavi land station, users only need a device with a modern browser and an internet connection, whether on board or on land.

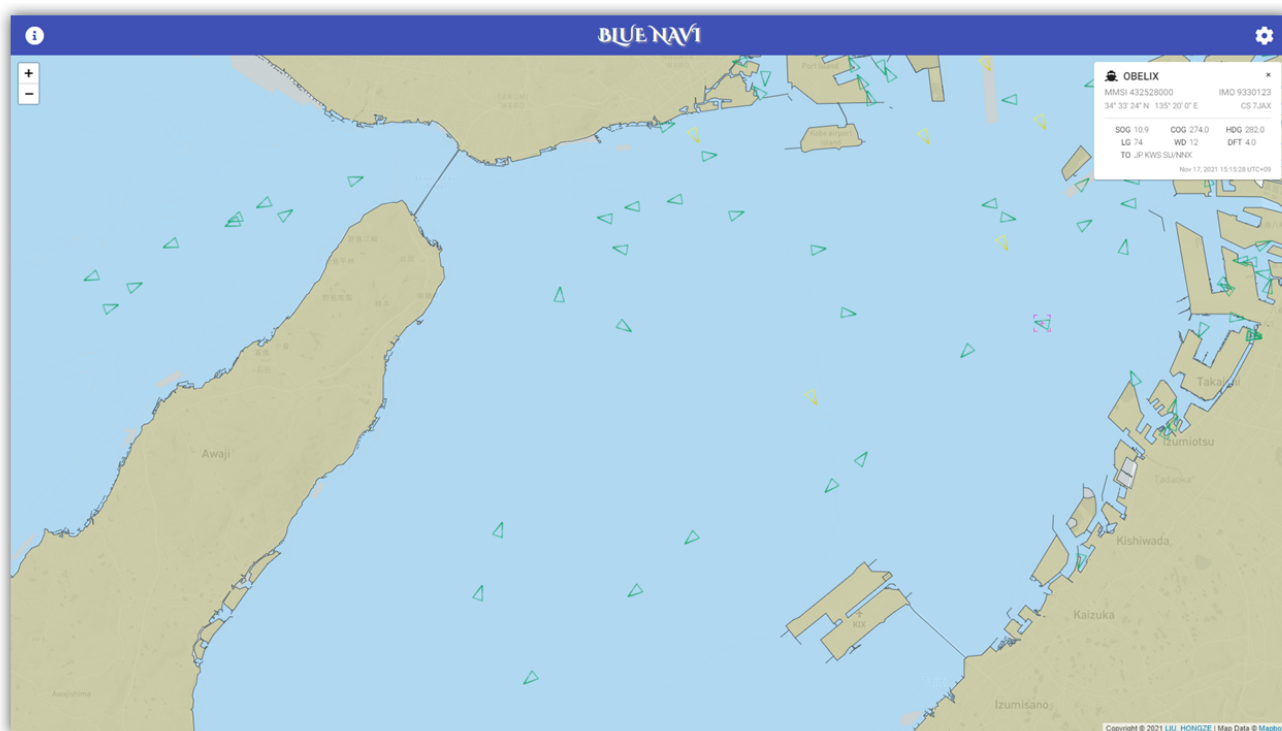


Figure 10. Screenshot of BlueNavi app in land test: via Chrome on Windows laptop.

