ZESZYTY NAUKOWE AKADEMII MARYNARKI WOJENNEJ
SCIENTIFIC JOURNAL OF POLISH NAVAL ACADEMY

2015 (LVI) 3 (202)

DOI: 10.5604/0860889X.1178570

Sylwester Kaczmarek*, Magdalena Młynarczuk*, Maciej Sac*

DATABASE FOR INTEGRATION OF INFORMATION IN DISTRIBUTED DATA EXCHANGE SYSTEM ELEMENTS OF BORDER GUARD

ABSTRACT

The paper presents the database solution for integration of information in distributed data exchange system elements of the Polish Border Guard. The proposed database solution is described in the context of data exchange system elements which control position and store identification data of vessels (fishing, sports and sailing boats) and other suspicious objects on the territorial sea, sea-coast and the internal sea-waters controlled by Maritime Polish Border Guard. The paper presents the general architecture of the data exchange system elements of Border Guard, the general MapServer architecture as a component of the Border Guard system. To explain the challenges being faced during implementation the conceptual model of database is also presented. Moreover main functionalities of database client and functional tests results are described.

Key words:
database, SQL, database client, exchange system, Maritime Border Guard.

INTRODUCTION

The Border Guard of the Republic of Poland is a unified, uniformed authority tasked with protection of the state border on land and at sea and with cross border traffic control. The Maritime Regional Unit of Border Guard in Gdańsk is the only one regional unit responsible both for land and blue border surveillance as well as for surveillance of the Polish maritime area [3].

* Gdańsk University of Technology Faculty of Electronics, Telecommunications and Informatics, Department of Teleinformation Networks, G. Narutowicza 11/12 Str., 80-233 Gdańsk, Poland; e-mail: {kasyl; magdam; Maciej.Sac}@ eti.pg.gda.pl

The maritime surveillance market offers specialized solutions for maritime area surveillance. The example is the integrated Vessel Traffic Management System (VTMS) designed to provide advanced solutions for the control and management of safe vessel traffic, maritime environment monitoring and Search & Rescue (SAR) operations support which is presented in [10]. The other integrated maritime safety systems operating in Poland and the European Union, their potential and ways of acting are fully described in papers [8, 9, 12, 13].

Currently the Polish Border Guard (BG) is equipped with the Automatic National System of Radar Control for Maritime Areas of Poland (zautomatyzowany system radarowego nadzoru polskich obszarów morskich — ZSRN) which develops and distribute complex surface image covering the territorial sea and the internal sea-waters [6]. Using to the system the BG observes the usage of Polish sea-waters, controls the suspicious vessels, detects undefined surface objects, participates in Search And Rescue operations. The ZSRN system supports protection against illegal immigration, narcotic, drug trafficking.

The system is composed of centers: Main Control Centre (CON), Reserve Control Centre (ZCON), Division Control Centers (DONs), Local Control Centers (LONs) and Observation Posts (OPs) distributed along the Polish coastline on the Baltic Sea. Additional parts of the system are Mobile Observation Posts (MOPs) [6].

Presented database solution is a part of MapServer solution which is a proposition for ZSNR system expansion. The concept of the MapServer was prepared in the scope of the project 'The concept and implementation of integration information in distributed data exchange system elements of Border Guard' which is co-financed by the National Centre for Research and Development.

The expansion comprises also consoles and Universal Radio Controllers (URCs) installed in MOPs which can move by sea, land or air. These mobile units are connected to designated OP by means of radio network and URC in OP relays MOPs communication to core BG's IP network.

In this approach it is assumed that central database is located in CON/ZCON and local databases are installed in MOPs in Universal Radio Controllers (URCs). The architecture of the system is presented in next section.

DATABASE IN BORDER GUARD SYSTEM

The architecture of distributed data exchange system elements of Border Guard proposed under the project [2] consists of:

- central MapServer with the central database which provides positions information of monitored objects (vessels, vehicles and other objects) on the territorial sea, sea-coast and the internal sea-waters of the Republic of Poland;

- Universal Radio Controllers (URCs) with mobile version of MapSever and local databases;
- consoles for operators: stationary (connected to the central database via broadband IP network) and mobile (connected to URC via radio network).

