

Marek A. Bednarczyk, Eliasz Haas, Piotr Kluk, Zygmunt Kitowski, Szymon Konefał, Jakub Neumann, Wiesław Pawłowski, Jerzy Pilecki, Agnieszka Przybyszewska-Chudzik, Tadeusz Puźniakowski, Sławomir T. Wierzchoń

Marek A. Bednarczyk

Polsko-Japońska Wyższa Szkoła Technik Komputerowych,
Zamiejscowy Wydział Informatyki w Gdańsku
Instytut Podstaw Informatyki PAN, Filia w Gdańsku
bednarczyk@pjwstk.edu.pl

Eliasz Haas

OPEGIEKA Elbląg

Piotr Kluk

Asseco Gdańsk

Zygmunt Kitowski

Polsko-Japońska Wyższa Szkoła Technik Komputerowych,
Zamiejscowy Wydział Informatyki w Gdańsku
Akademia Marynarki Wojennej w Gdyni

Jakub Neumann

Polsko-Japońska Wyższa Szkoła Technik Komputerowych, Zamiejscowy Wydział Informatyki
w Gdańsku
Uniwersytet Gdański, Wydział Matematyki, Fizyki i Informatyki

Wiesław Pawłowski

Instytut Podstaw Informatyki PAN, Filia w Gdańsku
Uniwersytet Gdański, Wydział Matematyki, Fizyki i Informatyki

Jerzy Pilecki

Instytut Badań Systemowych PAN

Agnieszka Przybyszewska-Chudzik, Tadeusz Puźniakowski, Szymon Konefał

Uniwersytet Gdański, Wydział Matematyki, Fizyki i Informatyki

Sławomir T Wierzchoń

Instytut Podstaw Informatyki PAN, Filia w Gdańsku
Uniwersytet Gdański, Wydział Matematyki, Fizyki i Informatyki

PLATFORMA ASVCASP

W pracy przedstawiono modułarny system sterowania i symulacji dla łodzi autonomicznych ASVCaSP (ang. Autonomous Surface Vehicle Control and Simulation Platform). W zależności od konfiguracji pozwala on na autonomiczne sterowanie łodzią, wspomaga planowanie i przeprowadzanie misji, umożliwia prowadzenie szkoleń oraz oferuje środowisko symulacyjne na potrzeby oceny algorytmów autonomicznego sterowania.

Słowa kluczowe: modelarny system sterowania, autonomiczne łodzie ASVCaSP.

1. WSTĘP

Głównym celem przedsięwzięcia było stworzenie uniwersalnej platformy, umożliwiającej realizację wielu zadań związanych z zarządzaniem ASV (ang. *Autonomous Surface Vehicle*). Platforma, poprzez odpowiednią konfigurację jej modułów może realizować następujące zadania:

1. Sterowanie łodzią w sposób autonomiczny. Platforma ma stanowić wysokopoziomowy system kontroli nad realizacją misji w trybie autonomicznym i przekazywać rozkazy do niskopoziomowego podsystemu, odpowiedzialnego za sterowanie fizycznymi urządzeniami łodzi. Ważną jej częścią są systemy gromadzenia informacji i fuzji danych z urządzeń takich jak np. GPS, AIS i Radar.

2. Oprogramowanie stanowiska dowodzenia. Platforma umieszczona poza łodzią ma stanowić centrum planowania misji. W jej skład wchodzi oprogramowanie wizualizujące bieżącą sytuację rejonu działań, z uwzględnieniem warunków pogodowych, jednostek widocznych w systemie AIS oraz definiowania zadań (misji) dla jednostki ASV.

3. Trener. Funkcjonalność zbliżona do zadań z pkt. 2. z tym, że działać ma w trybie symulacji, tj. może prezentować dane rzeczywiste lub sztucznie wymodelowane, w szczególności obraz z kamer.

4. Symulator do oceny algorytmów. Stanowi środowisko na potrzeby weryfikacji jakości algorytmów odpowiedzialnych za aspekty autonomicznego sterowania i realizacji misji. W ocenie pozwala na wykorzystanie zarówno predefiniowanych scenariuszy misji jak również danych rzeczywistych, np. pochodzących z systemu AIS.

WYMAGANIA DOTYCZĄCE OPROGRAMOWANIA

Możliwość realizacji tak wielu zadań spowodowała postawienie następujących postulatów dotyczących cech architektury systemu:

1. Otwartość: jednym z głównych problemów jest integracja wielu usług i technologii. Została ona osiągnięta dzięki zastosowaniu otwartej architektury, opartej o nowoczesne, a jednocześnie sprawdzone rozwiązania;

2. Przenośność: system ma pracować na wielu platformach sprzętowych takich jak łódź ASV, kontener ze stanowiskiem dowodzenia i sterowania oraz na stanowiska trenerskie;

3. Skalowalność: dotycząca głównie symulacji: system powinien realizować zarówno pojedyncze symulacje w czasie rzeczywistym jak i masowe, automatycznie powtarzalne;

4. Niezawodność: oprogramowanie przeznaczone do tego typu zadań musi charakteryzować się wysokim stopniem niezawodności. Osiągnięto go dzięki użyciu szeroko stosowanych i dobrze przetestowanych rozwiązań i bibliotek;

5. Modularność: ze względu na wielość celów, niezbędne było zapewnienie elastyczności wewnętrznej architektury systemu, gwarantującej możliwość wielokrotnego wykorzystania jej kluczowych elementów, a także łatwość zastępowania poszczególnych elementów alternatywnymi.

1. INTRODUCTION

The main objective of the undertaking was to create a universal platform enabling the realization of numerous tasks related to ASV (Autonomous Surface Vehicle) control. With proper module configuration, the platform may realize the following tasks:

1. Autonomous boat control. - the platform is to constitute a high level control system over mission realization in an autonomous mode and transfer commands to a low level subsystem responsible for the control of the boat's physical devices. Its key components consist of systems responsible for collecting information and merging data from such devices as e.g. GPS, AIS and Radar.

2. Command station software - the platform, located independently of the boat, is to serve as a mission planning centre. It includes software enabling the visualization of the current situation in an operational region, with the consideration of weather conditions, vessels detected by AIS system and task (mission) definition for an ASV.

3. Trainer - with functionality close to the tasks described in pt. 2, however available in the simulation mode, i.e. it may present real data or data artificially modelled, in particular pictures from cameras.

4. Algorithm assessment simulator - this creates an environment necessary for the verification of the quality of algorithms responsible for the aspects of autonomous control and mission realization. It allows for the assessment of both predefined mission scenarios and real data, e.g. from AIS system.

