



STRADAR — MULTIMEDIA DISPATCHER AND TELEINFORMATION SYSTEM FOR THE BORDER GUARD

Marek Blok¹, Bartosz Czaplewski², Sylwester Kaczmarek³,
Jacek Litka⁴, Marcin Narloch⁵, Maciej Sac⁶

Gdansk University of Technology, Faculty of Electronics, Telecommunications and Informatics, G. Narutowicza 11/12 Str., 80-233 Gdańsk, Poland; e-mail: mblok@eti.pg.edu.pl, bartosz.czaplewski@pg.edu.pl, kasy@eti.pg.edu.pl, {jlitka; marcin.narloch; macsac}@pg.edu.pl; ORCID ID: ¹0000-0002-8793-1697, ²0000-0001-7904-5567, ³0000-0003-2932-5610, ⁴0000-0002-3538-4118, ⁵0000-0002-7640-2941, ⁶0000-0002-6734-3046

ABSTRACT

Security of national borders requires utilization of multimedia surveillance systems automatically gathering, processing and sharing various data. The paper presents such a system developed for the Maritime Division of the Polish Border Guard within the STRADAR project. The system, apart from providing communication means, gathers data, such as map data from AIS, GPS and radar receivers, videos and photos from camera or audio from phone calls and SMS messages, from multiple mobile units and stationary points located along the coastline. All gathered data can be visualized on a multidisplay by the personnel that can dynamically select presented data and configure form of presentation. The paper describes system functionalities, its hardware and software implementation, and presents results of acceptance tests that the system have recently passed.

Key words:

surveillance system, maritime border monitoring, dispatch system, multimedia, data visualization.

Research article

© 2019 Marek Blok, Bartosz Czaplewski, Sylwester Kaczmarek, Jacek Litka,
Marcin Narloch, Maciej Sac
This is an open access article licensed under the Creative Commons
Attribution-NonCommercial-NoDerivatives 4.0 license
(<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

INTRODUCTION

The main task of the Border Guard is protection of the country's border which requires control over the border traffic in order to prevent illegal immigration and cross-border criminal activities such as terrorism, smuggling goods, drug trafficking or illegal fishing [6, 11–14]. Additionally, on the maritime border, the Border Guard participates also in search and rescue operations. All these tasks are not trivial and require real-time collection, sharing and analysis of multiple data related to observed objects and ongoing operations of the Border Guard, which is not possible without an equipment allowing for communication and data acquisition, visualization and analysis. An example of a system designed for the needs of the maritime border protection is the currently utilized by the Polish Border Guard integrated security system named the Automatic National System of Radar Control for Maritime Areas of Poland (*Zautomatyzowany System Radarowego Nadzoru Polskich Obszarów Morskich — ZSRN*) [6, 8].

The STRADAR project [2, 4] presented in this paper, is a distributed multimedia surveillance and dispatch system which was developed based on the KONSOLA project [1] which was designed as an extension to the ZSRN system which allowed for bidirectional communication and map data exchange with mobile BG units. The STRADAR project is the next step further and apart from map data, from AIS, GPS and radar systems, it incorporates support for gathering, archiving and providing access to multimedia data such as camera video data and photos as well as audio data including phone calls, SMS and data files. The concepts behind the STRADAR system are compatible with current tendencies in development of border guard advanced surveillance and information sharing systems which aim at integration of data from multiple sources, including multimedia data, which allows for additional analysis and automatic detection of anomalies [5, 9, 17].

Since the Poland is a member of the European Union (EU) the Polish Border Guard plays also an important role in protection of the EU's borders. Moreover, the headquarters of the European Border and Coast Guard Agency (FrontEx), which is responsible for coordination of operations of border guards of the EU's members, are located in Warsaw. In order to carry out its tasks, the FrontEx agency exploits the European Border Surveillance (EUROSUR) monitoring system that allows for a (nearly) real-time presentation of current situation [15, 16]. The Eurosur creates the overall image based on digital data gathered by the different monitoring systems to which Eurosur has access. This additionally increases need for the development of the advanced surveillance data managements system that would allow for efficient



cooperation with the Eurosur system allowing improved situational awareness and increased reaction capabilities at external borders. As the STRADAR system incorporates the Event Visualization Post (EVP) operating on tasks related to suspicious activities, it is well suited for cooperation with the Eurosur system.

In the next section the main functionalities of the STRADAR system are indicated while in the section 'Hardware and software of the system' some details of the hardware and software structure are given. Since the presented system has already passed the acceptance tests, the tests and their results are described in section 'Acceptance tests', where more details about system functions and operation can be found. In section 'MapServer processing and wait times measurements' more detailed analysis of system performance, based on measured wait and processing times, is described together with propositions for future system improvements. Finally, in the last section the paper is summarized.

SYSTEM FUNCTIONALITY

The overall diagram of the STRADAR system is presented in fig. 1. The system gathers position data from AIS, GPS and radar as well as video and photos from cameras installed on BG's Mobile Units (MUs) and in Observation Points (OPs), audio data including phone calls received by the BG's officers as well as SMSs and exchanged data files.

The system is accessed by the operators from stationary consoles, where tactical situation is monitored, or mobile consoles where operators located on MU can analyze tactical situation related to the current task of a particular MU. The consoles are part of telephony-dispatching system and operators can communicate using them. Additionally, the STRADAR system introduces the EVP where the operator can visualize and monitor multiple tactical situations using maps supplemented with related multimedia data in order to improve the overall situation awareness. Additionally operators can add notes to any of the available data as well as create task notification for the EVP which can group several data related to observed suspicious activity.

