We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

5,300
Open access books available

130,000
International authors and editors

155M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

ICARUS Training and Support System

Janusz Będkowski, Karol Majek, Michal Pełka,
Andrzej Masłowski, Antonio Coelho,
Ricardo Goncalves, Ricardo Baptista and
Jose Manuel Sanchez

Additional information is available at the end of the chapter

Abstract

The ICARUS unmanned tools act as gatherers, which acquire enormous amount of information. The management of all these data requires the careful consideration of an intelligent support system. This chapter discusses the High-Performance Computing (HPC) support tools, which were developed for rapid 3D data extraction, combination, fusion, segmentation, classification and rendering. These support tools were seamlessly connected to a training framework. Indeed, training is a key in the world of search and rescue. Search and rescue workers will never use tools on the field for which they have not been extensively trained beforehand. For this reason, a comprehensive serious gaming training framework was developed, supporting all ICARUS unmanned vehicles in realistic 3D-simulated (based on inputs from the support system) and real environments.

Keywords: training systems, support systems, real-time 3D reconstruction

1. Introduction

The ICARUS Training and Support system provides the command and control component that integrates different data sources of spatial information, such as maps of the affected area, satellite images and sensor data coming from GIS database and the unmanned robots in order to provide a situation snapshot to the rescue team that makes the necessary decisions. The system is implemented based on the concept of High Preformance Computing (HPC) in the Cloud . The integration and visualization of maps derived from UxVs is the main functionality of the proposed HPC solution. These maps are available in the Cloud; therefore, the information concerning disaster can be distributed over Ethernet within the connectivity constraints.

The second important functionality is the ICARUS serious games in the Cloud which are used for end-user training purposes. The proposed HPC solution is capable of streaming serious games over Ethernet; therefore, ICARUS international teams can train simultaneously from offices in different countries. The third functionality is the integration-evaluation cycle using HPC. ICARUS partners had access to the server installed in the Data Centre, which means that the integration between the different teams could be performed using this tool during project development and evaluation phase.

2. State of the art

The main and most important task of rescue services during a major crisis is to search for human survivors on the incident site. As such an endeavour is complex and dangerous, it often leads to loss of lives among the rescuers themselves. It is evident that the Unmanned Search And Rescue devices can improve search and rescue process. Many research efforts towards the development of unmanned SAR tools have been made [1]. One of these efforts is Neptus, a C3I (Command, Control, Communication and Information) framework, which aims to support coordinated operation of heterogeneous teams, including several types of UVs and human beings [2]. Another example is the German project I-LOV, which establishes a framework for integration of mobile platforms into a victim search mission [3]. Numerous attempts to use robotic systems in crisis situations were made: the 2001 World Trade Center attack [4], the 2004 earthquake in Mid Niigata, the 2005 USA hurricanes [5] or the 2011 Japan tsunami. Papers [3] and [6] give a broad overview of the effort done in this area. This research effort stands in contrast to the practical reality in the field, where unmanned SAR tools have great difficulty finding their way to the end users, due to a number of remaining bottlenecks in the practical applicability of unmanned tools [7].

The Training and Support system concerns the Serious Games (SGs) that are becoming increasingly popular in the corporate and research communities. However, there are still different definitions of what a serious game is. In this chapter, SGs are defined as follows: serious game applications which take advantage of all the features that make games fun and engaging and use them to empower training [12], promoting the trainees' interest by making the educational subject more exciting. Training is defined as "an organized activity aimed at imparting information and/or institutions to improve the recipient's performance or to help him or her attain a required level of knowledge or skill" [13]. Using games and structured learning activities in training is an excellent way to bring key topic areas to the learner [14]. On the SG taxonomy defined by Sawyer and Smith [15], games for training fall in areas like government, defence, education and industry. They cover different aspects such as occupational safety, skills, communications and orientation (e.g. Ref. [16]).

Currently, several tools exist to simulate and visualize unmanned vehicles operating since this type of software allows overall cost reduction and provides a platform for safer and faster testing. The relevant example is the USARSim [17], an open source framework built on top of

Unreal Engine [18] to simulate multirobots and environments. In addition, interfacing with the Mobility Open Architecture Simulation and Tools framework (MOAST [19]) provides a modular control system of robot control and customization. It allows the system users to add their own modules or alter the existing ones in order to obtain more complex robots than USARSim is capable of implementing. Another example of such a tool is Webots [20], which also allows some degree of robot customization at the shape and attribute level. This tool also has the advantage of allowing robots to be independent from the tool and communicate with it remotely through TCP/IP. To handle physics simulation, Webots uses Open Dynamic Engine (ODE [21]). Recent most popular simulation tool for multirobot systems is Gazebo integrated with ROS (Robot Operating System) [22].