SOFTWARE REQUIREMENTS

The possibility of realization of a great deal of tasks resulted in the specification of the following postulates concerning system architecture:

1. Openness: one of the main issues consists in the integration of multiple services and technologies. This was achieved with the use of open architecture based on modern, yet verified solutions;

2. Portability: the system is to work on numerous equipment platforms such as an ASV, container with the command and control stations, as well as training stations;

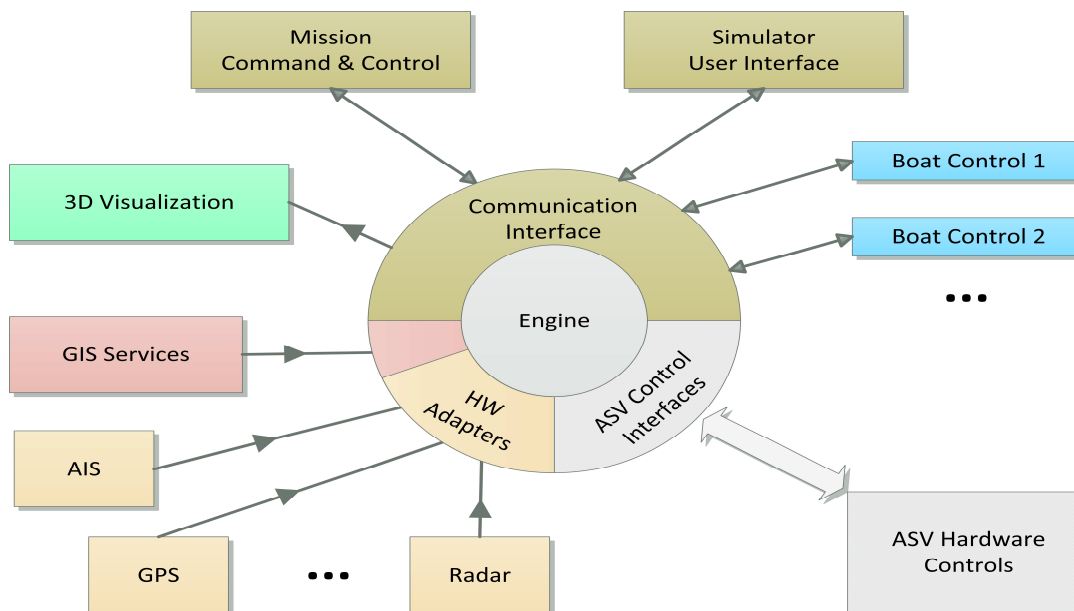
3. Scalability: *concerning mainly simulation:* the system should be able to realize both single simulations in real time as well as mass simulations, automatically repeated;

4. Reliability: the software created for this sort of tasks must be characterised by high reliability. This has been achieved with the use of widely applicable and well tested solutions and libraries;

5. Modularity: due to task plurality it was necessary to ensure flexibility of the system's interior architecture guaranteeing multiple use of its key elements, as well as the ease in the replacement of particular elements with alternatives.

2. ARCHITEKTURA SYSTEMU

Rysunek 1 prezentuje architekturę systemu z uwzględnieniem najważniejszych jego komponentów.



Rys. 1. Ogólna architektura systemu ASVCaSP.

Kernel – jądro systemu, stanowiące model środowiska, w którym odbywają się wszelkie obliczenia oraz gromadzone są dane, zarówno w trybie rzeczywistym, jak i w trybie symulacji. Współbieżna, wielowątkowa, a jednocześnie przejrzysta architektura modułu zapewnia wysoki poziom niezawodności i sprawność obliczeniową. Więcej informacji na temat implementacji modułu znajduje się w *Rozdziale 3*.

Communication interface – usługa umożliwiająca sterowanie modułem *Kernel*, przekazywanie do niego rozmaitych parametrów konfiguracyjnych dotyczących misji (*Mission Command & Control*), definiowanie symulacji (*Simulator User Interface*), udostępnianie stanu obliczeń na potrzeby symulowania obrazu z kamer (*3D Visualization*) oraz przyjmowanie informacji z cyfrowych urządzeń sterowania łodziami (*Boat Control*). Moduł bardziej szczegółowo opisany został w *Rozdziale 3*, poświęconym przyjętym rozwiązaniom technologicznym.

Mission Command & Control – moduł zarządzania i sterowania przebiegiem misji, definiowania scenariuszy i symulacji na potrzeby szkoleniowe oraz wizualizacji obszaru działań zgodnie ze standardami prezentacji przyjętymi w morskich elektronicznych urządzeniach wspomagających nawigację. Moduł ten wykonywany jest przez firmę Sprint i opisany w pracy [3].

2. SYSTEM ARCHITECTURE

Fig. 1 presents system architecture with the specification of its most important components.

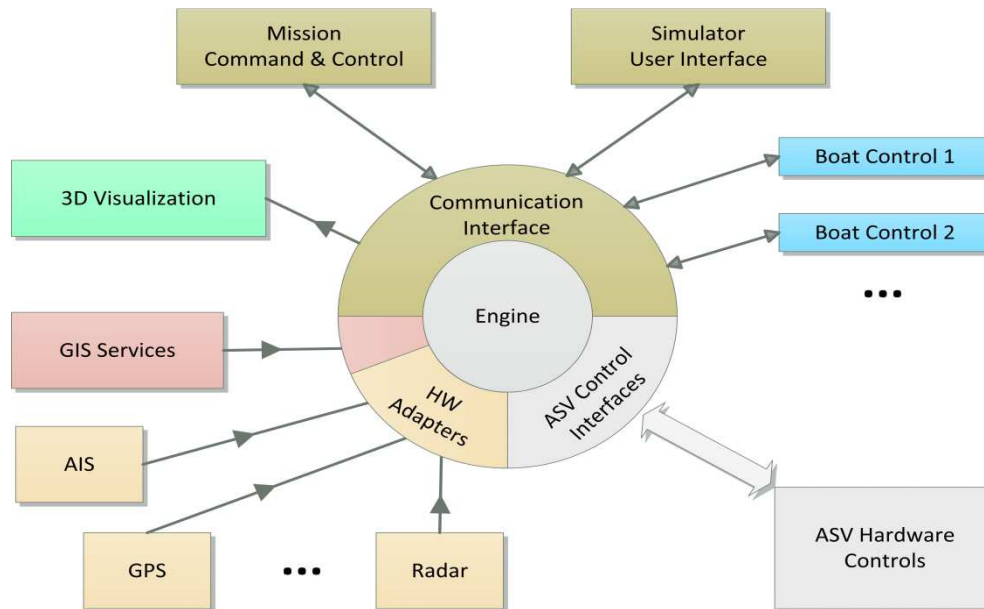


Fig. 1. General ASVCaSP system architecture.

Kernel – the core of the system constituting the system environment model where all the calculations are performed and data are collected, both in real mode and in simulation mode. The module's concurrent, varied and yet clear architecture ensures a high reliability level and computational competence. More information on the subject of module implementation has been presented in Section 3.

Communication interface – a service enabling control of the Kernel module, providing it with various configuration parameters concerning the mission (Mission Command & Control), simulation definition (Simulator User Interface), providing the state of calculations for the needs of camera picture simulation (3D Visualization), as well as obtaining information from digital boat control devices (Boat Control). A broader description of the module is available in Section 3 (devoted to the applied technological solutions).

Mission Command & Control – a module applied in mission command and control, scenario definition and simulation for training and visualization of an operational region (according to presentation standards used in the supporting marine navigation electronic devices). The module has been produced by the Sprint company and is described in article [3].

Simulator User Interface – moduł szczegółowego definiowania symulacji, umożliwiający edycję jej wewnętrznych parametrów, wizualizujący rozmaite warstwy usług GIS – zarówno własnych, jak i zewnętrznych – w szczególności siatkę nawigacyjną, sytuację pogodową, mapę rastrową, prezentujący stan symulacji. Bardziej szczegółowo został opisany w rozdziale 6.