The basic map data that the operators can visualize on maps and in tables are provided by the MapServer software [1, 3] available in the Central Server (CS) (in case of stationary consoles and the EVP) and URCs (in case of mobile consoles). The available map data is gathered by the MapServers (central and mobile) which form a distributed network collecting map data from different sources. All map data



gathered by the local MapServers located at MUs are transferred to the central MapServer during synchronization sessions. That way the central MapServer has direct access to all map data available in the system, which means that it can additionally monitor them and detect anomalies such as abnormal data changes or duplicate data associated to the different objects.

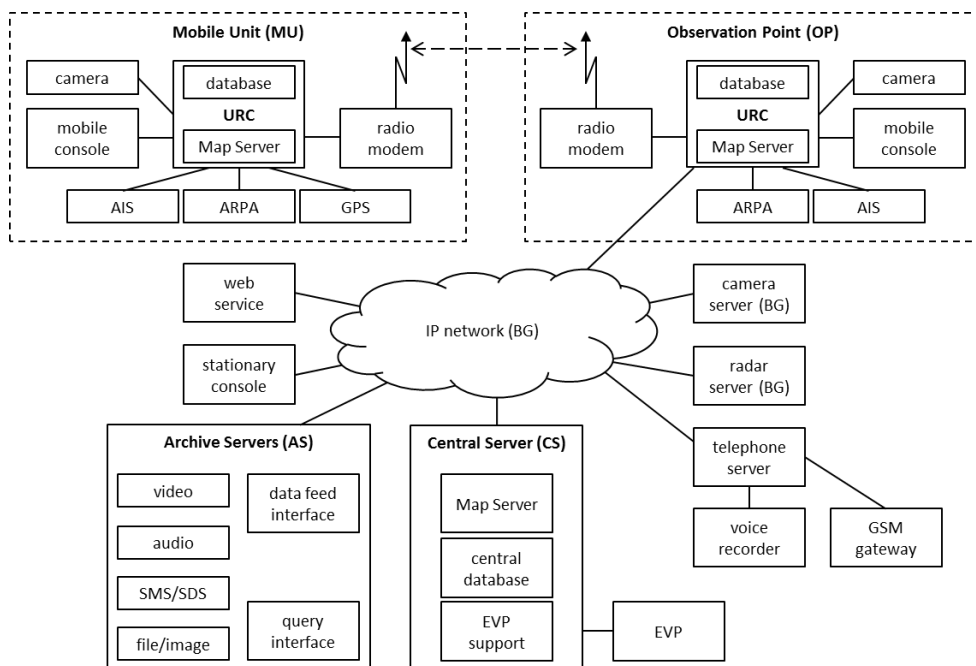


Fig. 1. Diagram of the multimedia distributed system (STRADAR)

All other data, the multimedia information that represent significantly larger volume, is stored in the Archive Servers (AS) that are based on cloud technology, which allow for data search and access. In order to control and facilitate access to the resources located in AS, the central MapServer provides the EVP support that manages tasks notifications and multimedia access, for example current or archival view from the camera located on a particular MU [3, 4].

HARDWARE AND SOFTWARE OF THE SYSTEM

Each MU is equipped with URC, mobile console, IP camera, radio link modem and a set of sensors providing map data for the system (particularly ARPA radar, AIS

and GPS receivers). The URC in MU is a custom build system developed by the DGT company (a member of the STRADAR project consortium). The URC system has a modular construction and is protected by a military class chassis (fig. 2a). It can work in unfavourable weather conditions and can be used in land vehicles, airplanes and sea vessels. The following interfaces are available in the URC: 4 RS-232 ports, 2 USB 2.0 ports, 1 diagnostic RS232 console port, 1 external Gigabit Ethernet port, Built-in 5 port Fast Ethernet Switch (4 ports are available), 2 AUDIO ports. The sources of map data can be connected to the URC RS-232 or USB ports with appropriate serial port converters.

The software installed in the URC consists of the Debian Linux 7 'Wheezy' operating system (for ARM processors), the MapServer application (version for MU), the PostgreSQL database, VoIP SIP telephony server and other communication as well as system management applications.

A mobile console is a military class laptop (fig. 2b) with touch screen, which is connected to URC's local IP network via built-in Ethernet switch. Its software mainly consists of the Windows 7 operating system and the DGT Dispatcher Application, which offers graphical user interface for the operators and, with the support of proper modules developed in the C# .NET language, provides particular functionalities of the mobile console (data visualization on maps, radiocommunications, telephony, etc.).

The OPs are equipped similarly to MUs and contain: URC, IP camera, radio link modem and a set of sensors providing map data for the system. In each OP a rack-mount variant of the URC (designed for standard working conditions) is installed, which runs the dedicated version of the MapServer application, with functionality limited to map data gathering and synchronization with CS, and optional possibility of mobile consoles attachment.

The Central Server (CS) of the STRADAR system is a rackmount server with high performance Intel Xeon processor, 32 GB of RAM and two 2TB hard disks. The software installed in the CS includes the Debian Linux 8 'Jessie' operating system, the MapServer application (CS version), the PostgreSQL central database as well as other communication and system management applications.

The variant of the MapServer installed in the CS is the most complex in the whole STRADAR system. It gathers map data from MUs, OPs and Web Service, stores them in the central database after removing duplicates and provides them to consoles and EVP. It also includes the EVP Support Module, which allows creating tasks for visualization of multimedia and map data in the EVP and coordinates the communication between the EVP and the AS for fetching proper types of multimedia data.