The ICARUS (Integrated Components for Assisted Rescue and Unmanned Search Operations) project is a large European research project with 24 partners from 10 European countries. It concentrates on the development of unmanned surface vehicles (USV), unmanned air vehicles (UAV) and unmanned ground vehicles (UGV) and search and rescue technologies for detecting, locating and rescuing humans. All these systems need to share information between them, and interoperability is an important issue to take in account [8, 9].

3. Hardware

The proposed HPC solution for the ICARUS project is based on the Server Supermicro RTG-RZ-1240I-NVK2 shown in **Figure 1**. This server provides the NVIDIA GRID Technology. NVIDIA GRID technology includes NVIDIA CUDA architecture, which means that the server provides a parallel programming solution for processing large data sets. These advantages of the server make it relevant for ICARUS needs, as data from many sources (UxV) can be efficiently integrated and visualized over Ethernet.

3.1. Key features

1. Dual socket R (LGA 2011) supports Intel® Xeon® processor E5-2600 v2 family

2. Up to 512GB ECC DDR3, up to 1866MHz; 8x DIMM sockets

3. 3x PCI-E 3.0 x16 slots (supports GPU cards), 1x PCI-E 3.0 x8 (in x16) low-profile slot

4. Integrated IPMI 2.0 with KVM and Dedicated LAN

5. Intel® X540 10GBase-T Controller

6. 4x Hot-swap 2.5″ SATA3 Drive Bays

7. 1800W Redundant Power Supplies Platinum Level (94%+)

8. 10x Counter rotating fans w/optimal fan speed control
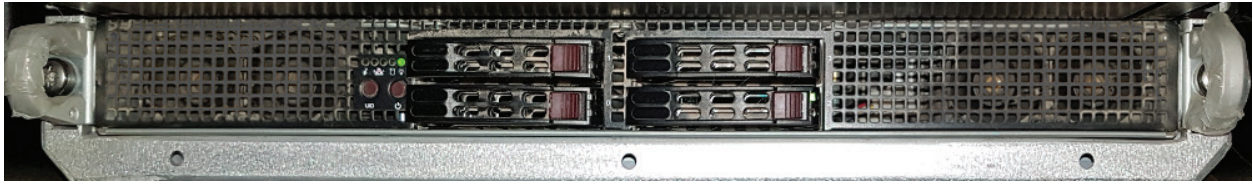
9. Smart server management tools.

Figure 1. The HPC solution for the ICARUS project is based on the Server Supermicro RTG-RZ-1240I-NVK2 (source: ICARUS).

3.2. Server's specification

- 2x Intel 2.8GHz Ten-Core Xeon CPU's (Max Turbo 3.6GHz)

- 256GB (16x16GB) Supermicro DDR3 Registered ECC Memory

- 2x 240GB Enterprise Class SATA 2.5″ SSD's extended by 1TB SSD Samsung Hard Drive

- 2xNVIDIA GRID K2 cards (4GPUs in Total).

3.3. Ruggedized chassis

Meant to be deployed in tough environmental conditions, the server was embedded into a ruggedized chassis as shown in **Figure 2**. The chassis can easily be carried by two people. It is even possible for a single person to carry the chassis. It protects the server from vibrations and mechanical stress. **Figure 3** shows the final fully equipped HPC solution for the ICARUS
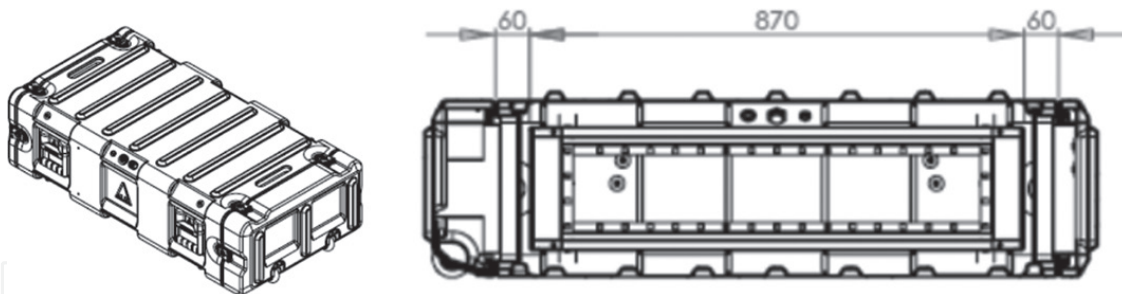

Figure 2. Chassis diagram and measurements (source: ICARUS).

Figure 3. Fully equipped HPC solution for the ICARUS project (source: ICARUS).

project. It is extended by a mobile display-keyboard-mouse component for on-site server management purpose. It is a fully integrated and autonomous solution. It requires 2kV AC power. The communication system is based on a WiFi router that can establish a local network within a range of 25 m. If Ethernet access is available, the server is connected directly to the network.