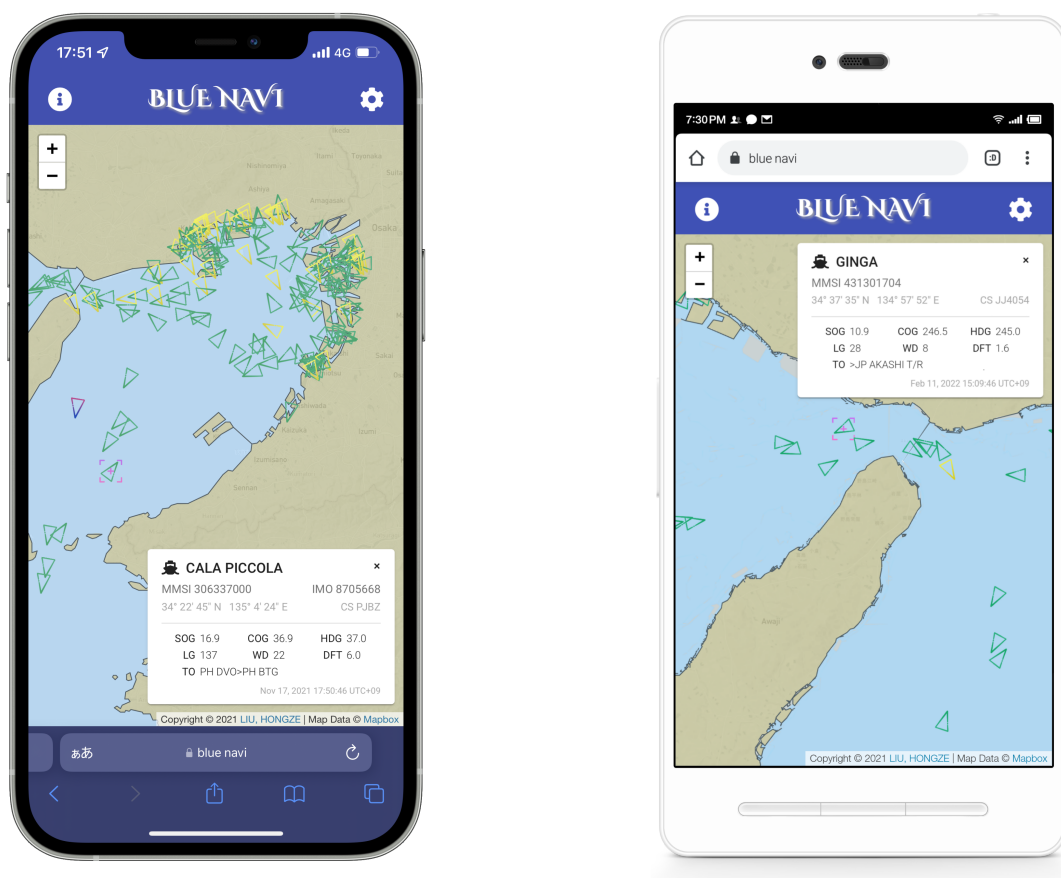


Figure 11. Screenshots of BlueNavi App in land test: via Safari and Chrome on iOS and Android mobile phones.

4.3. Live-Ship Test

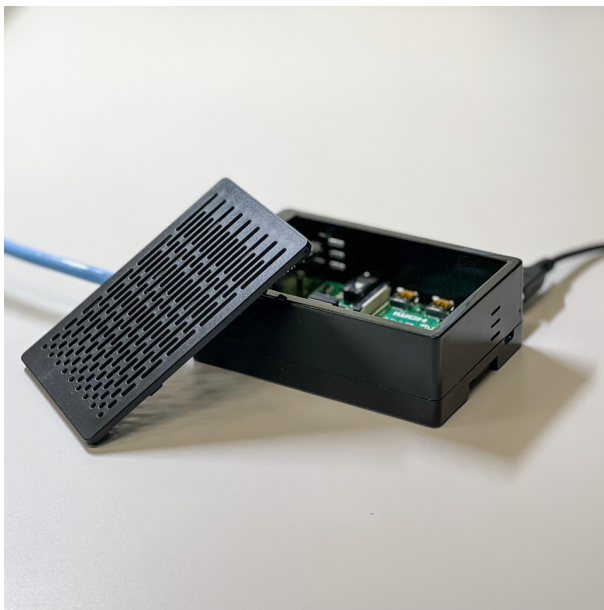
We also conducted two live-ship tests for BlueNavi. The first test was performed on the training boat, *Hiyodori*, Tokyo University of Marine Science and Technology, and the second test was conducted during the last research navigation of the Kobe University training ship, *Fukae Maru*. In the first test, we simulated the scenario of installing a Class-B AIS on small vessels which are not mandated. In the second test, we simulated the scenario of introducing BlueNavi to a ship already equipped with AIS. Figure 12 shows Hiyodori and the first live-ship test. Figure 13 shows Fukae Maru and the second live-ship test. BlueNavi ship stations were running on the Raspberry Pi, and the BlueNavi App can be accessed from anywhere on board.



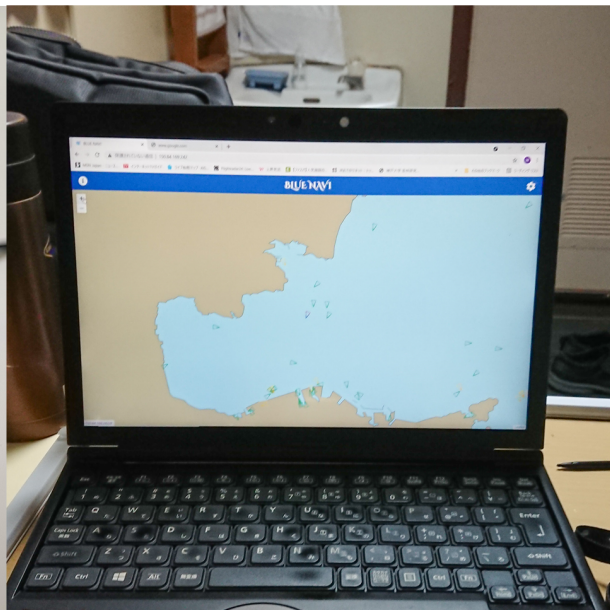
Figure 12. The first live-ship test: (a) training boat Hiyodori, (b) Raspberry Pi serves as the ship station, and (c) Class-B AIS device connected to the ship station.