3D Visualization – moduł trójwymiarowej wizualizacji przebiegu misji, symulujący obraz z kamer umieszczonych na ASV (dla celów szkoleniowych) lub na dowolnej jednostce biorącej udział w symulacji (np. dla celów oceny jakości algorytmów sterowania ASV). Bardziej szczegółowo opisany w rozdziale 5.

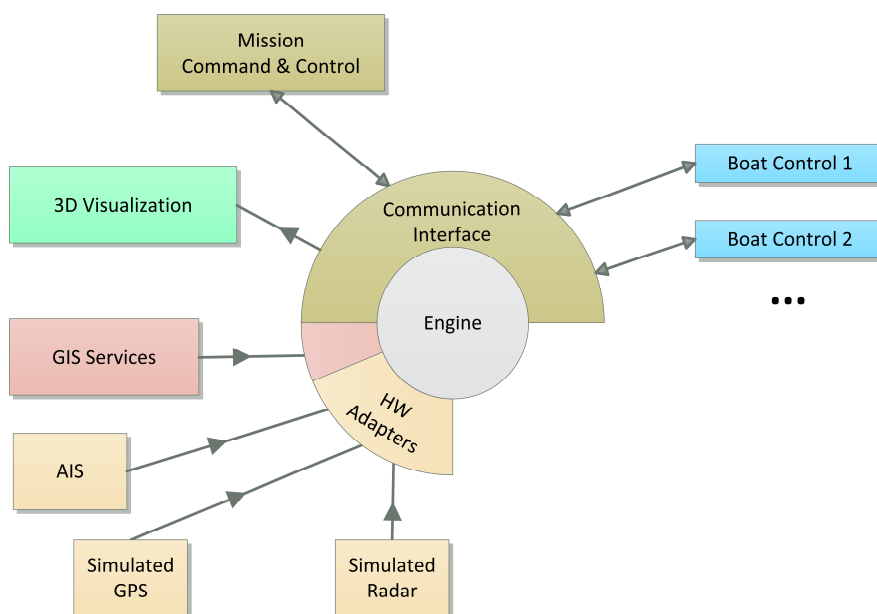
Boat Control – moduł reprezentujący urządzenia sterujące łodzią, pozwalający przejąć kontrolę zarówno nad ASV jak i innymi typami łodzi (w trybie symulacji). Nieco bardziej szczegółowo opisany w rozdziale 7.

GIS Services – moduł usług GIS udostępniający kluczowe dla systemu dane, takie jak siatka nawigacyjna oraz mapy rastrowe. Szczegółowo opisany został w rozdziale 4.

Hardware Adapters – moduły do pozyskiwania danych z rozmaitych urządzeń i źródeł, takich jak AIS, GPS lub Radar.

ASV Control Interfaces – moduły pośredniczące w przekazywaniu informacji o zamierzonym ruchu jednostki z symulatora do fizycznych urządzeń sterujących łodzią ASV. W opracowaniu AMW.

Modularność systemu pozwala na elastyczną konfigurację według potrzeb. Rysunek 2 przedstawia konfigurację modułów realizującą funkcjonalność trenażera.



Rys. 2. Moduły wchodzące w skład trenażera.

Podobnie, dalsze konfiguracje modułów definiują funkcjonalności pozostałych podsystemów zgodnie z ich opisami przedstawionymi we wstępie do pracy.

Simulator User Interface – a module for detailed simulation definition, allowing edition of internal parameters, visualizing various layers of GIS services – both own and external – in particular the navigational network, weather situation, raster maps and presenting the state of simulation. Its broader description is available in Section 6.

3D Visualization – a 3D mission visualization module simulating the picture from cameras placed on an ASV (for training purposes) or any vessel participating in the simulation (e.g. in order to assess the quality of ASV control algorithms). The module has been described in detail in section 5.

Boat Control – a module representing boat control devices. Boat Control facilitates taking control over an ASV as well as other boat types (in a simulation mode). Some more information on the module is provided in Section 7.

GIS Services – a GIS services module provides the system with key data, such as a navigational network or raster maps. A detailed description is available in Section 4.

Hardware Adapters – modules enabling obtainment of data from various devices and sources, such as AIS, GPS or Radar.

ASV Control Interfaces – modules taking part in transmitting information on the vessel's intended movement from the simulator to physical ASV control devices. Prepared by the Polish Naval Academy.

System modularity allows for its flexible configuration according to current needs. Fig. 2 presents module configuration in a trainer mode.

Similarly, further module configurations define the functionality of other subsystems in accordance with their description presented in the introduction – the Subject matter section.

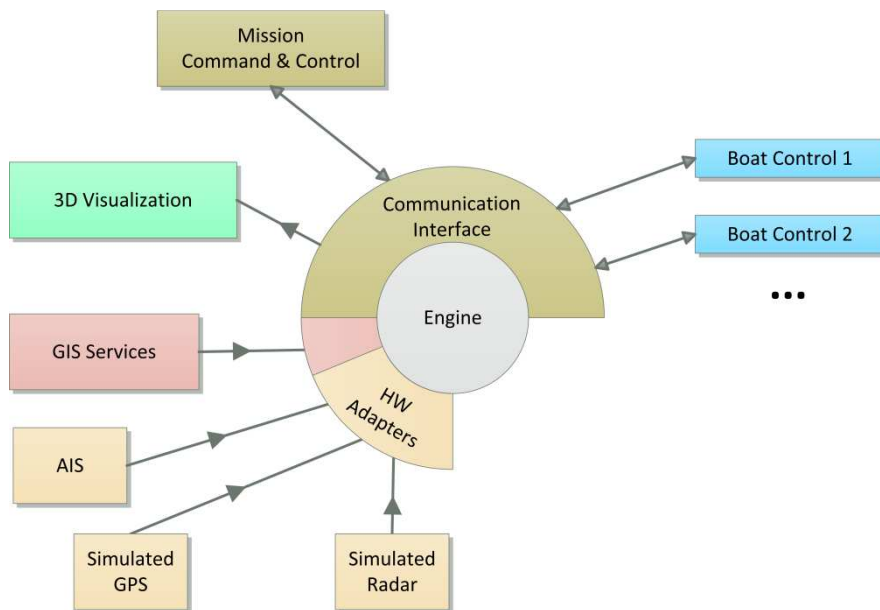


Fig. 2. Trainer modules.

3. PRZYJĘTE ROZWIĄZANIA TECHNOLOGICZNE

SCALA I AKTORZY

Zdecydowana większość modułów systemu ASVCaSP (poza klienckimi) napisana została w języku Scala [7]. Jest to nowoczesny język programowania łączący w sposób bardzo udany podejście obiektowe oraz funkcyjne. Wśród wielu zalet Scali na wymienienie zasługuje jej zgodność z językiem Java, pozwalającą na bezproblemowe korzystanie ze wszystkich bibliotek i narzędzi stworzonych w oparciu o ten język.