4. Software infrastructure

The Training and Support system in the Cloud is designed based on two models: VDI (Virtual Desktop Infrastructure, **Figure 4**) and SaaS (Software as a Service, **Figure 5**). These models support vGPU (virtualization of Graphic Processing Unit) technology provided by NVIDIA GRID processors. GPU virtualization provides robust rendering over Ethernet
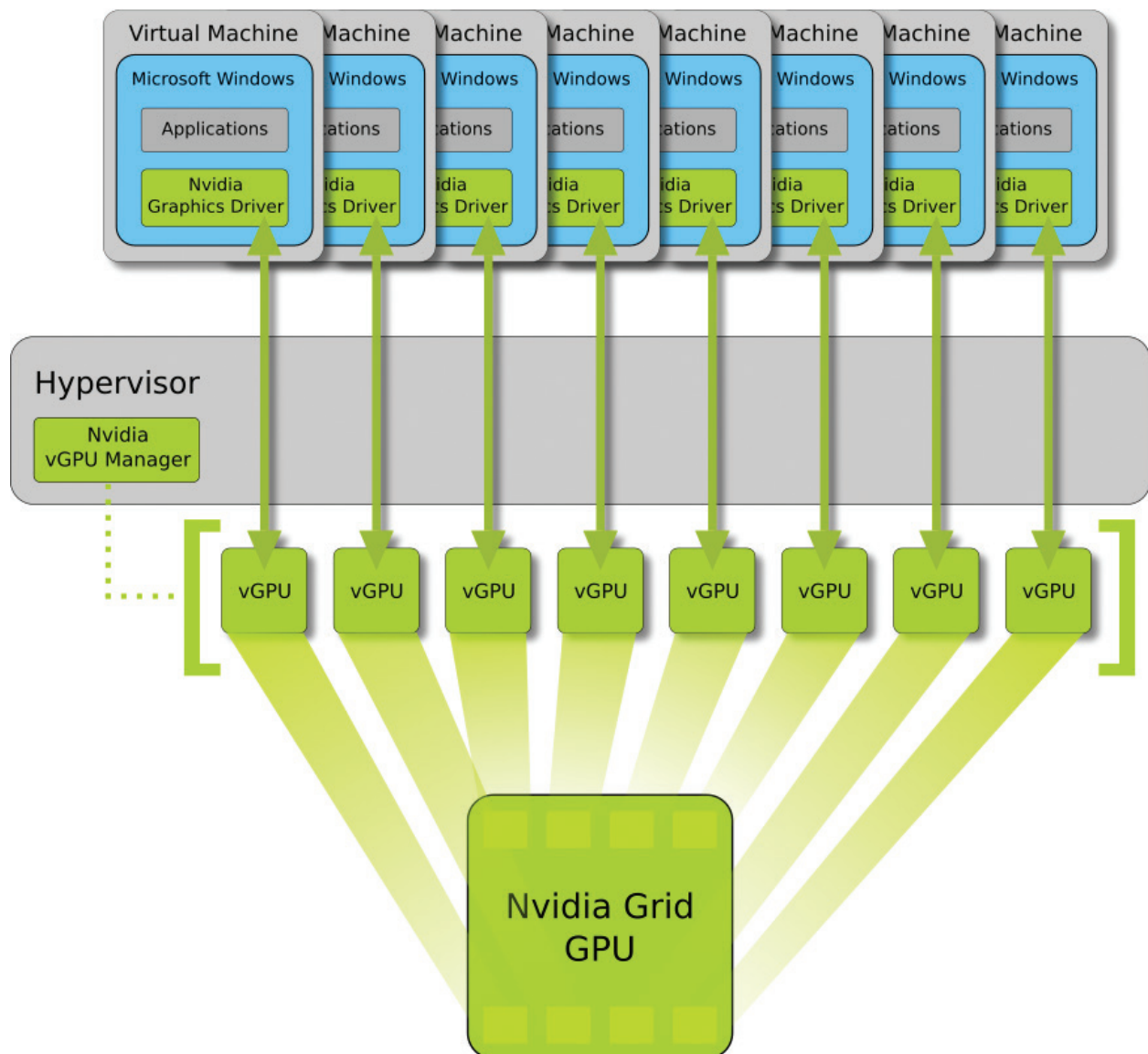