(a)



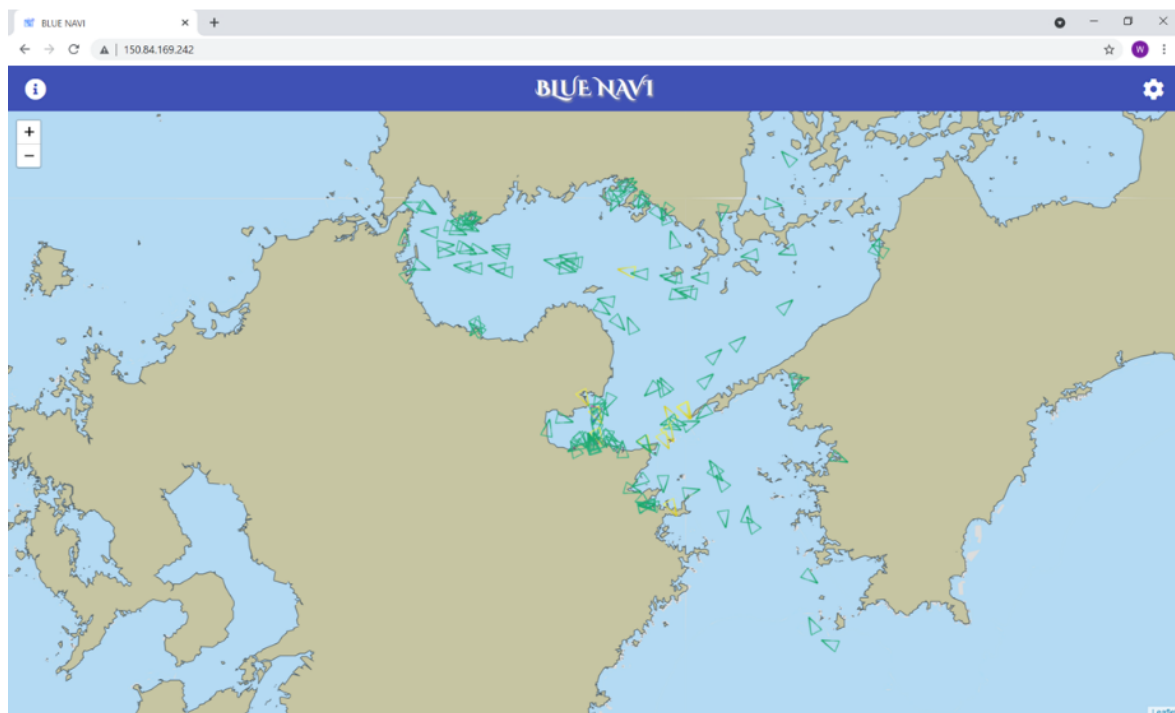
(b)



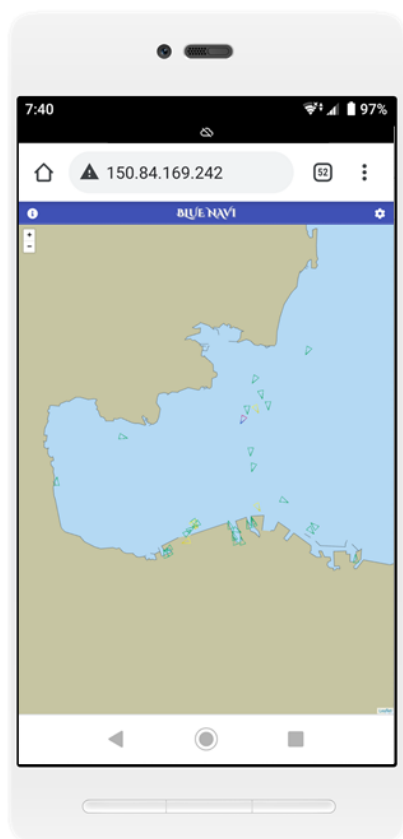
(c)

Figure 13. The second live-ship test: (a) training ship Fukae Maru, (b) Raspberry Pi serves as the ship station, and (c) BlueNavi App running on the client computer, located in crew living quarters.