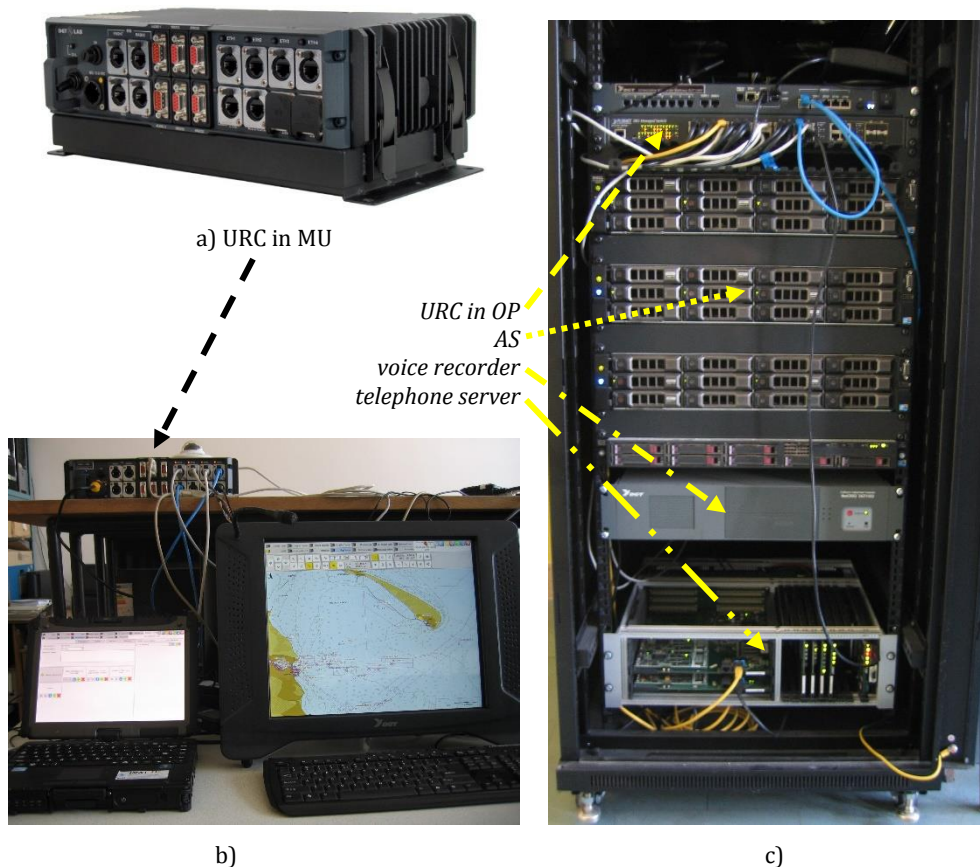


Fig. 2. Selected elements of the demonstrator of the STRADAR system:
a) URC in MU, b) consoles (mobile left, stationary right), c) AS

Stationary consoles are all-in-one PCs developed by the DGT company (fig. 2b) with touch screens and a set of software similar to mobile consoles. Due to the fact that stationary consoles communicate directly with the CS using the IP network of BG, they have slightly different configuration of the DGT Dispatcher Application and its modules.

The EVP (fig. 6) is based on a PC with a high resolution multi-screen display and a custom application for simultaneous visualization of different types of multimedia data with synchronization in time. The EVP PC includes a high performance Intel Core i7 processor, 32 GB of RAM and two graphic cards: integrated Intel HD Graphics for operator's management screen and a dedicated one with AMD Radeon Chipset (Club3D Radeon 7850 2 GB Eyefinity 6) for a multi-display visualization screen. To construct the visualization screen the AMD Eyefinity 6 technology was

used, which allows up to six physical displays connected via miniDisplayPort interfaces to be recognized as a single high resolution logical display in the EVP operating system (Windows 8.1 Pro 64-bit). The EVP application was developed using the C# .NET language with division of its functions into management module, visualization module and communication module. In the visualization module the concept of visualization controls (separate software modules) was applied with different controls responsible for presentation of map data, files/images, SMS, video and audio.

The AS (fig. 2c) are responsible for receiving, interpreting and collecting data from external systems and providing them on demand. The types of the collected information include map data, files/images, SMS, video and audio. The set of the elements, which exchange information with the AS, consists of among others the CS, URCS, EVP, voice recorder and consoles. For fast and reliable access to gathered information the AS was developed as an independent system based on the BigData technology with distributed services, load balancing and several copies of the stored data in different locations. The AS system comprises three identical rackmount Dell R510 servers. Each of them is equipped with 2 Intel Xeon X5675 processors (24 logical processors), 96 GB of RAM and 5 hard disks (3 SSD disks for gathered data, 2 HDD disks in RAID1 mode for software). The software of each server includes the Debian Linux 8 'Jessie' operating system, noSQL Elasticsearch 2 database, nginx HTTP server, ActiveMQ queuing system and additional application modules developed in the Java language. The elements of the AS software performing specific functions are launched in separate virtual environments (Linux Containers, LXC).

In the STRADAR system there are three elements providing functionalities related to IP-based voice communication (VoIP) between users: the telephone server, the voice recorder and the GSM gateway. The heart of the applied solution is the rackmount DGT miniMillenium telephone server, which supports standardized SIP and DGT proprietary SUIP communication protocols. The SIP protocol is used for handling standard VoIP telephones and for communication with the GSM gateway, which allows connections to public telecommunication network.