The central MapServer (fig. 1) collects in the central database objects information on the territorial sea, sea-coast and the internal sea-waters of the Republic of Poland. The source of information for central database is Web Service of Maritime Office which provides information from Automatic Identification System (AIS) tracks [11]. The Central MapServer with central database provides data (current or archival) to stationary consoles which are presented on maps [5] or in tables [4] according to console's operator demands.

The local databases in URC collect data concerned with objects on local area in range of local sources such as AIS receivers, GPS receivers, ARPA radars dependent of type of MOP. The mobile MapServer in URC provides data (current or archival) to mobile consoles attached to URC for visualization on maps and in tables. Data collected in local database in URC, apart from being stored in local database in URC, are transferred to central MapServer and integrated there with data from other MOPs and external source (Web Service). Communication between mobile MapServers and central MapServer is managed by URCs in MOP and designated OP and involves communication through digital radio channel and fixed IP network.

The functionality of mobile and stationary consoles is determined by Border Guard requirements.

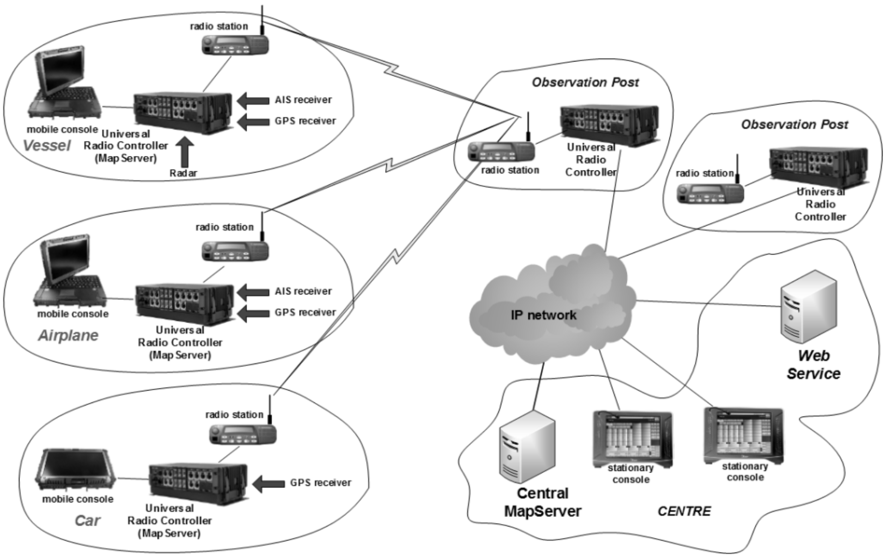

Fig. 1. The architecture of distributed data exchange system elements of Border Guard

The presented system also support voice transmission (by radio and IP), call recoding, data transfer between consoles, transfer radar's data from radar system in range of actual installed ZSNR. Additional elements of the system for supplying these functions are not depicted in figure 1 for the sake of the clarity of architecture.

General MapServer architecture

As depicted in figure 1 in distributed data exchange system elements of Border Guard MapServer is installed in two different variants. The Central MapServer is implemented in Centre and its mobile version is installed in URCs. The mobile MapServers in URCs are connected with Central MapServer through the radio channel using Dual Transfer Protocol (DTM). The general architecture of MapServer in both variants is similar but differs in offered functions. Comprehensive description of MapServer functionality is presented in [2]. In this paper only general information is presented, which is important in database aspect.

The MapServer in URC is composed of the following modules: communication module 1, communication module 2, preprocessing and deduplication, database client and the MapServer kernel. Communication module 1 (depicted in figure 2 as comm. module 1) provides communication functions (map #, data #) between MapServer and local consoles by sending messages to a specific service (Map Service or Data Service) in console attached to the URC. Communication module 2 provides communication functions between URCs and the Central MapServer. The processing and deduplication module provides data from AIS, ARPA and GPS sensors with eliminated redundant object data. The database client offers API (Application Programing Interface) for other MapServer modules to access database resources. The MapServer kernel in URC is responsible for coordination between MapServer elements for the purpose of transfer requests from mobile consoles and send required data indispensable for map visualization or table presentation. The general architecture of MapServer in URC with two mobile consoles attached is presented in figure 2.