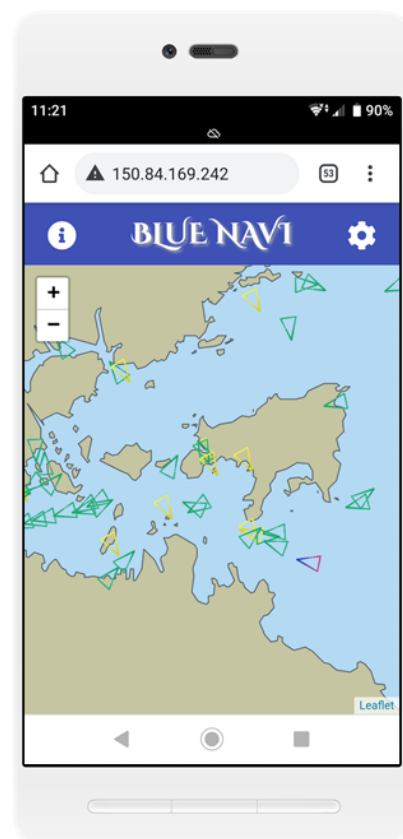
Figure 14 shows the screenshots of the BlueNavi App provided by the ship station. Information received by the shipborne AIS equipment is accurately reflected on the user's terminal.



(a)



(b)



(c)

Figure 14. Screenshots of BlueNavi app in the second live-ship test: (a) accessed at Bungo Channel/Suo Sea, Japan, via Chrome on Windows laptop, (b) accessed at Beppu Bay, Japan, via Chrome on Android mobile phone, and (c) accessed near Shodoshima, Japan, via Chrome on Android mobile phone.

5. Discussion

AIS is considered to be an ideal source of information for safe navigation. AIS greatly improves navigation safety by providing information regarding surrounding vessels in real time. However, not all ships are mandated to have AIS installed. In addition, in order to have visual access to navigational information, expensive ECDIS is often required in addition to AIS equipment. These problems prevent many smaller vessels from obtaining and utilizing AIS data. Moreover, for ships currently equipped with ECDIS, the OOW can only operate the system from its position to access information.

Our microservices architecture-styled BlueNavi platform has virtually solved these problems. BlueNavi consists of ship stations, land stations, and data center. Depending on the needs of different users in different scenarios, they can be introduced independently or in a combined way.

5.1. Working Range

Users of fishing boats, yachts, and other small vessels that do not wish to add any additional equipment can use data collected by other stations and stored in the data center, as well as services provided by land stations as long as they have access to the internet. In this case, the working range of BlueNavi depends on the coverage of the internet at sea, that is, the range of radio waves that can be received between a base station installed on land and a terminal such as a mobile phone.

Figure 15 illustrates the internet coverage of NTT DOCOMO, a Japanese operator whose mobile internet service is available at sea [33]. As can be seen from the figure, in Japan, internet service is available within about 10 nautical miles of the sea (areas in Alice blue), which already exceeds the maximum navigable waters range, 5 nautical miles offshore, for *small coastal vessels* defined by the government.