The voice recorder (DGT NetCRR2) is a rackmount device connected directly to one of the Ethernet ports of the telephone server. It is controlled by the telephone server and can record voice calls established using both SIP and SUIP signalling protocols. Recorded calls are stored into two internal hard disks configured in RAID1 mode and can be accessed remotely by the AS.

The distinguishing feature of the presented system is distributed processing of all types of information, including the data obtained from geographically distributed sensors (located in OPs and MUs) and data regarding real time voice communication



between personnel. Final information presented to the operators on EVP or consoles screens is the result of proper acquisition, storing and processing of that data in multiple elements of the distributed system. Thus, communication between all presented elements play a vital role in the STRADAR system.

Communication in the STRADAR network is based on the IP protocol making this system 'all IP' or 'pure IP' system (fig. 1), consisting of small wired LAN islands built around URCs in MUs, connected by broadband IP packet based radio links via radio modems in OPs (with their own dedicated LANs built around their URCs) to the core network and main system elements in the CENTER (CS, AS, EVP, VoIP telephony servers, stationary consoles) attached to a wired LAN. Due to fact that applicable broadband packet radio link was developed in concurrent project STRUGA [7, 10], emulator of a radio link based on Ethernet LAN was utilised in the STRADAR demonstrator network to connect distinct parts of the system with option to mimic the behaviour of IP radio link. Building small LAN around URC is greatly simplified by the circumstance that in every URC in any type of chassis exists a built-in Ethernet switch.

IP based networking in the project allows utilisation of almost any form of application communication software called the 'middleware'. The communication software in the STRADAR project is based on the Message Oriented Middleware (MOM) paradigm with Advanced Message Queuing Protocol (AMQP) as an application layer transport protocol used for exchanging map data. AMQP is an open standard, initially developed by a consortium of major financial and IT companies, which since 2012 is OASIS standard and since 2014 is ISO/IEC standard. In the STRADAR project AMQP based communication is used between the MapServer software and software modules called controls being basic elements of the EVP software and Dispatcher Application in consoles. Moreover, AMQP based communication is used between MapServers in MUs or OPs and the central MapServer in CS.

Another interface between software entities is communication between the AS and MapServer, which utilises Hypertext Transfer Protocol (HTTP) as a general transport protocol for middleware. That form of communication is well suited for a scenario where requests in the JavaScript Object Notation (JSON) format from MapServer as a HTTP client are send to the AS which responds with appropriate information playing a role of a sophisticated application server, providing archival and current data also in the JSON format. In the scenario where current data are obtained from the AS, a form of information subscription for MapServer is organised. In that scenario the AS asynchronously responds for the HTTP request and constantly provides information for the MapServer (and later for EVP and consoles) via internal



ActiveMQ broker queues, which is retrieved by the AMQP protocol when needed, till the subscription is cancelled.

Yet another protocol suite utilised in the STRADAR system is the communication with surveillance cameras with Ethernet/IP interfaces. In that case Real Time Streaming Protocol (RTSP) is used to control the cameras operation and Real-time Transport Protocol (RTP) is used to retrieve video data in order to store it in the AS and provide it to the operator via the EVP visual interface.

RTP is also used to transport real time data in VoIP services, particularly in telephony communication between Border Guard operators. In the STRADAR system every console is equipped with earphones and microphone and play a role of a VoIP telephone with telephony server utilising Session Initiation Protocol (SIP) for control of multimedia sessions. In that application VoIP telephony services are similar to classical telephony from the Border Guard operator perspective.

ACCEPTANCE TESTS

Before the end of the project, the acceptance tests were held. The goal of the acceptance tests was to verify the functionality of the implemented system. Test scenarios covering the full functionality of the system have been prepared. Representatives of the consortium, i.e. Gdańsk University of Technology and DGT sp. z o.o., as well as representatives of the gestor, i.e. the Polish Border Guard, were present during the acceptance tests. Employees of the Gdańsk University of Technology and DGT performed each single test, and Border Guard employees judged the test result and decided to pass the test.

The test environment for the acceptance tests included all devices from the STRADAR system, i.e.: Central Server, URC on the mobile unit, URC at the observation point, Events Visualization Post, Stationary console (including in it: MapControl software, BrowserControl software, TaskGeneratorControl software, NavalAnomalyControl software), Mobile console (including in it: MapControl software, BrowserControl software, TaskGeneratorControl software).

During the tests, real-time data on naval objects on sea were used, which were received from two AIS receivers with antennas of different range and one GPS device. In order to visualize data from the ARPA radar, a simulator was used to generate radar packets. Video recordings came from two IP cameras, one of which was connected to the URC on the mobile unit, and the other one to the URC at the observation



point. Other data, i.e. audio recordings, SMS messages, files as well as static images, were generated during the tests using the stationary console, mobile console, EVP, and mobile phones.

The tests have been conducted according to the schedule. A list of scenarios to be tested has been devised to test cooperation between the functional elements of the system. The scenarios covered a wide number of events in which the STRADAR system could be utilized to its full extent. Scenarios have been divided into four groups. Each group of scenarios focused on a single entity in the system. Those groups were:

1. *Telecommunication Services* — group of tests that validated the services allowing for making phone calls and setting up phone conferences between consoles connected to the STRADAR system.
2. *Mobile Console* — group of tests that validated the functionality of Mobile Consoles connected to the STRADAR system.
3. *Stationary Console* — group of tests that validated the functionality of Stationary Consoles connected to the STRADAR system.
4. *EVP* — group of tests that validated the functionality of Events Visualization Post connected to the STRADAR system.