The MapServer in Centre has a similar set of main modules to the mobile MapServer version in USR. It is composed of the following modules: communication module 1, communication module 2, preprocessing and deduplication, database client and the MapServer kernel. Communication module 1 provides communication functions (map #, data #) between MapServer and local consoles by sending messages to a specific service (Map Service or Data Service) in console attached to the Central MapServer. Communication module 2 provides communication functions between URCs and the Central MapServer. The processing and deduplication module provides data from Web Service with eliminated redundant object data. The database client offers API (Application Programing Interface) for other MapServer modules to access database resources. The MapServer kernel in Centre is responsible for coordination between MapServer elements for the purpose of transfer requests from stationary consoles and sending required data indispensable for map visualization or table presentation to the stationary console and on request to all mobile consoles connected indirectly through digital radio channel and IP network.

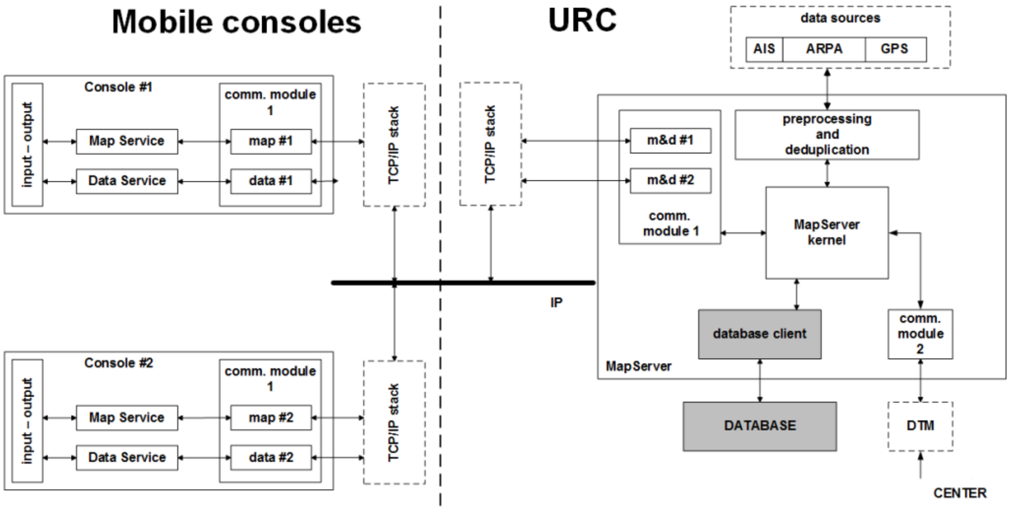

Fig. 2. The general MapServer architecture in URC with two mobile consoles attached [1]

Integration of information in distributed system elements of Border Guard is performed within synchronization process when local databases in URCs are integrated into central database in Central MapServer.

Taking into consideration the specific structure of the system, type of data sources (AIS, ARPA, GPS, Web Service) and radio channel limitations only position data are transferred into central database. The next section presents the structure of the database implemented in the Border Guard system.

Database structure

The database structure in the Border Guard system architecture is determined by source of information. In URCs source of information are ARPA radar, AIS or GPS receiver. The local database additionally maintains notes associated with selected objects which are manually added by the mobile consoles' operators.

The source of information for central database are: Web Service which support Automatic Identification System (AIS) and data transferred from URCs. Additionally, the central database stores notes added by operators of stationary consoles and mobile consoles (provided during synchronization sessions with URCs).

The information stored in the central database and in local databases in URCs could be grouped into four categories as follows [7]:

- permanent data concerned with identification of object (name of the ship, type of the ship, call sign, dimensions of ship);
- low frequency data represented by rarely changed data corresponding to object (information concerned with current destination port, navigational status);
- high frequency data represented by positions of objects and URCs (longitude and latitude), position accuracy, speed over ground, course of the ground, true heading, navigational status);
- additional data — notes added by operators of stationary and mobile consoles and data concerned with identification of consoles' operators.

It is assumed that the structure of the local database in URC is similar to the central database. The central database located in central MapServer in Centre and local database in URC maintain object information concerned with: vessels, merchant ships, passenger and ferries ships, boats, sport boats, scientific research and scuba diving vessels, border police vessels, buoys, wrecks. The category of object is determined on the basis on AIS messages or could be manually inserted by the user of console operators (notes). The conceptual model for database implementation of the system is presented in figure 3.