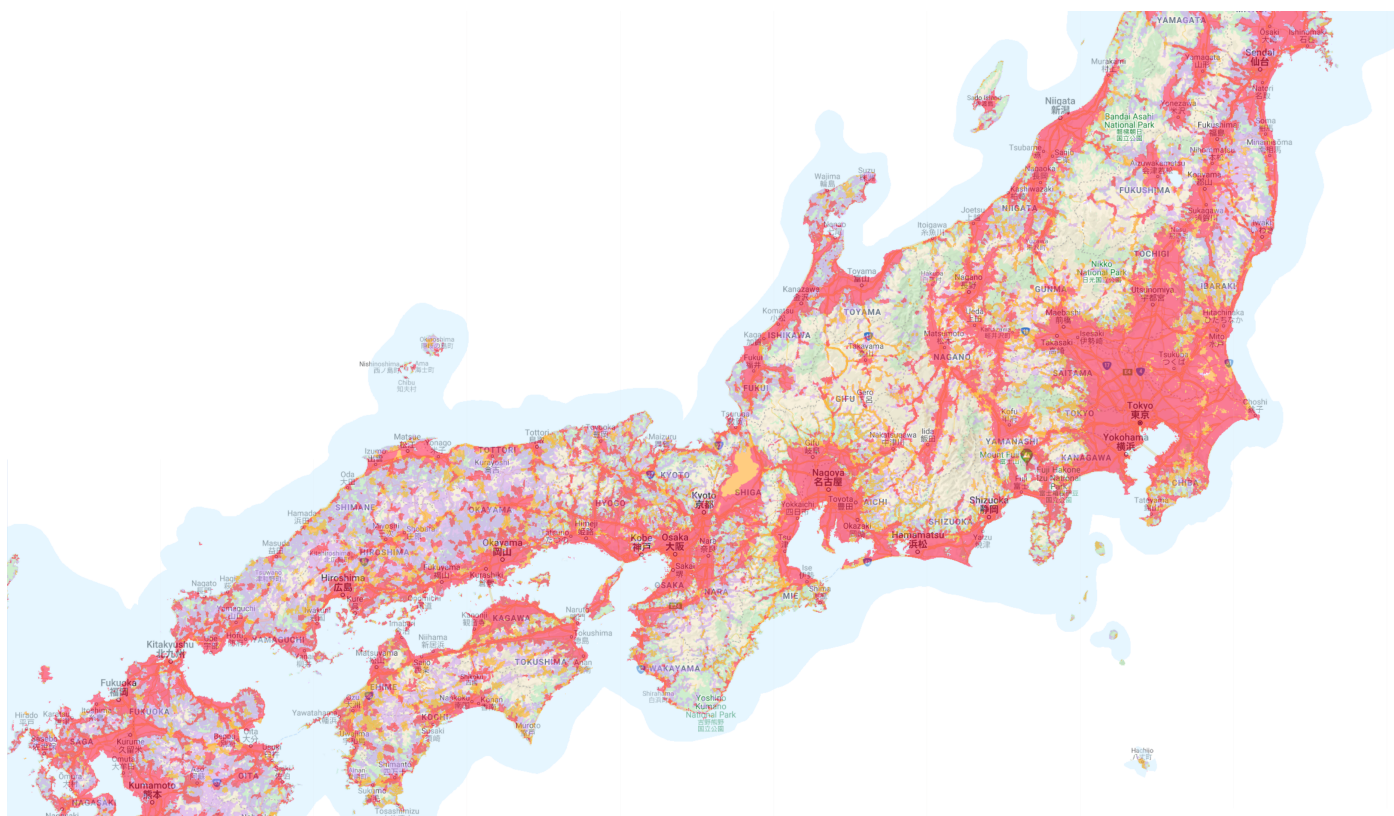


Figure 15. Service area map of NTT DOCOMO in offshore Japan (As of 26 December 2021).

For small vessels that do not have access to the internet or do not want to rely on the internet in order to obtain navigational information, it is possible to retrofit both AIS

equipment and a BlueNavi ship station on such vessels. For large ships and passenger ships where AIS is mandatory, BlueNavi is compatible with all Class-A or Class-B AIS equipment. In both cases, BlueNavi has the same horizontal working range of about 40 nautical miles from the ownship as the existing AIS equipment.

5.2. Costs

BlueNavi also has a major cost advantage. Table 1 shows a cost comparison of shipborne equipment between BlueNavi and traditional solutions. The aforementioned scenario that relies entirely on services provided by land stations and the internet is denoted as BlueNavi 1. The scenario that ownship already equipped with AIS is denoted as BlueNavi 2. Scenarios that need equipping AIS Receiver or Class-B AIS are denoted as BlueNavi 3 and 4, respectively.

Table 1. Hardware cost comparison of shipborne equipment between different solutions.

| | Class-A AIS | Class-B AIS | AIS Receiver | ECDIS | Raspberry Pi | Signal Converter | Network | Total |
|-------------|-------------|-------------|--------------|----------|--------------|------------------|---------|----------|
| BlueNavi 1 | | | | | | | | \$0 |
| BlueNavi 2 | | | | | \$60 | \$40 | \$200 | \$300 |
| BlueNavi 3 | | | \$300 | | \$60 | \$40 | \$200 | \$600 |
| BlueNavi 4 | | \$1000 | | | \$60 | \$40 | \$200 | \$1300 |
| Traditional | \$3000 | | | \$30,000 | | | | \$33,000 |

Compared with the Class-A AIS plus ECDIS solution (denoted as Traditional) widely adopted by large ships, BlueNavi can serve at a relatively low cost, when not considering the user terminal (usually smartphones, tablets, or laptops, which we default to users already owning) costs, the internet usage costs (which vary by country, region, and communication carrier and are relatively not very expensive), and the cost of land station and data center constructions.