Each group consists of a number of scenarios, each presenting a feature of the element's functionality. The list of example scenarios for consoles and EVP is as follows:

- for *Mobile Console*: Map loading and configuration, Displaying objects on map, Displaying windows with object's details, Displaying tracks of the objects, Displaying archived situation on map, Operations on current map objects, Displaying data on archived objects, Creation of text notes, Task generation utilizing element forms, Task generation utilizing macros, Utilizing task history;
- for *Stationary Console*: All scenarios of Mobile Console, Utilizing list of naval objects anomalies, Displaying details on naval objects anomalies, Searching for naval objects anomalies;
- for *EVP*: Task generation utilizing element forms, Task generation utilizing macros, Utilizing task history, Presentation of tasks in the form of a list, Accepting task from the list (archived data), Accepting task from the list (subscription), Declining task from the list, Visualization of map data, Visualization of browser data, Visualization of file/image, SMS, video, and audio data, Visualization with time synchronization, Utilizing list of naval objects anomalies, Displaying details on naval objects anomalies, Searching for naval objects anomalies.



In this part of section, an example of the test conducted on the system is provided. The test's objective was to verify the process of generating a tasks using element forms on a Stationary Console and sending them to EVP for visualization. During the test two tasks have been created. One consisting of two elements (map data element and image data element), second consisting of a single video data element. The test proceeded as follows. On the mobile console the form of creation of a new task has been loaded (as presented in fig. 3).

An element of type map has been chosen to be added as an element of the task. A set of parameters, e.g. geographical coordinates and objects filtering options, has been provided to define search requirements.

After that image data has been added and another set of parameters, e.g. a creation time range and a text note, has been defined.

This formulated task has been sent do EVP for visualization. After that, a second task has been procured. This task consisted of a single video, it was added via video query form. This time a set of search criteria consists only of a creation time range. The example of new task generation is presented in fig. 4.

This task has been sent to EVP. On the EVP both tasks have been noticed on the Management Screen as accepted for visualization. After the acceptance of the tasks, the metadata on the documents of interest has been found in the Archival Server and send to the EVP. Next, the visualization of both tasks started which is presented in fig. 5 and 6.

Fig. 3. New task creation form

After displaying the multimedia, a number of functionalities has been checked, including, but not limited to: forwarding the video, pausing the video, changing of its volume, zooming in and out of the picture, resizing the picture, resizing the map and tracking of an object on the map. The positive result of the test required for the EVP to visualize all the data from the tasks sent from console. The EVP must have been able to manage the video recording, map and image and allow the operator to work on them with the functionality provided by the multimedia control software. All of the tested data, i.e. naval objects on map, video stream and an image, have been synchronized in time during the visualization.

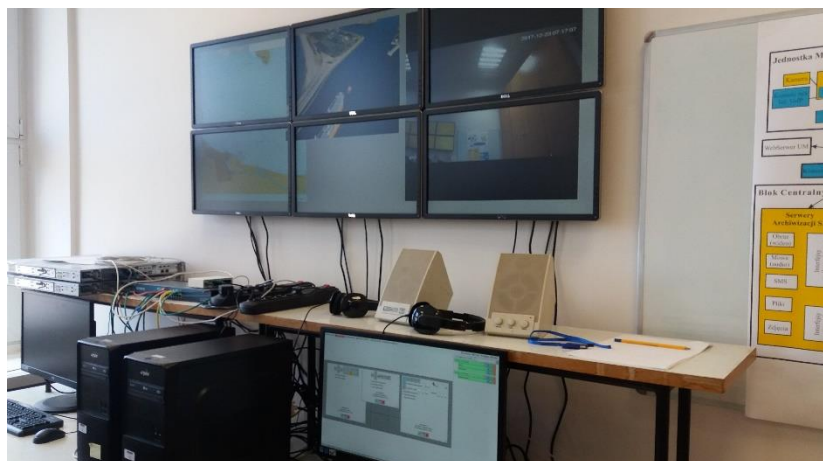


Fig. 6. Placement of the multimedia data on the Management Screen and corresponding visualization on the Multidisplay

All of the tests, including presented here example, have been conducted and produced positive results. No error occurred during the tests, leading to positive feedback from the gestor. The gestor acknowledges the results and confirms that the project fulfills all expectations. Gestor's feedback and commentary during the acceptance tests lead to conclusions regarding further development of the project. These have been incorporated into the newest version of the software. Since functionality tests give only information whether the functionality under test works as expected and whether the user experience is satisfactory, additional measurements of internal wait and processing times have been performed. Such measurements offer the possibility to predict the system performance under the increased data and requests loads in real environment for system deployment with increased number of MUs, OPs and stationary consoles. The measurement and results analysis are described in the following section.



MAPSERVER PROCESSING AND WAIT TIMES MEASUREMENTS

The mobile and stationary consoles as well as the EVP controls access data through the MapServer which also gathers all the map data. This means that its performance is crucial to the overall system response times. In previous work STRADAR system elements models have been presented and analyzed [3] which helped in identification of potential system weak points. At current system development stage it was now possible to measure the processing and wait times of messages in MapServer modules [1, 4] which have been used to verify system performance and predict potential bottlenecks. Based on this analysis propositions for the system improvements have been suggested. Additionally, measured data allow for validation of models presented in [3] and identification of their weak points and can help in development of more accurate models.