Scala oferuje bardzo interesujący model programowania równoległego, oparty na pojęciu aktorów (ang. actors) [5]. ASVCaSP bardzo intensywnie ten model wykorzystuje. Za pomocą aktorów modelowane są w szczególności wszystkie jednostki biorące udział w symulacjach oraz cała infrastruktura symulatora. Przepływ informacji wewnątrz systemu ASVCaSP odbywa się na zasadzie wymiany komunikatów pomiędzy odpowiednimi aktorami. Wykorzystanie Scali, z jej bogatym systemem typów oraz mechanizmem aktorów, pozwoliło osiągnąć założony od początku wysoki poziom modularności i skalowalności rozwiązania, a przejrzysta i ekspresywna składnia tego języka bardzo pozytywnie wpłynęła na niezawodność systemu.

REST JAKO ZEWNĘTRZNY KANAŁ KOMUNIKACYJNY

Komunikacja pomiędzy jądrem (Kernel), a komponentami klienckimi systemu ASVCaSP odbywa się za pośrednictwem usług REST [1], wykorzystując powszechnie znany i używany protokół komunikacyjny HTTP.

Użycie standardu REST umożliwiło zrealizowanie postulatu otwartości systemu i pozwoliło na łatwą i bezproblemową integrację z heterogenicznymi modułami klienckimi: *Simulator User Interface* (aplikacja webowa), *Mission Command & Control* (tradycyjna aplikacja) oraz *Boat Control* (aplikacja mobilna dla systemu Android).

Przykładowo, adresy usługi włączającej korzystanie z AIS oraz usługi zwracającej informację o stanie procesu symulacji mają odpowiednio postać:

- `http://{serwer:port}/rest/config/enableAIS`
- `http://{serwer:port}/rest/simulation/state`

Standard REST wykorzystywany jest przez platformę ASVCaSP również w kontaktach z serwerem GIS oferującym usługi mapowe.

4. USŁUGI MAPOWE (GIS)

Na potrzebę prezentacji danych o środowisku, w jakim odbywa się symulacja, w graficznym interfejsie użytkownika udostępnione zostały następujące treści:

- uproszczona postać elektronicznej mapy nawigacyjnej (ENC)
- obszar bezpiecznego manewrowania
- wizualizacja aktualnej prognozy pogody i stanu morza.

Dane przestrzenne w formie graficznej udostępnione zostały poprzez usługi mapowe (Map Service) oprogramowania ArcGIS Server [8]. Publikowanie usług mapowych w środowisku firmy Esri wymaga szeregu informacji: o warstwach udostępnianych przez daną usługę, ich stylistyce, widoczności w określonych przedziałach skali i wykorzystanych źródłach danych. Zarówno skomponowanie projektów mapowych dla poszczególnych usług, jak i uprzednie przygotowanie dla nich danych, wykonane zostały w środowisku ArcGIS Desktop.

3. ADOPTED TECHNOLOGICAL SOLUTIONS

SCALA AND ACTORS

The majority of ASVCaSP system modules (except for the customized ones) have been prepared in Scala language [7]. This is a modern programming language which successfully combines the approaches related to object and function. Among Scala's advantages we have to mention its compatibility with Java language allowing an unproblematic use of all the libraries and tools created on its basis.

Scala offers a very interesting model of parallel programming based on the notion of actors [5]. ASVCaSP uses this model intensely. In particular, actors are used in modelling all the vessels participating in simulations as well as the entire simulation infrastructure. The flow of information inside the ASVCaSP system is realized on the basis of message exchange between proper actors. The application of Scala, with the rich system of actor types and mechanisms led to the creation of the intended high modularity and scalability level of solutions, and the clear and expressive syntax of the language has a very positive effect on system reliability.

REST AS AN EXTERNAL COMMUNICATION CHANNEL

The communication between the Kernel and the customized components of the ASVCaSP system is realized through REST services [1], with the commonly known and used communication protocol: HTTP.

The application of the REST standard made it possible to realize the postulate of system openness and allowed a simple and unproblematic integration with the heterogeneous customised modules: Simulator User Interface (web application), Mission Command & Control (traditional application) and Boat Control (mobile application for Android system).

By way of example, the addresses of the service activating AIS and the service providing the feedback on simulation process take the following form:

- <http://{serwer:port}/rest/config/enableAIS>
- <http://{serwer:port}/rest/simulation/state>

The REST standard is also used by the ASVCaSP platform in contacts with the GIS server offering map services.

4. MAP SERVICES (GIS)

The graphic user interface has been provided with the following information for the purpose of presenting the data on the simulated environment:

- a simplified electronic navigational chart (ENC)
- a safe manoeuvre area
- a current weather condition and sea state visualization.

Spatial data in a graphic form are available through the use of the Map Service of the ArcGIS Server software [8]. Publishing of the map service requires the provision of information such as: the layers provided by a given service, their stylistics, visibility in particular scale ranges and the used data sources. Both the composition of map designs for particular services as well as the previous data preparation has been carried out in the ArcGIS Desktop environment.

USŁUGA PREZENTUJĄCA UPROSZCZONĄ POSTAĆ ELEKTRONICZNEJ MAPY NAWIGACYJNEJ

Danymi źródłowymi są wybrane w obszarze zainteresowania komórki pasma drogowego, podejściowego i portowego mapy elektronicznej. Dane w formacie S57 zaimportowane zostały do środowiska ArcGIS Desktop dzięki wykorzystaniu rozszerzenia Esri Nautical Solution. Dodatek ten pozwala na uzyskanie przybliżonej postaci ENC, przy czym sposób wizualizacji poszczególnych warstw definiowany jest automatycznie i zapisywany razem z zaimportowanymi danymi. Podejście takie pozwala na znaczne skrócenie czasu koniecznego na skonfigurowanie usługi, natomiast przy udostępnianiu danych dla dużego obszaru ma negatywny wpływ na szybkość renderowania obrazu.



Rys. 1. Obraz fragmentu portu gdyńskiego zwrócony przez usługę prezentującą uproszczoną postać elektronicznej mapy nawigacyjnej.

USŁUGA PREZENTUJĄCA OBSZARY BEZPIECZNEGO MANEWROWANIA

Na potrzebę symulacji dla wybranych głębokości wyznaczone zostały obszary bezpiecznego manewrowania. Dane do wyznaczenia tych obszarów pochodziły z tych samych komórek elektronicznej mapy nawigacyjnej, co w przypadku usługi uproszczonej wizualizacji. Ponieważ w tym przypadku nie była potrzebna wizualizacja danych, do ich importu do środowiska ArcGIS Desktop zastosowano rozszerzenie Data Interoperability. Jego zastosowanie pozwala na większą kontrolę niż w przypadku rozszerzenia Nautical, między innymi wierne odwzorowanie struktury danych oraz uwzględnianie lub nie poszczególnych plików uaktualnień danych S57.

Wyznaczenie obszarów bezpiecznego manewrowania wykonane zostało z wykorzystaniem środowiska ArcGIS Model Builder, pozwalającego na automatyzację wykorzystania narzędzi geoprzetwarzania poprzez budowę modelu, definiującego sekwencję operacji przetwarzania danych.

Generowanie obszarów bezpiecznego manewrowania odbywało się przy pomocy dwóch modeli. Pierwszy z nich pozwala na otrzymanie obszarów bezpiecznego manewrowania na konkretnym paśmie komórek S57 (drogowe, podejściowe, nawigacyjne).