The conceptual model includes following entities:

→ Operator
→ Notatka
→ Obiekt
→ Map_poz_AIS
→ Dane_szczeg_AIS
→ Map_poz_ARPA
→ Map_poz_WebS
→ Dane_szczeg_WebS
→ Map_poz_GPS.

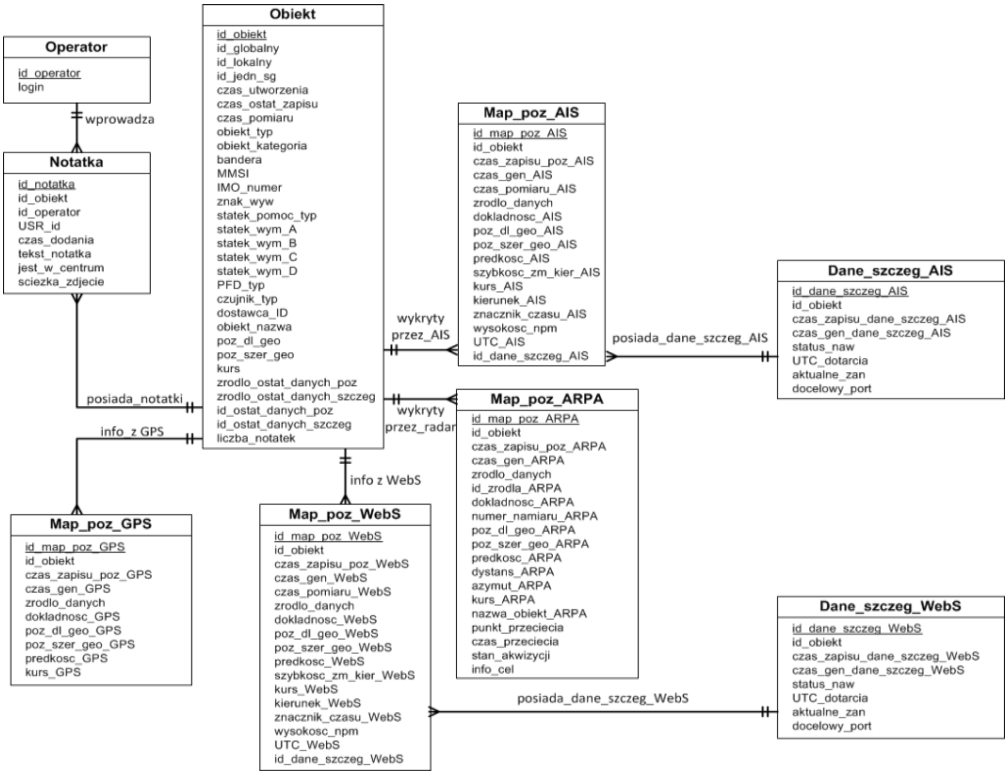

Fig. 3. The conceptual model of database for the Border Guard system

Data concerned with console's operator is represented in the Operator entity. The entity Notatka represents a note added by the console's operator. The Obiekt entity corresponds to object identification in the system. Map_poz_ARPA, Map_poz_AIS, Map_poz_GPS, Map_poz_WebS represent high frequency data from ARPA, AIS and GPS sensors as well as Web Service. Entities named Dane_szczeg_AIS and Dane_szczeg_WebS correspond to low frequency data from AIS and Web Service.

The transformation from conceptual model to Physical Data Model (PDM) is performed based on the destination platform, which is PostgreSQL 9.1 database management system installed on Linux platform [9].

The local databases in URCs and central database in Centre have tables to store data in the Border Guard system as follows:

→ Operator — stores consoles' operators identification information (id, logins)
→ Notatka — maintains notes added by the consoles' operators
→ Obiekt — consists of information corresponding to identification of object in Border Guard system especially object identifier, local identifier, global identifier,

Regional Unit of Border Guard identifier, time stamp when the data was generated and measured, MMSI (Maritime Mobile Service Identity) number, IMO number, call sign, ship dimensions, current latitude and longitude, source type of last position
→ Map_poz_AIS — maintains positions data from AIS receiver
→ Dane_szczeg_AIS — stores information provided by AIS receivers corresponding to navigational status, UTC time stamp of arrival, actual draught, destination port
→ Map_poz_ARPA — maintains positions data from ARPA receiver
→ Map_poz_WebS — maintains positions data from external source (Web Service)
→ Dane_szczeg_WebS — stores information provided by Web Service which corresponds to navigational status, UTC time stamp of arrival, actual draught, destination port
→ Map_poz_GPS — maintains positions data from GPS sensors.