In order to perform the measurements the following times have been registered and logged: time when the received message is added to the modules queue, time when the message is removed from the queue and passed to the processing, and finally the time when the received message has been processed and the response message is ready to be sent. After the testing session the logs have been analyzed and apart from the message wait (T_{wait}) and processing times (T_{proc}), the queue length changes have been calculated based on the assumption that tests start with empty system queues (fig. 7).

The measurement session presented in fig. 7 consists of two parts: (1) from 13:20 till 14:05 — without additional map controls and (2) from 14:05 till 14:45 with four additional map controls working in the background. In both cases the central MapServer have been provided with additional map data by the MU's URC. During both test sessions the EVP operator performed task similar to the task done during functionality tests. The measurements obtained during tests are presented also in tab. 1 where for each MapServer queue number of handled messages N , messages intensity λ , mean $E(\cdot)$, variance $V(\cdot)$ and maximum values of wait and processing times, as well as, processing module load ρ , are given.

As it can be seen in fig. 7 queues of EQPM (Events Queries Processing Module) and DSM (Data Synchronization Module) never contained more than one message. In case of the EQPM, which processes tasks for the EVP created by the operators, in the test scenario only one operator was creating tasks, one at a time, and the processing of tasks is relatively simple which results in small processing times (tab. 1). On the other hand, the DSM is fed with map position data automatically within cyclic



database synchronization session, occurring every minute. In this case the processing times are larger (tab. 1), though, they do not exceed several seconds. This means that with just one URC providing data to DSM, for each message processing is finished before the next synchronization session begins.

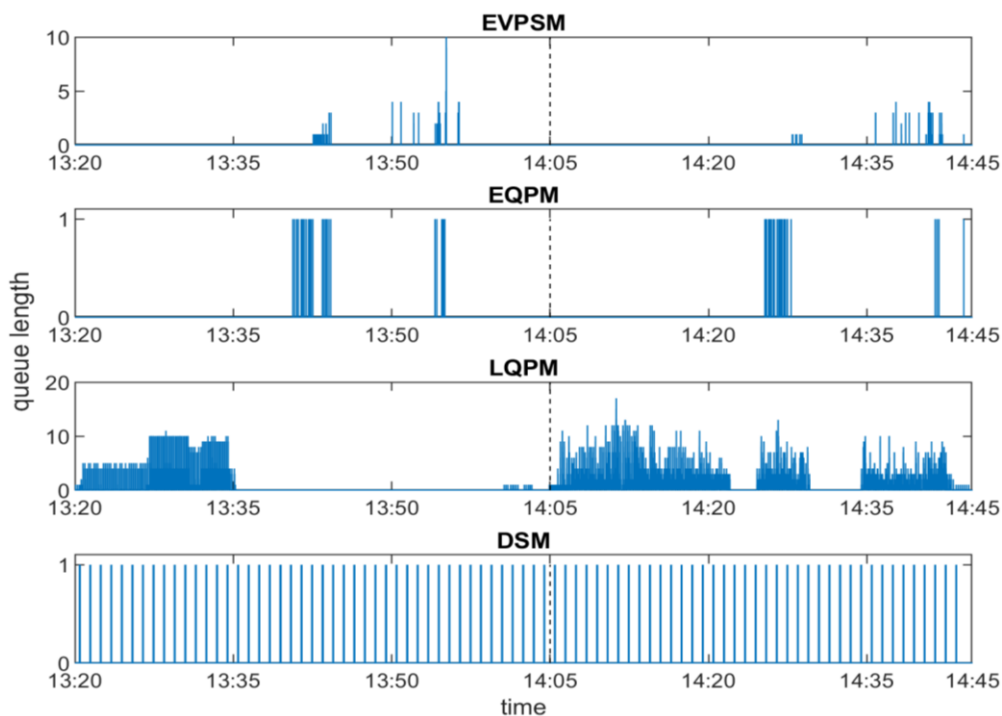


Fig. 7. Changes of lengths of the selected MapServer queues in time

Tab. 1. Measured message handling parameters in the main MapServer modules

module	N	$\lambda[1/s]$	$E(T_{wait})[s]$	$V(T_{wait})[s^2]$	$\max(T_{wait})[s]$	$E(T_{proc})[s]$	$V(T_{proc})[s^2]$	$\max(T_{proc})[s]$	ρ
EVPSM	232	0.045	8.5E-02	6.8E-02	1.58	0.0776	1.3E-01	2.7341	0.0035
EQPM	60	0.012	5.4E-05	7.8E-11	6.9E-05	0.0129	5.6E-04	0.1327	0.0002
LQPM	3325	0.652	1.03	0.98	4.83	0.2709	2.8E-01	2.9714	0.1766
DSM	168	0.033	4.3E-05	1.4E-10	7.4E-05	0.9235	8.4E-01	3.2451	0.0304

Different situation can be seen for EVPSM (EVP Support Module) and LQPM (Local Queries Processing Module) for which maximum input queue lengths are 10 and 17, respectively (fig. 7). In both cases, appropriate visualization controls typically send a group of messages related to single operator's action. In the result the subsequent

messages usually enter the queue before a previous message has been processed. Additionally, since some of the messages, such as video data requests for EVPSM or map object tracks requests for LQPM, are characterized with processing times up to several seconds, they can affect observed wait times if such messages directly precede other messages. This is especially noticeable for LQPM module, where mean processing time is around a quarter of a second with low variance and the average module load is below 20% but the mean wait time is equal to one second with high variance and maximum wait time reaching 5 seconds (tab. 1). The more detailed analysis of the measurement data showed that this is a result of the CurrentDetailsRequest (CDQ) and ArchivalDetailsRequest (ADQ) typically following directly the CurrentTrackRequest (CTQ) and ArchivalTrackRequest (ATQ). Processing of the first two messages is simple and takes very little processing time, while processing of track data requests requires time-consuming database searches and the first two request need to wait in queue until this processing is finished even though by 80% of time the module was just waiting for new messages.