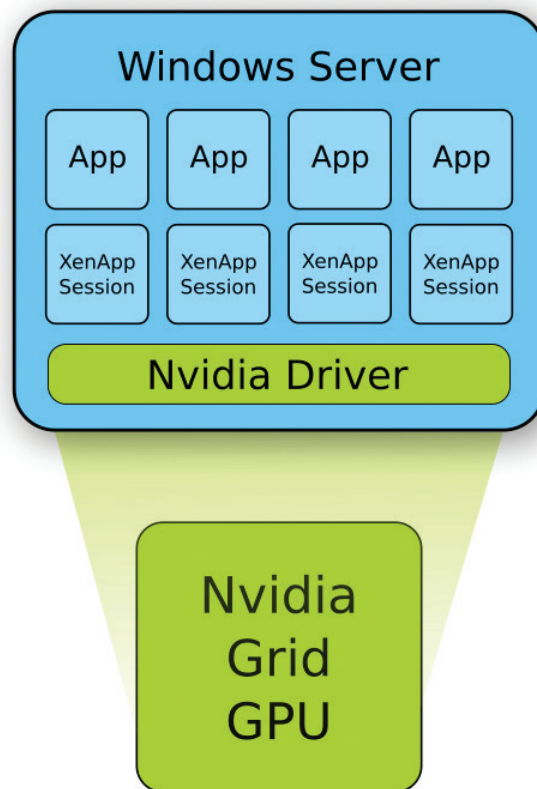**Figure 4.** Virtual desktop infrastructure (source: ICARUS).

Figure 5. Software as a Service architecture (source: ICARUS).

functionality and supports parallel computation with the CUDA framework. These func-tionalities are very promising technologies for mobile robotics applications where 3D maps derived from many sensors (3D lasers, photogrammetric cameras) have to be inte-grated. A very important aspect is that due to bandwidth limitations, the data transfer from robots to end users must be limited. To reduce the data flow, the rendering of such data is performed on the server, and only images (from the 3D rendering) are streamed over the network. This approach efficiently reduces the needed bandwidth for interaction with maps.

4.1. Virtual desktop infrastructure

The VDI technology allows operating system (in this case Win7) virtualization with the vir-tual hardware assignment per user. In this scheme, it is possible to allocate proper hardware resources needed for certain users. VDI technology allows vGPU sharing with many users simultaneously. Thus, users can be equipped with a full GPU (GPU Pass-Through mode), ½ GPU (2 users per GPU), ¼ GPU (4 users per GPU) or 1/8 GPU (8 users per GPU). It can be noticed that most demanding VDIs (simulation of UGV and simulation of USV) have been assigned a pass-through mode of GPU, allowing for the highest simulation performance. Less-demanding VDIs have been assigned only 1/8 GPU. It is possible to dynamically adjust the GPU placement policy to different needs and applications. The advantage of VDI is that

once assigned, virtual hardware belongs only to the user and other users do not affect it. This functionality guarantees a flexible and stable simulation environment for demanding training tools.

4.2. Integration of training and support with ICARUS system

Figure 6 shows data flow from/to the Remote Command and Control System (RC2, see chapter 8 of this book and [10]) to/from the Training and Support System (in training mode). The goal was to provide an interface to the RC2 via control interfaces. The final communication emulation module translates the ICARUS interface into the internal Training and Support communication scheme. There are two modes: training mode and support mode. During the training mode simulation, training and support tools are integrated with the ICARUS system to provide the training capabilities based on the real or the virtual components of the ICARUS system. **Figure 7** shows the support mode where the support tools are integrated with the ICARUS system. In this mode, the operator has access to additional information provided by the Command and Control Component via the Human Machine Interface.
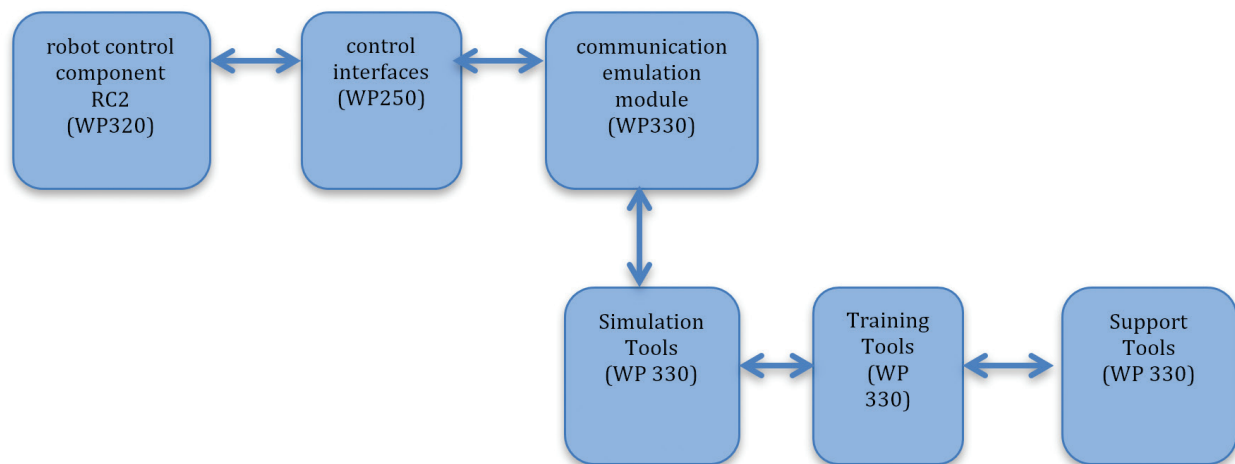
Figure 6. Data flow from/to RC2 to/from the training and support system (in training mode) (source: ICARUS).
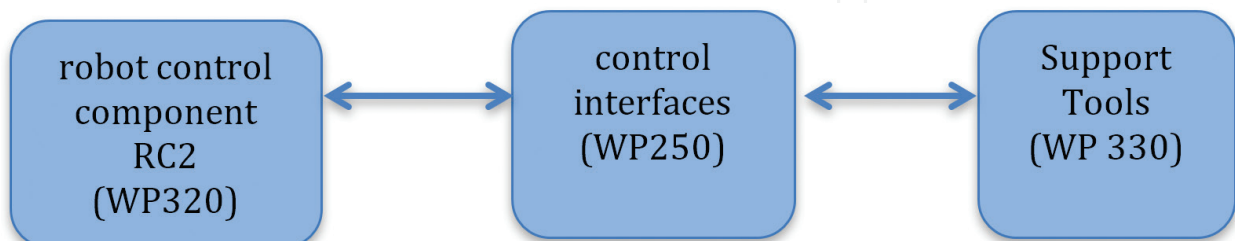
Figure 7. Data flow from/to RC2 to/from the training and support system (in support mode) (source: ICARUS).