In the remaining part of the paper database client implementation is presented.

Database Client

The database solution discussed in this paper is based on PostgreSQL 9.1 open source object-relational database system due to its important ACID compliance (Atomicity, Consistent Isolation Durability) [12]. The MapServer in URC and in Centre is implemented in C++ language on Linux platform.

The database client provides API (Application Programing Interface) for other MapServer modules to access database resources. The role of API is played by DatabaseClient class facilitating database client methods which take advantage of libpq library functions.

Different client methods allow to write data in local and central databases, to transfer data from local database into central database in synchronization process, to receive the results of queries for mobile and stationary consoles for data presentation on map and data presentation in tables. Objects from the DatabaseClient class called DbClient are separate for each thread using database client. Parallel access of MapServers modules to database resource is possible due to transaction mechanism. The transaction is initialized by invoking function StartDbAccess with appropriate status (read, write), afterwards database client method or methods of transaction are called. Transactions terminate when FinishDbAccess is invoked.

DatabaseClient class facilitates database client methods for inserting into database and updating objects information from AIS, ARPA, GPS sensors and Web

Server. DatabaseClient class distinguishes methods for communication Map Service-MapServer in purpose of presentation of the following data on map [1]:
- current positions of object (CurrentScreenRequest service);
- archival positions of objects (ArchivalScreenRequest service);
- trails for current object or objects trails (CurrentTrailsRequest service);
- presentation of detailed object information (CurrentDetailsRequest service).

Moreover, specific database client methods for communication Data Service-MapServer in purpose of presentation of data in tables are implemented. The methods for table presentation support:
- presentation of current list of objects (CurrentListRequest service);
- presentation of detailed information for object with defined parameters (SearchObjectsRequest service);
- adding note to object (CreateNoteRequest service);
- selection and demonstration of note from database (ShowNoteRequest service).

Additional methods are implemented for synchronization process and deduplication. For example, specific GetNeighboringObjectsTracks method supports selection from database all tracks for declared amount of objects in determined time.

FUNCTIONAL TESTS

In order to verify the implemented and installed database solution, a set of functional tests has been executed. The tests were performed for two stationary consoles attached to Central MapServer and two mobile console attached to URC. The mobile MapServer in URC and Central MapServer were implemented under Linux platform on the equipment designed and manufactured under the grant [2] on real and simulated data. The mobile and stationary console were implement on Windows platform.

Performed test scenarios included validation of database client methods in:
- communication Map Service-MapServer for stationary and mobile consoles;
- communication Data Service-MapServer for stationary and mobile consoles;
- communication local database-central database;
- data service for deduplication modules in mobile MapServer in URC and in Central MapServer.

During execution of the tests the following implemented database client's methods have been checked for proper operation in aspect of the following services: CurrentScreenRequest, CurrentDetailsRequest, CurrentTrailsRequest, ArchivalScreenRequest, ArchivalDetailsRequest, CurrentListRequest, SearchObjectsRequest, CreateNoteRequest, ShowNoteRequest.

The database client methods responsible for inserting and updating objects information from AIS, ARPA, GPS sensors and Web Service have been verified. The methods dedicated for deduplication operations have been also checked.

The figure 4 presents the example of CurrentScreenRequest service and CurrentDetailsRequest service. In figure 4 apart from current object presentation, current details for two objects are presented in tables. In these tables a flag, IMO number, MMSI number, call sign, global identifier of object in whole system, local identifier, type of object, category of object, longitude, latitude, course, measurement time and type of data source are displayed. The object named as 'TAURUS' has also defined speed and draught.