During the test the most loaded module was LQPM with over three thousands of messages being processed. This module was also mainly affected by the change in the testing setup after 14:05 when additional map controls have been included into the test scenario. This was expected, since map controls' requests are processed by the LQPM. The most evident differences between these two test parts are increased load (about 8% before 14:05 and 29% after 14:05) and significantly larger variance of the queue length after 14:05 (fig. 7). It is worth noticing that the mean wait time was larger in the first test part (about 1.5 sec) than in the second test part (about 0.9 sec) while the mean processing time was similar in both test parts (about 0.27 sec). This module behavior can be blamed on aforementioned grouping of messages sent by map controls, which with several independent map controls active during second test part was disrupted. Messages from different map controls were mixed in the module's input queue.

Apart from DSM, which processes map data received from MUs, the central MapServer collects and processes map data from its own sources (WebService) within NDPM (New Data Processing Module). This module processes a large volume of mapdata packets (tab. 2) but the mean processing time is small (around 0.01 sec) with small variance. Nonetheless, the maximum observed processing time exceeds 2 seconds. Such long processing times happen incidentally when DSM and NDPM attempt at the same time to update the data in the database about the same map object. This does not affect significantly the NDPM load but with expected larger map data volume load in the target environment, improvement of such cases handling



might be necessary. Additionally, ADM (Anomalies Detection Module) analyses new map data in search of abnormal changes in main information about map objects. Since only main object's data is analyzed, the number of processed in this module packets is significantly lower than in the case of NDPM (tab. 2). Moreover, the processing time is very small, which lead to marginal module load.

Tab. 2. Measured map data handling parameters

module	N	$\lambda[1/s]$	$E(T_{proc})[s]$	$V(T_{proc})[s^2]$	$\max(T_{proc})[s]$	ρ
NDPM	36058	7.070	0.0125	5.5E-04	2.183817	0.0883
ADM	3067	0.601	0.0004	1.1E-06	0.032662	0.0003

Analysis of measurement data allowed for observation of queues' length changes and determination whether the messages processing is done with satisfying efficiency. On the basis of the obtained results we have verified that during the tests the processing in the MapServer was executed efficiently enough and in most cases with large reserves of processing powers. This means that in the target environment system overload should not be observed. Nonetheless, based on the measurement data detailed analysis the messages with long processing times have been identified which helped us to determine the areas where potential bottlenecks may appear when the system is used with increased number of OPs and MUs. These are map data deduplication and processing, including DSM, and processing of CTQ and ATQ messages in LQPM which negatively influences waiting times of other messages.

The gathered measurement data also allowed to verify the assumptions used for creation of models presented in [3]. Especially, the measured processing times and messages intensities can be used to verify assumed values of these parameters. Nonetheless, several weaknesses of proposed models have also been identified, which are mainly related to unevenness of intensities of messages generated by the EVP operator and, what is more important, grouping of messages in series with several different classes of messages with distinctly different processing times received by the LQPM, which should be considered in future models.

Apart of determination of potential weak points of the system under development, the tests and the aforementioned analysis allowed us to create propositions for system improvements. In case of map data preprocessing and deduplication, which is at risk of overloading when number of data sources will increase we propose splitting processing of messages received from DSM before sending them to the deduplication module into smaller chunks so the database transaction will be smaller with decreased processing times limiting risks of blocking other modules trying to access

the same data. Additionally, an option for moving preprocessing, data synchronization and deduplication processes out of the MapServer and creating external Deduplication Server managing its own central map data database copy which will send to the central MapServer only deduplicated map data.

Regarding long processing times of CTQ i ATQ the solution is forcing separate track requests for each map object. At the moment one track request message can ask for data for several objects, which blocks processing of other messages until track data for all the objects are retrieved from the database. Secondly, what is more important, processing of CTQ and ATQ messages must be modified. The small observed processing times of CDQ and ADQ messages indicate that retrieving data necessary for displaying tracks on maps can be done based on infrequent messages CDQ and ADQ or their simplified variants with additional track data managements within the Map Control.

CONCLUSIONS

The paper describes the STRADAR system developed within a research and development project for the current needs of the maritime division of the Polish Border Guard. The designed and implemented system integrates IP technology based telephony-dispatch communication system with naval objects monitoring surveillance system supplemented with multimedia archiving and provisioning system. Using the system BG's operators can maintain communication and monitor current tactical situation on maps. Additionally, they can signal important tasks, supplemented with metadata pointing at the map and multimedia data, to the EVP located in the CENTER. Based on such tasks the EVP operator can visualize and analyze several current or archival tactical situations supplemented with other data such as videos from cameras or audio of conversations exchanged between BG's operators. The system is currently at 7th level of technological readiness and recently has successfully passed the acceptance tests performed for the NCBiR and the BG. Apart from the functional tests the wait and processing times of messages in the central MapServer have been measured which helped us to identify potential bottlenecks and propose targets for further system improvements.

Acknowledgements

This work has been co-financed by NCBiR (National Center for Research and Development), project DOB-BIO6/10/62/2014.