5. Communication emulation module

The communication emulation module (**Figure 8**) is responsible for the simulation of the properties of an existing, planned and/or nonideal network with a certain propagation model. It is possible to simulate several nodes with a realistic radio propagation model among themselves that could be chosen depending on the situation of the network. The communication module mainly emulates the common attributes in a typical network, such as the packet losses and the delay on the established connection between the Unmanned Vehicle and the Operator (RC2).

To determinate the network behaviour, this module receives telemetry data from Unmanned Vehicles, containing information regarding their position and expected traffic load, supposing that this module controls the rest of the nodes in the network. That is why this module is composed mainly of two components:

- Network Emulator: The aim is to emulate the network conditions such as topology, congestion, load balance, node failure and so on that could vary depending on, for example, the position of the UV, new obstacles or environmental conditions.

- RF Emulator: The aim is to emulate the radio propagation models to predict the received signal in the network nodes. The wireless technologies which will be used between UV and RC2 are 802.11 and DMR.

The telemetry data (position, speed, yaw, pitch, roll, etc.) from the Unmanned Vehicle Simulator are received at the RF emulator in real time, that is without delay. From this information,
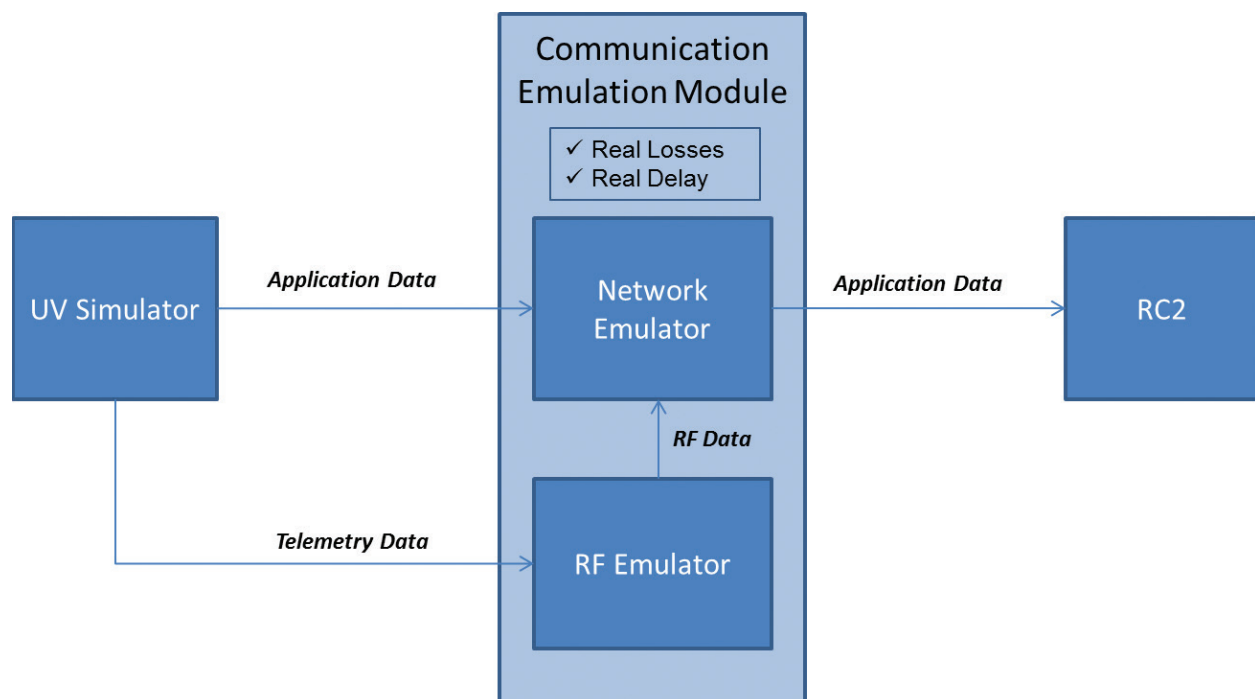