Zasadę działania tego modelu opisać można następującą sekwencją działań:

SERVICE FOR PRESENTING A SIMPLIFIED VERSION OF AN ELECTRONIC NAVIGATIONAL CHART

The source data comes from selected cells of a navigational chart presenting road lanes, access ways and ports. Data in S57 format is imported into the ArcGIS Desktop environment with the use of the Esri Nautical Solution extension. This addition allows the obtainment of an approximate ENC format, whereas the manner of visualization of particular layers is defined automatically and recorded together with the imported data. Such an approach enables significant shortening of the time necessary for service configuration, however with the data concerning a larger area it has a negative impact on the picture rendering pace.

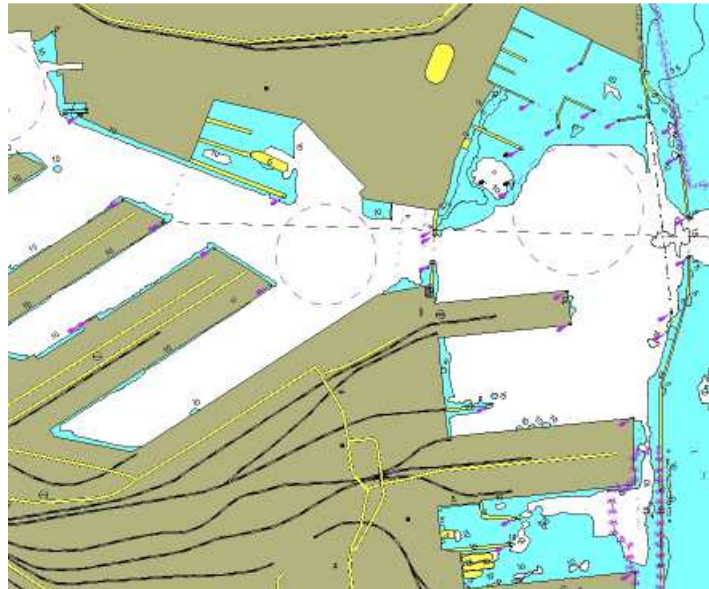


Fig. 3. A fragment of Gdynia port picture rendered by the service providing a simplified electronic chart.

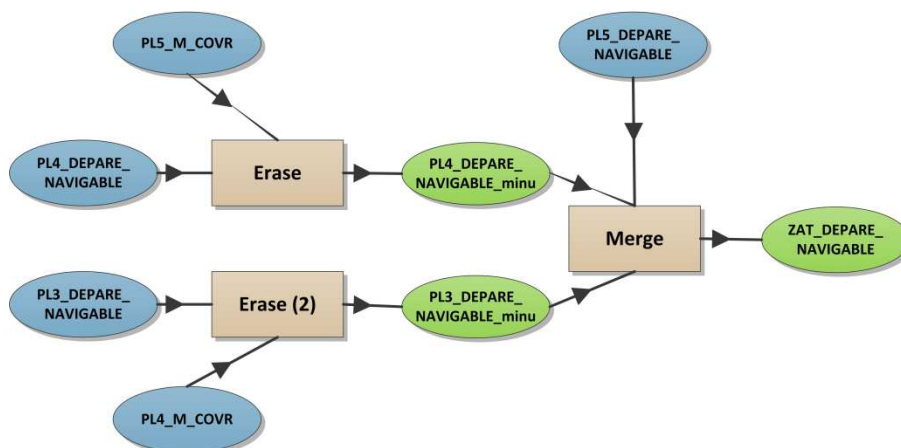
SERVICE FOR PRESENTING SAFE MANOEUVRE AREAS

For simulation needs, safe manoeuvre areas have been designated with regard to selected depths. Data used in the demarcating of such areas came from the same cells of an electronic navigational chart as in the case of the service concerning simplified visualization. Since in this case data visualization was not necessary for their importation into the ArcGIS Desktop environment, the Data Interoperability extension was used. Its application ensures a greater control as compared with the Nautical extension, a faithful reflection of data structure and enables making the decision whether to consider particular S57 data update files.

The designation of safe manoeuvre areas has been realized with the use of the ArcGIS Model Builder environment, allowing the automation of use of geo-processing tools by building a model defining the sequence of data processing operations.

Safe manoeuvre areas have been generated with the use of two models. The first of them allows obtainment of safe manoeuvre areas on a particular S57 cell range (roads, access ways, navigational cells). The principle behind the functioning of such a model may be described with the following sequence of operations:

1. Wybranie przeszkód nawigacyjnych,
2. Wygenerowanie buforów i uproszczenie geometrii tych buforów (wielkość bufora w zależności od charakteru przeszkody),
3. Zsumowanie przestrzenne wygenerowanych buforów jako warstwy przeszkód,
4. Odjęcie przestrzenne warstwy przeszkód od warstwy bazowej (w praktyce jest to suma obszarów głębokości, DEPART i innych, mniej istotnych jak obszary pogłębione, DRGARE). Drugi model pozwala na połączenie obszarów z trzech pasm w jeden zasób, w myśl zasady, że dane powinny posiadać możliwie wysoki poziom szczegółowości. Zasadę łączenia obrazuje Rysunek 4.



Rys. 2. Łączenie danych w obszarach bezpiecznego manewrowania z różnych pasm komórek S57.

Tak wyznaczona warstwa zawiera dane dla wszystkich głębokości. Wybór obszaru dla zadanej głębokości możliwy jest dzięki prostemu zapytaniu atrybutowemu (minimalna głębokość \geq zadana wartość). Ponieważ do symulacji wykorzystywana była plansza zdefiniowana przez siatkę nieregularnych trójkątów, konieczna była triangulacja obszarów. Przykłady wizualizacji siatki triangulacyjnej zawierają Rysunki 6 i 7.

USŁUGA PREZENTUJĄCA WIZUALIZACJĘ AKTUALNEJ PROGNOZY POGODY I STANU MORZA

Trzecia z usług prezentuje dane pogodowe i oceanograficzne, niezbędne do podjęcia decyzji o możliwości rozpoczęcia pracy przy rzeczywistym uruchamianiu autonomicznego trybu ASV.

Wykorzystane zostały dane o prędkości i kierunku wiatru przyziemnego, wysokości, okresie i kierunku falowania oraz kierunku i prędkości prądów morskich na pierwszym poziomie głębokości.

5. SYMULACJA OBRAZU 3D

Moduł służy do symulowania obrazu 3D z kamer. Obraz jest renderowany na podstawie stanu bieżącej symulacji oraz modelu otoczenia (np. okolic portu Gdynia).

WIZUALIZACJA TRÓJWYMIAROWA WYKONYWANEJ MISJI

Do trójwymiarowej wizualizacji wykonywanej misji wykorzystano rozwiązania ArcInfo oparte na bibliotekach udostępnionych przez produkt ArcScene.

1. Selection of navigational obstacles
2. Buffer generation and buffer geometry simplification (buffer size depending on obstacle nature)
3. Spatial summation of the generated buffers as an obstacle layer
4. Spatial subtraction of the obstacle layer from the base layer (in practice this is a sum of depth areas, DEPARE and other, less significant areas such as the deepened areas, DRGARE).

The other model allows for the combination of areas from three ranges into one resource following the rule that data accuracy should be as high as possible. Such a combination is presented in Fig. 4.