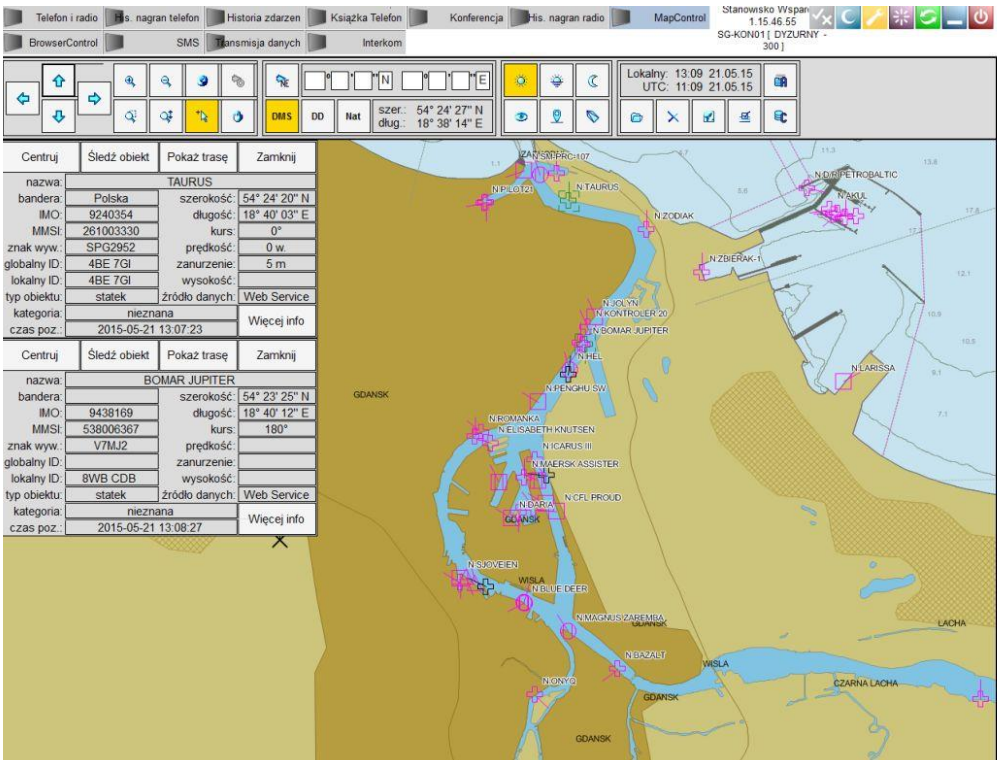

Fig. 4. The example of stationary console's service of CurrentScreenRequest with object details displayed

The figure 5 presents CurrentScreenRequest service, CurrentDetailsRequest service and CurrentTrailsRequest service. The trail is depicted for object which name is SG-112.