Figure 8. Communication Emulation Module (source: ICARUS).

the RF Emulator applies the appropriate propagation model and calculates the link budgets among possible network links. We suppose that the only node with movement would be the Unmanned Vehicle, although the rest of the nodes could also cause changes in the link budgets.

6. Network emulator

The Network Emulator module is based on UML (User-Mode-Linux), for instance, nodes in the network. UML allows starting a Linux Machine as a user process running within the host machine. From the point of view of the host machine, UML is a normal user process. From the point of view of a process that runs within UML, UML is a kernel, offering virtual memory and accessing to devices, what we call virtual machine. The UML architecture is shown in **Figure 9**.

The UML Kernel does not communicate directly with the hardware; it does it through Linux Kernel of the host machine. The processes that run within UML work the same way that within the Linux real machine, so that UML offers its own addressing space of kernel and process, its system of management of memory and planning.

The file system that UML uses to start each virtual machine is stored in a single file. This file is where the kernel of Linux and the configuration of a virtual machine are found. When we want to start a virtual machine, UML starts it basing on the kernel installed in that system file.

Thanks to the UML, we can execute a group of virtual machines within a real machine, the host machine. The virtual machines are connected to virtual collision domains. A virtual machine can work like a terminal machine, a router or a switch. Network Emulator module is developed using UML technology, thus a group of commands allows configuring and connecting virtual machines and the file system.

The devices may incorporate a varying amount of standard network attributes like: the round-trip time across the network (latency), the amount of available bandwidth, a given degree
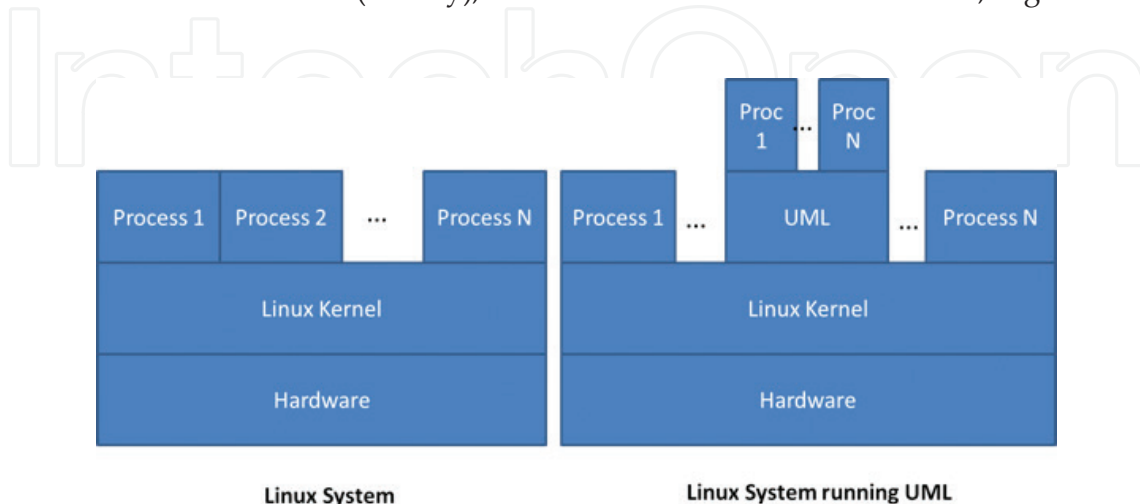

Figure 9. UML architecture (source: ICARUS).

of packet loss, duplication of packets, reordering of packets, corruption and modification of packets and/or the severity of network jitter. It can also mimic typical Layer 1 physical errors, such as Bit Error Rate, Loss of Signal, Output Bit Rotation and others.

7. Validation of the 3D-modelling capabilities of the support system

7.1. Validation in a marine incident scenario

The main feature of the support system is the ability to quickly merge raw 3D scans into complete 3D models of the environment. The 6DSLAM algorithm used for this task was tested on several different occasions in various environments. A major test of mapping capabilities was performed during the Icarus Sea Demo in Lisbon, Portugal. During the trial, a 3D model of the area where the trial took place was made by a robot shown in **Figure 10**—a husky robot with rotating SICK 500 LMS and a LadyBug 3 spherical camera. The final model is shown in **Figure 11** . The created model consists of over 90 million points.