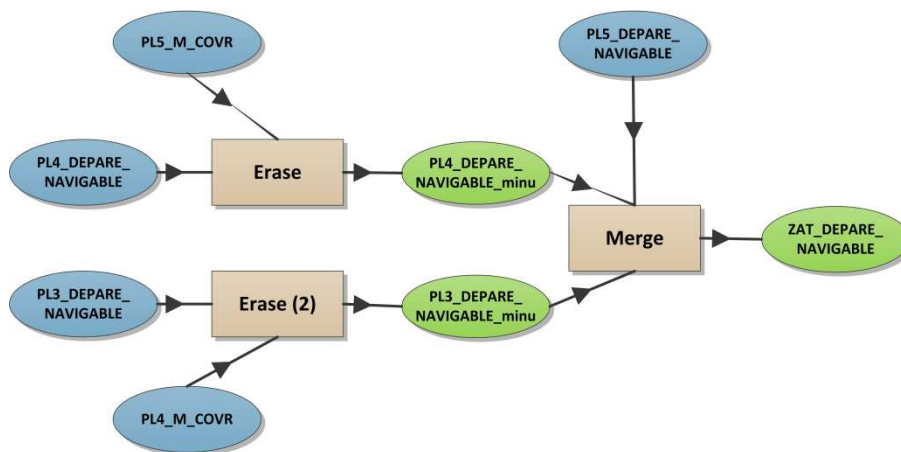


Fig. 4. Data combination in safe manoeuvre areas from various S57 cell ranges.

The layer designated in this manner encompasses data concerned with all the depths. Area selection for a particular depth is possible thanks to a simple attribute query (minimum depth \geq set value). Since a chart defined by a network of irregular triangles was used in the simulation, it was necessary to conduct area triangulation. Exemplary triangulation network visualizations have been presented in Fig. 6 and 7.

SERVICE PRESENTING THE VISUALIZATION OF CURRENT WEATHER CONDITIONS AND SEA STATE

The third service presents weather and oceanographic information necessary to take the decision on operation commencement with the activation of an autonomous mode in an ASV. The data in use include the information on the speed and direction of surface wind, the height, period and direction of waves, as well as the direction and speed of sea currents at the first depth level.

5. 3 D PICTURE SIMULATION

The module is used to simulate 3D pictures from cameras. The picture is rendered on the basis of the current state of simulation as well as the model of the surroundings (e.g. the surrounding of Gdynia port).

A 3-DIMENSIONAL SIMULATION OF THE MISSION IN PROGRESS

3D visualization of the mission in progress is created on the basis of ArcInfo solutions using the libraries available through ArcScene product.

Wybór ten podyktowany był faktem posiadania trójwymiarowego modelu portu w Gdyni zapisanego w geobazie systemu ArcInfo. Dane te pozyskano metodą lotniczego skaningu laserowego LIDAR, z którego otrzymana chmura punktów została poddana konwersji do postaci wektorowej.

Dane do wizualizacji pochodzą z kilku źródeł:

- model terenu i obiektów statycznych (budynki, nabrzeża, pozostała infrastruktura portu),
- informacje z symulatora o położeniu obsługiwanych przez niego obiektów ruchomych,



Rys. 5. Przykład trójwymiarowej wizualizacji misji.

- informacje pochodzące z systemu AIS.

Wizualizacja wykonywana jest z perspektywy autonomicznej łodzi realizującej misję. Aplikacja udostępnia 3 okna prezentujące symulowany obraz z trzech kamer. Położenie kamer można niezależnie zmieniać (wysokość, kąt, przybliżenie). Możliwe jest również przełączenie podglądu na dowolną jednostkę biorącą udział w symulacji.

6. MODUŁ SYMULACYJNY

Moduł *Simulator User Interface* to aplikacja webowa służąca do konfiguracji i obsługi części symulacyjnej systemu ASVCaSP. Aplikacja napisana została z wykorzystaniem biblioteki Dojo języka JavaScript [4] oraz komponentu GIS autorstwa firmy Esri [8].

Rysunek 6 obrazuje standardowy układ interfejsu użytkownika modułu symulatora. Panel konfiguracyjny (po lewej) umożliwia definiowanie zarówno parametrów samej symulacji, jak i sposobu jej obrazowania poprzez komponent GIS. W trakcie trwania symulacji możliwe jest ukrycie panelu konfiguracyjnego tak, aby cały obszar okna przeglądarki wypełnił komponent mapowy.

This choice stems from the fact that there is an available 3D model of Gdynia port recorded in the ArcInfo system geobase. The data have been obtained via LIDAR air laser scanning, from which the received cloud of points were converted into a vector form.



Fig. 5. An example of a 3D mission visualization.

Visualization data are derived from several sources:

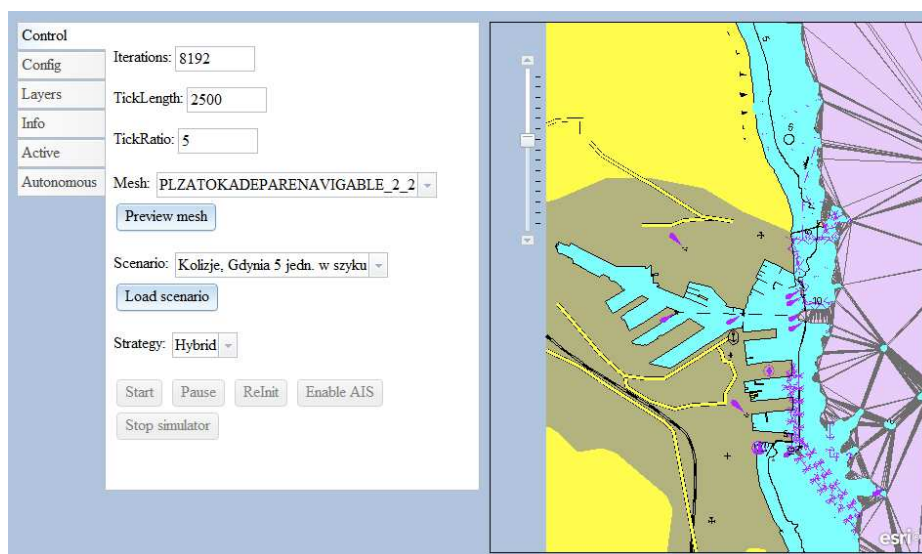
- the model of the area and static objects (buildings, quays, the remaining port infrastructure)
- simulator information on the location of the operated moveable objects
- information from AIS system

The visualization is realized from the perspective of an autonomous boat executing the mission. The application provides 3 windows presenting a simulated picture from three cameras. Camera parameters may be altered independently of each other (the height, angle, close-up). It is also possible to switch view to any other vessel taking part in the simulation.

6. SIMULATION MODULE

The Simulator User Interface module is a web application used in the configuration and operation of the simulation component of the ASVCaSP system. The application has been written with the use of the Dojo library of JavaScript language [4], along with a GIS component provided by the UK based company Esri [8].

Fig. 6 depicts a standard arrangement of simulator user interface module. The configuration panel (on the left) enables definition of both the parameters of the simulation itself as well as its display via the GIS component. In the course of a mission it is possible to hide the configuration channel so that the entire viewer window is reserved for the chart component.



Rys. 6. Interfejs użytkownika modułu symulatora.