REFERENCES

- [1] Blok M., Kaczmarek S., Młynarczyk M., Narloch M., *Mapserver — information flow management software for the border guard distributed data exchange system*, 'Polish Maritime Research', 2016, Vol. 23, No. 3(91), pp. 13–19.
- [2] Blok M., Czaplewski B., Kaczmarek S., Młynarczyk M., Narloch M., Sac M., *Multimedia distributed system for visualization of ongoing and archival events for BG*, The International Tech-Science Conference on 'Naval Technologies for Defence and Security', NATCon 2016, pp. 61–76.
- [3] Blok M., Kaczmarek S., Narloch M., Sac M., *The central server of the Border Guard's distributed multimedia system for monitoring and visualisation of ongoing and archival events*, 'Journal of Marine Engineering and Technology', 2017, Vol. 16, Issue 4, pp. 257–272.
- [4] Blok M., Czaplewski B., Kaczmarek S., Litka J., Narloch M., Sac M., *The System of the Supervision and the Visualization of Multimedia Data for BG*, 12th International Conference on Marine Navigation and Safety of Sea Transportation "TransNav 2017", Gdynia 21–23 June 2017, ed. A. Weintrit, CRC Press/Balkema, pp. 227–236.
- [5] Cetin F. T., Yilmaz B., Kabak Y., Lee J. H., Erbas C., Akagunduz E., Lee S. J., *Increasing Maritime Situational Awareness with Interoperating Distributed Information Sources*, 18th International Command & Control Research & Technology Symposium, ICCRTS 2013, pp. 9–22.
- [6] Fiorini M., Maciejewski S., *Lesson Learned During the Realization of the Automated Radar Control System for Polish Sea-waters (ZSRN)*, 'Marine Navigation and Safety of Sea Transportation: Advances in Marine Navigation', 2013, CRC Press, pp. 217–221.
- [7] Gajewski S., *Design of OFDM-based radio communication systems for coast-to-sea and coast-to-air propagation environments*, 'Polish Maritime Research', 2016, Vol. 23, No. 1(89), pp. 12–19.
- [8] Gałęziowski A., *Automatic National System of Radar Control for Maritime Areas of Poland* [in Polish], 'Przegląd Morski', 2005, No. 5, pp. 50–70.
- [9] Hassene R., Mondher M., *Advanced Technologies for Marine Border Security*, 1st International Symposium on Land and Maritime Border Security and Safety, Dhahran 15–19 October 2017, pp. 112–121.
- [10] Kosz P., Rajchowski P., Cwalina K., Sadowski J., Stefański J., *Research and analysis of high-speed data transmission radio link designed for maritime environment*, 'Telecommunication Review — Telecommunication News', 2017, No.8–9, pp. 891–894.
- [11] Mickiewicz P., *Maritime safety and development programs of the Polish state until 2030* [in Polish], 'Rocznik Bezpieczeństwa Międzynarodowego', 2014, No. 8(2), pp. 77–95.
- [12] Napolitano J., *Northern Border Strategy*, Department of Homeland Security, 2012.
- [13] Paleri P., *Coast Guards and Maritime Partnerships: An Over-the-Horizon Perspective*, 'Journal of the Maritime Affairs', 2015, Vol. 11, Issue 2, pp. 99–111.
- [14] Pleszkun P., *The Baltic Sea under surveillance* [in Polish], 'Przegląd Morski', 2011, No. 5, 26–34.
- [15] Tazzioli M., Eurosur, *Humanitarian Visibility and (nearly) Real-Time Mapping in the Mediterranean*, 'ACME: An International Journal for Critical Geographies', 2016, Vol. 15, No. 3, pp. 561–579.
- [16] Tikanmäki I., Ruoslahti H., *Increasing Cooperation between the European Maritime Domain Authorities*, 'International Journal of Environmental Science', 2017, Vol. 2, pp. 392–399.
- [17] Valera M., Velastin S. A., *Intelligent distributed surveillance systems: a review*, 'IEE Proceedings — Vision, Image and Signal Processing', 2005, Vol. 152, Issue 2, pp. 192–204.



STRADAR — MULTIMEDIALNY SYSTEM DYSPOZYTORSKO-TELEINFORMATYCZNY DLA STRAŻY GRANICZNEJ

STRESZCZENIE

Zapewnienie bezpieczeństwa granic państwowych wymaga stosowania multimedialnych systemów nadzoru automatycznie gromadzących, przetwarzających oraz udostępniających różnego rodzaju dane. W artykule przedstawiono tego typu system opracowany w ramach projektu STRADAR na potrzeby Morskiego Oddziału Polskiej Straży Granicznej. System ten oprócz zapewnienia łączności dyspozytorskiej gromadzi: dane mapowe z odbiorników AIS i GPS oraz radarów, wideo i zdjęcia z kamer, dźwięk z połączeń telefonicznych i wiadomości SMS pochodzące z wielu jednostek mobilnych i stacjonarnych punktów obserwacyjnych rozlokowanych wzdłuż linii brzegowej. Wszystkie te dane mogą być wizualizowane na wieloekranie, gdzie operator może je dynamicznie wybierać i konfigurować formę ich prezentacji. W artykule opisano funkcjonalność systemu, jego implementację sprzętową i programową oraz przedstawiono wyniki testów akceptacyjnych, które system niedawno przeszedł.

Słowa kluczowe:

system nadzoru, monitoring morskiej granicy, system dyspozytorski, multimedia, wizualizacja danych.

Article history

Received: 28.05.2018

Reviewed: 21.09.2018

Revised: 04.02.2019

Accepted: 07.02.2019