5.3. Development and Deployment

Since BlueNavi uses microservices architecture, it has advantages over traditional monolithic architecture in both development and deployment. For example, both ship and land stations need a data microservice to process messages from AIS devices and also an app microservice to provide applications to the user. It is sufficient to develop these microservices once and then deploy them in different ship and land stations, instead of developing two single monolithic systems with overlapping functions and codes for both ship and land stations. In addition, BlueNavi is an easy-to-maintain platform. When a service inevitably encounters a problem, only the related microservices need to be shut down, without affecting the remaining system.

The independent nature of microservices also determines that BlueNavi is a highly scalable platform. When demand for a specific function and concurrency increases, it can be scaled up in a targeted manner, such as expanding the server capacity. Furthermore, in terms of functionality, we can design and implement new features for BlueNavi just by deploying new microservices.

5.4. Compatibility and Availability

In addition to the station being compatible with all kinds of shipborne AIS devices, the station itself can also run on almost all kinds of hosts, from a small Raspberry Pi to a server. Moreover, the adoption of web application development technology makes the BlueNavi App available from almost all smart devices as well. Under the same local area network (LAN), users can access BlueNavi's services anywhere on bridge or board, no longer limited to some fixed position, such as where ECDIS is located.

6. Conclusions and Future Work

In this study, we investigated issues regarding the lack of standard AIS which provides ship identification and other maritime information on vessels where it is not mandatory. We surveyed the literature to examine the AIS carriage rates in different sea bodies. Based on our findings and analysis of probable user scenarios, we designed and developed BlueNavi, using microservices as the architecture to make the platform solution not only capable of meeting the goal of enhancing marine navigation safety but also sustainable throughout future upgrades and process developments. We also designed and developed an interface for BlueNavi using the latest web development technologies to visualize and enable users to interact with data. We were able to deploy BlueNavi to conduct both land and live-ship tests. Our tests demonstrated BlueNavi's efficacy in providing a channel for vessels without AIS to make use of AIS data at a low cost and hence improve the safety of navigation. We also discussed how BlueNavi could be integrated into current AIS-equipped vessels to possibly expand its functionality.

The main purpose of AIS is to identify ships, and the information is mainly about the ships themselves. On the other hand, in coastal areas where internet access is available, various other kinds of information can also be obtained. Weather conditions, sea conditions, and other information can also be integrated with BlueNavi by continuously developing and launching new microservices. For example, Maritime Safety Information (MSI) is also crucial to the safety of navigation. At present, MSI along the coast is mainly available to ships by Narrow Band Direct Printing (NBDP). Our previous research explored the MSI classification problem through machine learning methods based on a digitalization problem [34]. In the future, we should be able to develop related microservices for further preprocessing and broadcasting so that MSI with specific geographic location information can be overlaid directly on the chart of the BlueNavi App for additional safety and convenience of navigation.

In recent years, further upgrades and modernization of the maritime communications network are required, and efforts to improve maritime communications have been conducted [35,36]. For example, the VHF Data Exchange System (VDES), also called the next generation AIS, is being developed for future maritime data exchanges and communications [37]. When VDES becomes available, with the use of satellites, a variety of additional usage scenarios in terms of information broadcast, data fetching, ship reporting, route exchange, etc., will also become possible, even in the open ocean where internet access is currently not available or extremely expensive. BlueNavi has the potential to add a seamless contribution to sustainable maritime transportation safety as VDES becomes available in the near future and broadens its usability platform.

Author Contributions: Conceptualization, H.L., I.J., N.L. and N.W.; data curation, H.L. and N.W.; formal analysis, H.L., I.J., N.L. and N.W.; investigation, H.L. and N.W.; methodology, H.L. and N.W.; project administration, N.W.; resources, N.W.; software, H.L.; supervision, I.J. and N.W.; validation, H.L. and N.W.; visualization, H.L.; writing—original draft, H.L.; writing—review and editing, H.L., I.J., N.L. and N.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: The authors are grateful to Toshifumi Hayashi of Tokyo University of Marine Science and Technology and Tatsuya Yashiro and his group of YDK Technologies Co., Ltd. for their helpful input and support.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

| | |
|--------|--|
| AIS | Automatic Identification System |
| API | Application Programming Interface |
| ARM | Advanced RISC Machines |
| COG | Course Over Ground |
| CPU | Central Processing Unit |
| CSS | Cascading Style Sheets |
| ECDIS | Electronic Chart Display and Information System |
| ETA | Estimated Time of Arrival |
| GT | Gross Tonnes |
| HTML | HyperText Markup Language |
| IMO | International Maritime Organization |
| JSON | JavaScript Object Notation |
| LAN | Local Area Network |
| MKD | Minimum Keyboard and Display |
| MMSI | Maritime Mobile Service Identity |
| MSI | Maritime Safety Information |
| NBDP | Narrow Band Direct Printing |
| NMEA | National Marine Electronics Association |
| OOW | Officer Of the Watch |
| SOG | Speed Over Ground |
| SOLAS | International Convention for the Safety of Life at Sea |
| SOTDMA | Self Organizing Time Division Multiple Access |
| VDES | VHF Data Exchange System |
| VHF | Very High Frequency |
| VTs | Vessel Traffic Services |