Symulacje przeprowadzać można zarówno w czasie rzeczywistym, jak i w trybie przyspieszonego upływu czasu. „Przyspieszony upływ czasu” oznacza, że używana przez symulator długość jednostki czasu jest odpowiednim ułamkiem swojej wartości rzeczywistej (tzn. czas „płyne szybciej niż w rzeczywistości”). W skrajnym przypadku możliwe jest również prowadzenie symulacji tak szybko, jak tylko pozwala na to sprzęt, na którym działa system. Pozwala to przeprowadzać symulacje masowe.

Symulację możemy konfigurować i przeprowadzać w oparciu o 4 rodzaje jednostek:

- *pasywne* – pochodzące z predefiniowanego scenariusza,
- *autonomiczne* – realizujące misje z wykorzystaniem algorytmów sterowania ASV,
- *aktywne* – sterowanie nimi (w czasie rzeczywistym) odbywa się za pośrednictwem modułów *Boat Control*,
- jednostki, których dane pochodzą z systemu AIS (ich wykorzystanie ma sens wyłącznie wtedy, kiedy symulacja odbywa się w czasie rzeczywistym).

W procesie symulacji możemy wykorzystywać różne siatki triangulacyjne dla danego obszaru – np. w zależności od wymaganej dokładności danych batymetrycznych.

Kolejne dwa ważne parametry symulacji to *scenariusz* i *strategia*. *Scenariusze* (wspomniane już powyżej przy opisie rodzajów jednostek dostępnych w symulatorze) reprezentują „interesujące sytuacje morskie”, w kontekście których chcemy weryfikować zachowanie algorytmów sterowania łodzią autonomiczną. W nomenklaturze symulatora algorytmy te noszą nazwę *strategii*.

Rysunek 7 przedstawia sytuację, w której biorą udział trzy rodzaje jednostek: pasywne (pochodzące ze scenariusza), aktywna (sterowana ręcznie) oraz autonomiczna, wykonująca w danej chwili misję śledzenia jednej z jednostek pasywnych. Na rysunku zaobserwować możemy też siatkę triangulacyjną, z której korzysta algorytm sterowania łodzi autonomicznej.

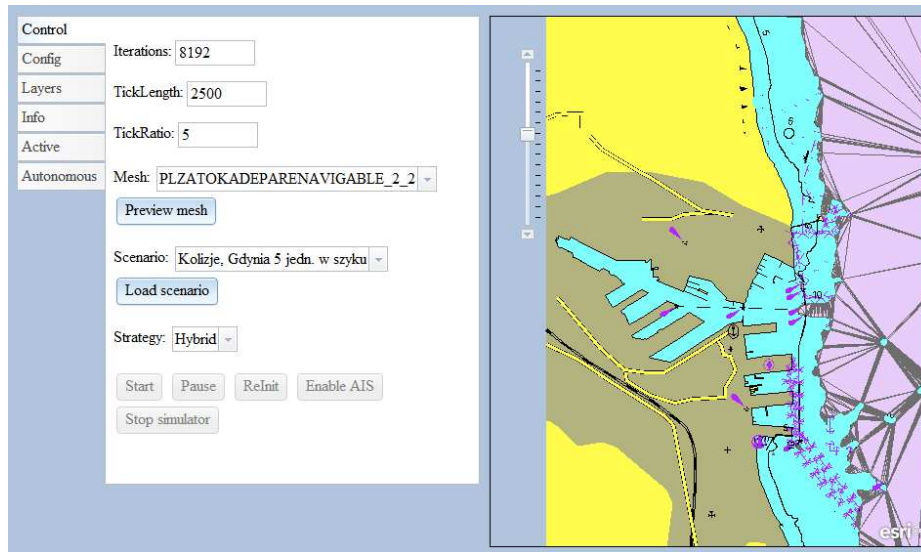


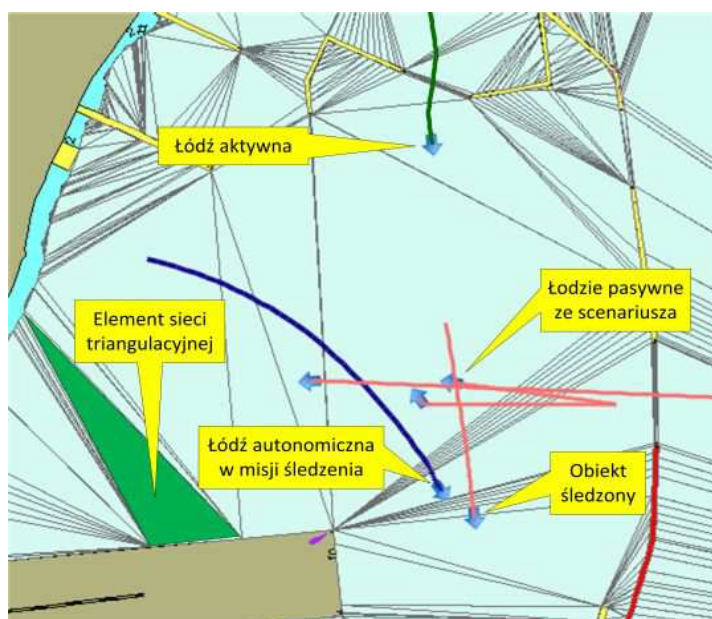
Fig. 6. Simulator user interface.

Simulations may be conducted both in real time and in an accelerated time mode. The “accelerated time mode” means that the time unit applied by the simulator constitutes a proper fraction of its actual value (i.e. the time “lapses faster than in reality”). In an extreme case it is also possible to perform simulations as fast as the equipment on which the system operates allows. It also enables carrying out mass simulations.

The simulation may be configured and conducted based on 4 vessel types:

- passive – from a predefined scenario
- autonomous – realizing missions with the use of ASV control algorithms
- active – controlled (in real time) with the use of the Boat Control modules
- vessels whose data are derived from AIS system (using them is reasonable only when the simulation is performed in real time).

Fig. 7 presents a situation with the presence of three vessel types: passive (from the scenario), active (manually controlled) and autonomous, realizing at a given time the mission consisting in tracking one of the passive vessels. The picture also shows a triangulation network that the autonomous boat control algorithm is using.



Rys. 7. Symulator w działaniu – przykład.

Interfejs symulatora pozwala na daleko idące modyfikacje sposobu obrazowania danych GIS – możemy aktywować lub dezaktywować warstwy mapy niosące różnego rodzaju informacje. Wśród dostępnych warstw są typowe warstwy GIS, warstwy AML, warstwy obrazujące warunki hydrometeorologiczne, jak i warstwy związane z przebiegiem symulacji (np. ślady tras jednostek).

Oczywiście wszelkie parametry i polecenia dostępne w module *Simulator User Interface*, nie związane z obrazowaniem danych GIS (za co odpowiada komponent Esri), możemy również definiować i przekazywać bezpośrednio do modułu *Kernel* za pomocą usług REST (np., moduł *Mission Command & Control* i sam *Simulator User Interface* też z nich korzystają).

7. BOAT CONTROL

Moduł ten jest aplikacją na urządzenia mobilne wyposażone w system operacyjny Android [6] (np. tablety, smartfony). W trybach symulacji oraz trenażera pozwala na przejęcie kontroli nad dowolną łodzią i sterowanie nią w sposób manualny. Interfejs aplikacji zobrazowany jest na Rysunku 8. Panel dotykowy umożliwia zmiany prędkości oraz kursu sterowanej jednostki i prezentuje ich aktualne wartości. Dodatkowo podaje identyfikator (numer) łodzi, którą sterujemy.