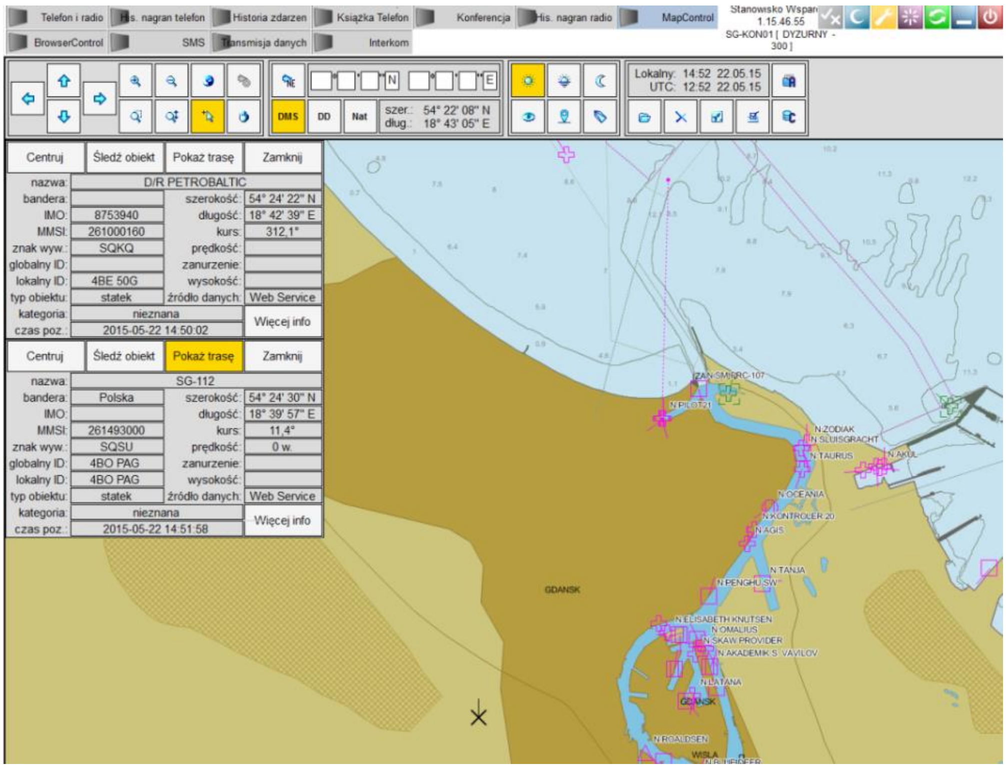

Fig. 5. The example of stationary console's service of CurrentScreenRequest with object details displayed and track presentation

The figure 6 presents the example of CurrentListRequest — the request was generated by mobile console's operator. In figure 6 the information concerned with number of notes corresponding with the object is also displayed.

All performed test scenarios confirmed correctness of database client methods implementation.

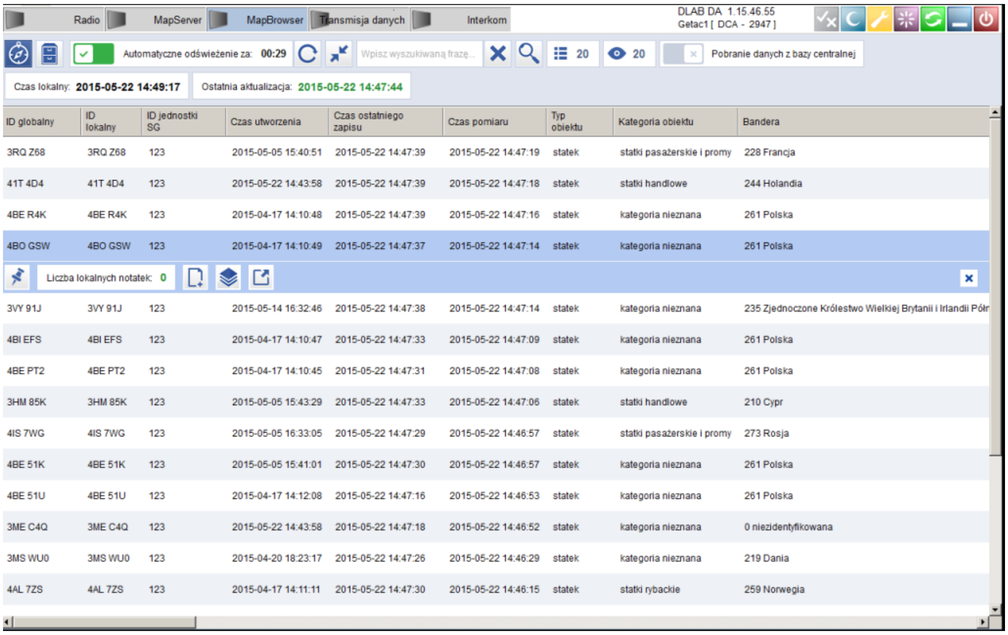

Fig. 6. The service of CurrentListRequest on mobile console connected to USR with note information

SUMMARY

The database is one of the most important components of any kind of large-scale system, especially for the system for the Border Guard. Land and marine border surveillance requires dedicated database to provide and store information concerned with vessels (fishing, sports and sailing boats), vehicles and other suspicious objects on the territorial sea, sea-coast and the internal sea-waters of the Republic of Poland.

The approach presented in the paper meets the requirements of the large-scale distributed computing environment, which provides scalability, high availability, high performance and reliability. Performed functionality tests confirmed that the PostgresSQL database implemented in system architecture is appropriate as a database for the Border Guard for the integration of information in distributed data exchange system elements. The ACID model ensures strict data consistency of transaction processing while using an exclusive lock mechanism.