References

1. International Maritime Organization (IMO). *Revised Guidelines for The Onboard Operational Use of Shipborne Automatic Identification Systems (AIS)*; International Maritime Organization (IMO): London, UK, 2015.
2. International Telecommunication Union Radiocommunication Sector (ITU-R). *Recommendation ITU-R M.1371-5: Technical Characteristics for an Automatic Identification System Using Time Division Multiple Access in the VHF Maritime Mobile Frequency Band*; International Telecommunication Union Radiocommunication Sector (ITU-R): Geneva, Switzerland, 2014.
3. International Association of Marine Aids to Navigation and Lighthouse Authorities (IALA). IALA Guideline 1082—An overview of AIS. Available online: <https://www.iala-aism.org/product/an-overview-of-ais-1082/> (accessed on 2 November 2021).
4. Plass, S.; Poehlmann, R.; Hermenier, R.; Dammann, A. Global Maritime Surveillance by Airliner-Based AIS Detection: Preliminary Analysis. *J. Navig.* **2015**, *68*, 1195–1209. [CrossRef]
5. Transportation Research Board. *Shipboard Automatic Identification System Displays: Meeting the Needs of Mariners—Special Report 273*; The National Academies Press: Washington, DC, USA, 2003; pp. 84–86.
6. Clazzer, F.; Munari, A.; Berio, M.; Blasco, F.L. On the Characterization of AIS Traffic at the Satellite. In Proceedings of the OCEANS 2014—TAIPEI, Taipei, Taiwan, 7–10 April 2014; pp. 1–9.
7. International Maritime Organization (IMO). *International Convention for the Safety of Life at Sea (SOLAS), Chapter V. Safety of Navigation*; International Maritime Organization (IMO): London, UK, 2002.
8. Norris, A. AIS Implementation—Success or Failure? *J. Navig.* **2007**, *60*, 1–10. [CrossRef]
9. Subash, T.D.; Pradeep, A.S.; Joseph, A.R.; Jacob, A.; Jayaraj, P.S. Intelligent Collision Avoidance system for fishing boat. In Proceedings of the International Conference on Advances in Material Science and Nanotechnology (ICMN), Kanyakumari, India, 29–30 April 2019; pp. 2457–2463.
10. Morten Bruun-Sørensen. Radar and AIS—Which One to Choose?—FURUNO Maritime Training. Available online: <https://www.furunotrainig.com/news-radar-and-ais-differences> (accessed on 9 January 2022).
11. Wu, B.; Cheng, T.; Yip, T.L.; Wang, Y. Fuzzy logic based dynamic decision-making system for intelligent navigation strategy within inland traffic separation schemes. *Ocean Eng.* **2020**, *197*, 106909. [CrossRef]
12. Li, L.; Lu, W.; Niu, J.; Liu, J.; Liu, D. AIS Data-based Decision Model for Navigation Risk in Sea Areas. *J. Navig.* **2018**, *71*, 664–678. [CrossRef]
13. Bakdi, A.; Glad, I.K.; Vanem, E.; Engelhardt, Ø. AIS-Based Multiple Vessel Collision and Grounding Risk Identification based on Adaptive Safety Domain. *J. Marit. Sci. Eng.* **2020**, *8*, 5. [CrossRef]