7.2. Validation during euRathlon 2015 multirobot multidomain competition

The support system was also used during the euRathlon 2015 competition [11]. The 3D mapping system was mounted on one of the Icarus team's robots: Teodor. During robot operation the system was gathering 3D data about the mission area. It was also the key element of semiautonomous operation of the robot, as 3D data was used for planning the motion of the robot.

Figure 10. Dedicated support system—mobile mapping platform (source: ICARUS).
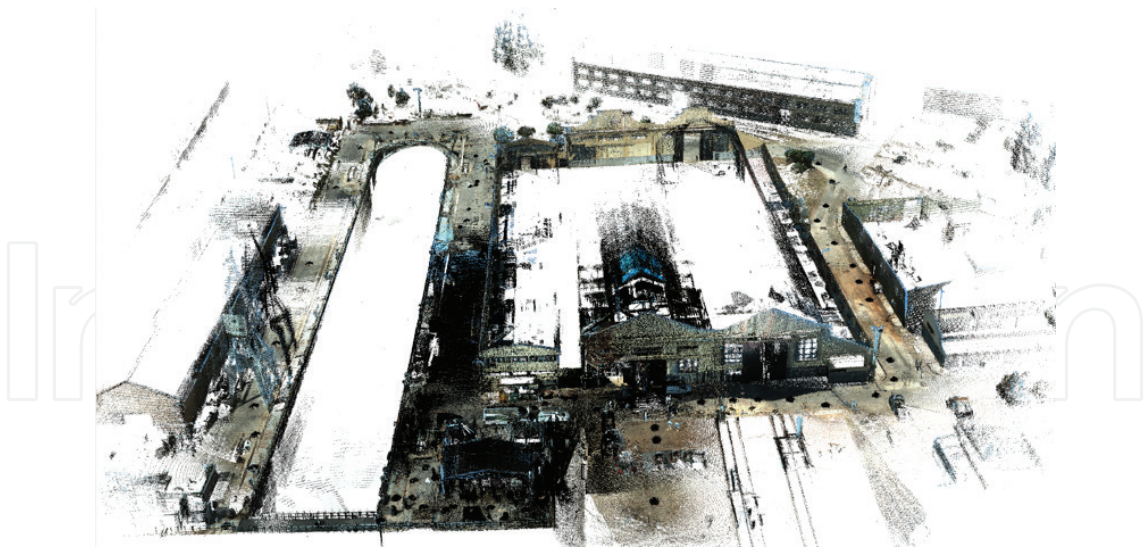
Figure 11. Initial environment model from Sea Trials (source: ICARUS).

The three main areas in which the support system was used during the operation was creating 3D maps of the environment, finding objects of interest and allowing semiautonomous operation of the robots.

The 3D mapping capabilities of the support system were used to create an outdoor map of the area. The map was coloured based on data from the ladybug. For the final scenario, the Grand Challenge, the created land map was merged with a 3D map obtained from an unmanned aerial vehicle to create a multilayer complex map of the area (**Figure 12**). The result was highly praised by the judges.

Using a 360° camera allowed to find a number of objects of interest undetected by the operator and the automatic algorithms connected to classical robot camera (**Figure 13**). Apart from

Figure 12. Multilayer map of euRathlon Grand Challenge area; top: land map layer, bottom: aerial map layer (source: ICARUS).

Figure 13. Ladybug spherical image with a number of OPIs (source: ICARUS).

a wider field of view, tools available in the support system allowed to enhance the gathered images in post-processing. After this process, some objects of interest that were previously not visible in raw data (and as such were not detected by either the operator or the automatic algorithm) became visible (**Figure 14**).

Figure 14. Enhanced LadyBug image. The marker inside the house is visible (source: ICARUS).

8. Conclusion

In this chapter, the ICARUS Training and Support system is discussed, introducing the High-Performance Computing (HPC) in the Cloud concept for improving the command and control system. The integration and visualization of maps derived from the different unmanned vehicles is the main functionality of this system. These maps are made available in the Cloud by the presented system, such that all the information concerning the disaster can be distributed over Ethernet, while respecting bandwidth limitations. A second important functionality is the serious games in the Cloud which are used for training end users. The proposed HPC solution is capable of streaming serious games over Ethernet, which means that using this system, international search and rescue teams can train simultaneously from offices in different countries.