Podobnie jak moduły *Simulator User Interface* i *Mission Command & Control*, aplikacja kontaktuje się z jądrem systemu ASVCaSP za pośrednictwem interfejsu REST-owego. Przykładowo – do sterowania łodzią przeznaczone są usługi o adresach w postaci:

- <http://{serwer:port}/rest/activeBoat/changeSpeed/id/{boatId}/value/{value}>
- <http://{serwer:port}/rest/activeBoat/turn/id/{boatId}/value/{value}>

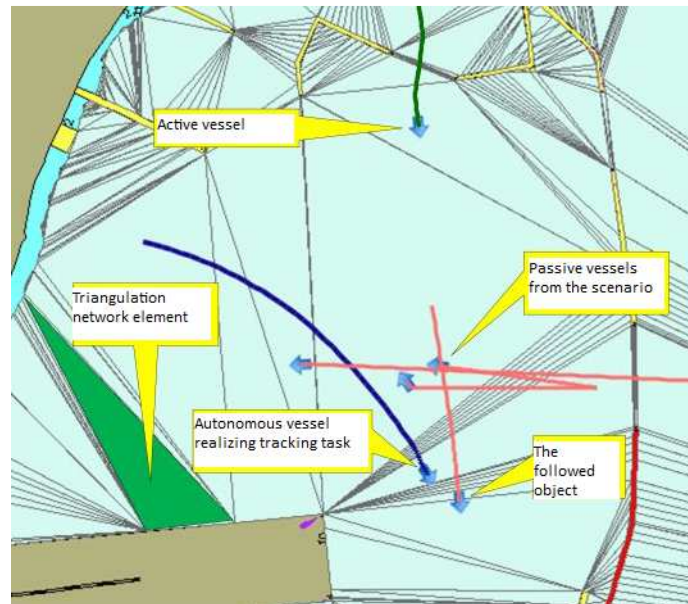


Fig. 7. Simulator at work – an exemplary situation.

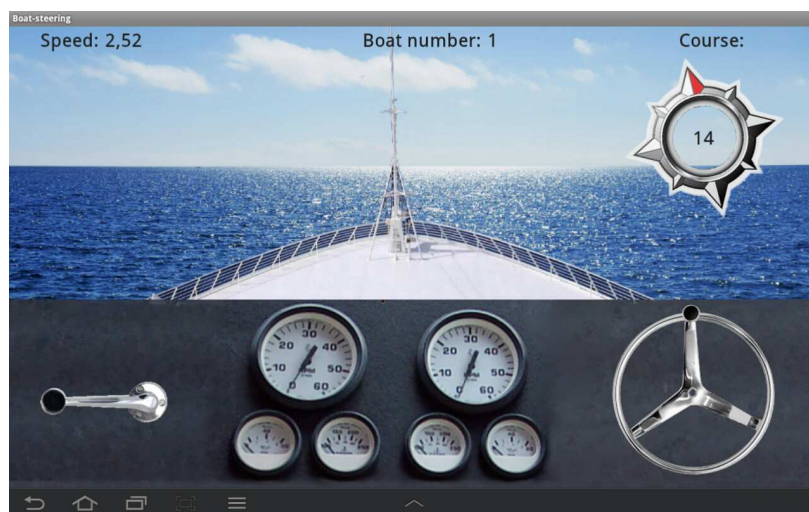
Simulator interface allows far-reaching modifications of the manner of presentation of GIS data – we may activate or deactivate chart layers providing various kinds of information. Among the available layers there are typical GIS layers, AML layers, those depicting hydro-meteorological conditions, as well as layers connected with the process of simulation (e.g. routes covered by vessels). Of course all the parameters and commands available through the Simulator User Interface modules, that are not related to GIS data presentations (for which the Esri component is responsible), may also be defined and transferred directly to the Kernel module via the REST services (the Mission Command & Control module and the Simulator User Interface also use the REST services).

7. BOAT CONTROL

This module is an application for mobile devices equipped in the Android operational system [6] (e.g. tablets, smartphones). In simulation and trainer modes it allows taking over a given boat and controlling it manually. Its interface is presented in Fig. 8. A touch panel enables speed and course modifications of the controlled vessel and displays their current values. Additionally it provides an identification number of the controlled vessel.

Similarly to Simulator User Interface and Mission Command & Control modules, the application contacts the ASVCaSP system kernel via the REST interface. As an instance – boat control is realized via services with the following addresses:

- `http://{server:port}/rest/activeBoat/changeSpeed/id/{boatId}/value/{value}`
- `http://{server:port}/rest/activeBoat/turn/id/{boatId}/value/{value}`



Rys. 8. Interfejs użytkownika modułu Boat Control.

Praca naukowa finansowana ze środków na naukę w latach 2010 – 2012 jako projekt rozwojowy Nr O R 00 0106 12.



Fig. 8. User interface of Boat Control module.

Scientific work financed from educational funds in the years 2010 – 2012 as a development project No. O R 00 0106 12.

LITERATURA/ BIBLIOGRAPHY

1. S. Allamarajuz, RESTful Web Services Cookbook: Solutions for Improving Scalability and Simplicity, O'Reilly 2011;
2. M.A. Bednarczyk, E. Haas, P. Kluk, Z. Kitowski, S. Konefał, J. Neumann, W. Pawłowski, J. Pilecki, A. Przybyszewska-Chudzik, T. Puźniakowski, S.T. Wierzchoń, Realizacja zadań autonomicznych przez ASV z wykorzystaniem metod sztucznej inteligencji;
3. A. Baranowski, M. Czepczyński, Działalność Sprint SA w ramach projektu rozwojowego pod nazwą Zintegrowany system planowania perymetrycznej ochrony i monitoringu morskich portów i obiektów krytycznych, oparty o autonomiczne jednostki pływające;
4. D. Flanagan, JavaScript: The Definitive Guide, 6th Edition, O'Reilly, 2011;
5. P. Haller, F. Sommers, Actors in Scala, Artima Press 2011;
6. R. Meier, Professional Android Application Development, Wrox 2008;
7. M. Odersky, L. Spoon, B. Venners, Programming in Scala, Second Edition, Artima Press 2010;
8. M. Price, Mastering ArcGIS, McGraw-Hill Science/Engineering/Math; 4th edition, 2009.

ASVCaSP PLATFORM

The article presents control and simulation system modules for autonomous boats: ASVCaSP (Autonomous Surface Vehicle Control and Simulation Platform). Depending on the configuration, the system allows autonomous vehicle control, supports mission planning and execution, facilitates the conducting of trainings and offers a simulated environment with the purpose of evaluation of autonomous control algorithms.

Key words: modular control system, ASVCaSP surface boats.

ПЛАТФОРМА ASVCaSP

В этой статье мы представляем собой модульную систему для управления и моделирования для автономных лодок ASVCaSP (ang. Autonomous Surface Vehicle Control and Simulation Platform). В зависимости от конфигурации, система позволяет для автономного управления лодкой, поддерживает планирование и выполнение миссии, обеспечивает подготовку и предлагает моделирование окружающей среды для оценки алгоритмов автономного управления.

ключевые слова: модульная система управления, автономные лодки ASVCaSP.