14. Chen, P.; Huang, Y.; Mou, J.; van Gelder, P.H.A.J.M. Ship collision candidate detection method: A velocity obstacle approach. *Ocean Eng.* **2020**, *170*, 186–198. [\[CrossRef\]](#)
15. Zhang, L.; Meng, Q. Probabilistic ship domain with applications to ship collision risk assessment. *Ocean Eng.* **2019**, *186*, 106130. [\[CrossRef\]](#)
16. Zhang, W.; Goerlandt, F.; Kujala, P.; Wang, Y. An advanced method for detecting possible near miss ship collisions from AIS data. *Ocean Eng.* **2016**, *124*, 141–156. [\[CrossRef\]](#)
17. Guo, S.; Zhang, X.; Zheng, Y.; Du, Y. An Autonomous Path Planning Model for Unmanned Ships Based on Deep Reinforcement Learning. *Sensors* **2020**, *20*, 426. [\[CrossRef\]](#) [\[PubMed\]](#)
18. Zhang, S.; Shi, G.; Liu, Z.; Zhao, Z.; Wu, Z. Data-driven based automatic maritime routing from massive AIS trajectories in the face of disparity. *Ocean Eng.* **2018**, *155*, 240–250. [\[CrossRef\]](#)
19. Yang, D.; Wu, L.; Wang, S.; Jia, H.; Li, K.X. How big data enriches maritime research—A critical review of Automatic Identification System (AIS) data applications. *Transp. Rev.* **2019**, *39*, 755–773. [\[CrossRef\]](#)
20. Matsumoto, H. Research on the Effective Use of Simple AIS Onboard Fishing Vessels. Ph.D. Thesis, Kobe University, Kobe, Japan, 2016.
21. Serra-Sogas, N.; O'Hara, P.D.; Pearce, K.; Smallshaw, L.; Canessa, R. Using aerial surveys to fill gaps in AIS vessel traffic data to inform threat assessments, vessel management and planning. *Mar. Policy* **2021**, *133*, 104765. [\[CrossRef\]](#)
22. Tanaka, T. AIS Usage in Various Regions of the World. Available online: <https://j-nav.org/mte/pdf/201404.pdf> (accessed on 9 January 2022).
23. Fujii, M.; Yamashita, K.; Urakami, M.; Wakabayashi, N. The study of simple navigation system for small craft using Class B AIS. In Proceedings of the OCEANS 2014—TAIPEI, Taipei, Taiwan, 7–10 April 2014; pp. 1–4.
24. Wolff, E. *Microservices: Flexible Software Architecture*; Addison-Wesley Professional: Boston, MA, USA, 2016.
25. Koschel, A.; Astrova, I.; Dötterl, J. Making the move to microservice architecture. In Proceedings of the 2017 International Conference on Information Society (i-Society), Dublin, Ireland, 17–19 July 2017; pp. 74–79.
26. Erl, T. *Service-Oriented Architecture: Analysis and Design for Services and Microservices*; Pearson: Boston, MA, USA, 2016.
27. Monteiro, D.; Maia, P.H.M.; Rocha, L.S.; Mendonça, N.C. Building orchestrated microservice systems using declarative business processes. *Serv. Oriented Comput. Appl.* **2020**, *14*, 243–268. [\[CrossRef\]](#)
28. Newman, S. *Building Microservices: Designing Fine-Grained Systems*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2021.
29. National Marine Electronics Association (NMEA). *NMEA 0183 Standard for Interfacing Marine Electronic Devices Version 3.01*; National Marine Electronics Association (NMEA): Severna Park, MD, USA, 2002.
30. International Organization for Standardization (ISO). Information Technology—The JSON Data Interchange Syntax (ISO/IEC Standard No. 21778). Available online: <https://www.iso.org/standard/71616.html> (accessed on 10 November 2021).
31. Docker. Docker Overview | Docker Documentation. Available online: <https://docs.docker.com/get-started/overview/> (accessed on 8 January 2022).
32. Liu, H. NoSQL-Based AIS Data Application. Master's Thesis, Dalian Maritime University, Dalian, China, 2019.
33. NTT DOCOMO. Communications/Areas | NTT DOCOMO. Available online: <https://www.nttdocomo.co.jp/english/area/> (accessed on 8 January 2022).
34. Liu, H.; Liu, Z.; Liu, D. Application of Machine Learning Methods in Maritime Safety Information Classification. In Proceedings of the 2018 Tenth International Conference on Advanced Computational Intelligence (ICACI), Xiamen, China, 29–31 March 2018; pp. 735–740.
35. Lopac, N.; Jurdana, I.; Lerga, J.; Wakabayashi, N. Particle-Swarm-Optimization-Enhanced Radial-Basis-Function-Kernel-Based Adaptive Filtering Applied to Maritime Data. *J. Mar. Sci. Eng.* **2021**, *9*, 439. [\[CrossRef\]](#)
36. Jurdana, I.; Lopac, N.; Wakabayashi, N.; Liu, H. Shipboard Data Compression Method for Sustainable Real-Time Maritime Communication in Remote Voyage Monitoring of Autonomous Ships. *Sustainability* **2021**, *13*, 8264. [\[CrossRef\]](#)
37. International Association of Marine Aids to Navigation and Lighthouse Authorities (IALA). IALA Guideline G1117—VHF Data Exchange System (VDES) Overview. Available online: <https://www.iala-aism.org/product/vhd-data-exchange-system-vdes-overview-1117/> (accessed on 26 January 2022).