Acknowledgements

Author details

Janusz Będkowski[1]*, Karol Majek[1], Michal Pełka[1], Andrzej Masłowski[1], Antonio Coelho[2], Ricardo Goncalves[2], Ricardo Baptista[2] and Jose Manuel Sanchez[3]

*Address all correspondence to: januszbedkowski@gmail.com

1 Institute of Mathematical Machines, Warsaw, Poland

2 DEI, Faculdade de Engenharia, Universidade do Porto/INESC TEC, Porto, Portugal

3 IntegraSys S.A., Las Rozas, Spain

References

[1] Kruijff GM, Colas F, Svoboda T, van Diggelen J, Balmer P, Pirri F, Worst R. Designing intelligent robots for human-robot teaming in urban search and rescue. In: AAAI Spring Symposium Series on Designing Intelligent Robots; AAAI Publications; 2012

[2] Dias PS, Gomes RMF, Pinto J. Mission planning and specification in the Neptus framework. In: IEEE International Conference on Robotics and Automation (ICRA); 15-19 May; IEEE; 2006. pp. 3220-3225. DOI: 10.1109/ROBOT.2006.1642192

[3] Hamp Q, Gorgis O, Labenda P, Neumann M, Predki T, Heckes L, Kleiner A, Reindl L. Study of efficiency of USAR operations with assistive technologies. Advanced Robotics. 2013;**27**(5):337-350

[4] Murphy RR. Trial by fire [rescue robots]. Robotics Automation Magazine. 2004;**11**(3):50-61

[5]   Murphy RR, Tadokoro S, Nardi D, Jacoff A, Fiorini P, Choset H, Erkmen AM. Search and rescue robotics. In: Siciliano B, Khatib O, editors. Handbook of Robotics. Springer-Verlag; 2008. pp. 1151-1173. DOI: 10.1007/978-3-540-30301-5_51

[6]   Liu Y, Nejat G. Robotic urban search and rescue: A survey from the control perspective. Journal of Intelligent & Robotic Systems. 2013;**72**(2):147-165

[7]   Doroftei D, De Cubber G, Chintanami K. Towards collaborative human and robotic rescue workers. In: Human Friendly Robotics; 2012

[8]   Marques MM, Martins A, Matos A, Cruz N, Almeida JM, Alves JC, Lobo V, Silva E. REX14—Robotic Exercises 2014—multi-robot field trials. In: MTS/IEEE OCEANS 2015; Washington, DC, USA. 2015. pp. 1-6. DOI: 10.23919/OCEANS.2015.7404497

[9]   Balta H, Bedkowski J, Govindaraj S, Majek K, Musialik P, Serrano D, Alexis K, Siegwart R, De Cubber G. Integrated data management for a fleet of search-and-rescue robots. Journal of Field Robotics. 2016;**34**(3). DOI: 10.1002/rob.21651

[10] Govindaraj S, Chintamani K, Gancet J, Letier P, Van Lierde B, Nevatia Y, De Cubber G, Serrano D, Bedkowski J, Armbrust C, Sanchez J, Coelho A, Palomares ME, Orbe I. The ICARUS Project—Command, Control and Intelligence (C2I). In: Safety, Security and Rescue Robots; October 2013; Sweden: IEEE; 2013

[11] Marques MM, Parreira R, Lobo V, Martins A, Matos A, Cruz N, Almeida JM, Alves JC, Silva E, Będkowski J, Majek K, Pełka M, Musialik P, Ferreira H, Dias A, Ferreira B, Amaral G, Figueiredo A, Almeida R, Silva F, Serrano D, Moreno G, De Cubber G, Balta H, Beglerović H, Govindaraj S, Sanchez JM, Tosa M. Use of multi-domain robots in search and rescue operations—contributions of the ICARUS team to the euRathlon 2015 challenge. In: IEEE OCEANS; April; Shanghai, China: IEEE; 2016

[12] Susi T, Johannesson M, Backlund P. Serious Games: An Overview. 2007

[13] Available from: http://www.businessdictionary.com/definition/training.html

[14] Sugar S, Whitcomb J. Simple and Effective Techniques to Engage and Motivate Learners. American Society for Training and Development (ASTD); 2006

[15] Sawyer B, Smith P. Serious game taxonomy. In: Paper presented at the Serious Game Summit 2008; San Francisco: USA; 2008

[16] Available from: https://www.americasarmy.com

[17] Carpin S, Lewis M, Wang J, Balakirky S, Scrapper C. USARSim: A robot simulator for research and education. In: IEEE International Conference on Robotics and Automation; 2007. pp. 1400-1405. DOI: 10.1109/ROBOT.2007.363180

[18] Available from: http://unity3d.com

[19] Available from: http://moast.sourceforge.net

[20] Olivier, M. / Cyberbotics Ltd - Webots TM: Professional Mobile Robot Simulation, Inernational Journal of Advanced Robotic Systems, 2004;**1**(1): ISSN 1729-8806 pp. 40-43

[21] Available from: http://www.ede.org

[22] Available from: http://wiki.ros.org/gazebo