The presented solution (local database in URC and central database in Centre) with implemented synchronization process, which transfers objects position data from local database to central database, provides integration of information in distributed data exchange systems elements of Border Guard.

Acknoledgements

This work has been co-financed by NCBiR, projects DOBR/0022/R/ID1/2013/03 and DOB-BIO6/10/62/2014.

REFERENCES

[1] Blok M., Czaplewski B., Kaczmarek S., Sac M., *Messages, structures and scenarios for communication console — local database and communication local database — central database,* report of project DOBR/0022/R/ID1/2013/03, Gdańsk, March 2014 [available in Polish].

[2] Blok M., Kaczmarek S., Młynarczuk M., Sac M., *The Map Server software*, report of project DOBR/0022/R/ID1/2013/03, Gdańsk 2015 [available in Polish].

[3] *Colonel Karol Bacz Maritime Regional Unit of Border Guard*, [online], http://www.morski.strazgraniczna.pl/eng/, [access 17.04.2015].

[4] Cwalina K., Kaczmarek S., *Data visualization in tables (BrowserControl)*, report of project DOBR/0022/R/ID1/2013/03, Gdańsk 2015 [available in Polish].

[5] Czaplewski B., Kaczmarek S., *Data visualization on maps (MapControl)*, report of project DOBR/0022/R/ID1/2013/03, Gdańsk 2015 [available in Polish l].

[6] Fiorini M., Maciejewski S, *Lesson Learned During the Realization of the Automated Radar Control System for Polish Sea-water*, 'Marine Navigation and Safety of Sea Transportation', 2013, pp. 217–221.

[7] Kaczmarek S., Młynarczuk M., *Assumptions and conception of database realization*, report of project DOBR/0022/R/ID1/2013/03, Gdańsk 2014 [available in Polish].

[8] Kopacz Z., Morgaś W., Urbański J., *European Vessel Traffic Monitoring and Information System*, 'Zeszyty Naukowe AMW', 2007, No. 2, pp. 41–58 [available in Polish].

[9] *PostgreSQL: Documentation: 9.1: PostgreSQL 9.1.14 Documentation*, [online], http://www.postgresql.org/docs/9.1/interactive/index.html, [access 18.04.2015].

[10] *Selex ES*, [online], http://www.selex-es.com/documents/737448/12431511/body_mm07821_VTMS_LQ_pdf, [access 17.04.2015].

[11] *Technical characteristics for an automatic identification system using time-division multiple access in the VHF maritime mobile band*, ITU-R Recommendation, M.1371-5, February 2014.

[12] Vieira M., Casmiro Costa A., Madeira H., *Timely ACID Transactions in DBMS*, [online], http://www.di.fc.ul.pt/~casim/papers/dsn04-fa/dsn04-fa.pdf, [access 18.04.2015].

[13] Wróbel K., *Integrated Maritime Safety Systems*, 'Scientific Journal of Gdynia Maritime University', 2009, No. 63, pp. 95–104 [available in Polish].

BAZA DANYCH DLA POTRZEB INTEGRACJI INFORMACJI W ROZPROSZONYCH ELEMENTACH SYSTEMU WYMIANY DANYCH STRAŻY GRANICZNEJ

STRESZCZENIE

W artykule zaprezentowano bazę danych dla potrzeb integracji informacji w rozproszonych elementach systemu wymiany danych Straży Granicznej. Proponowane rozwiązanie bazy danych zostało przedstawione w kontekście elementów systemu umożliwiających kontrolę położenia i przechowanie danych identyfikujących statki (rybackie, sportowe, żaglówki) i inne podejrzane obiekty na morzu terytorialnym, na brzegu morza i w obszarze morskich wód wewnętrznych patrolowanych przez Morski Oddział Straży Granicznej. Przedstawiono ogólną architekturę systemu wymiany danych Straży Granicznej i ogólną architekturę MapServera — kluczowego elementu systemu wymiany danych. Istotę problemu i wyzwania, jakim należało sprostać, poparto modelem konceptualnym bazy danych. W artykule zaprezentowano również główne funkcje klienta bazy danych oraz wyniki testów funkcjonalnych.

Słowa kluczowe:

baza danych, SQL, klient bazy danych, system wymiany, Morska Straż Graniczna.