Defence Science Journal, Vol. 64, No. 5, September 2014, pp. 445-450, DOI : 10.14429/dsj.64.5035

Designing Integration Unit to Integrate Navigational and Tactical Equipment Onboard Naval Platforms

Kishor Choudhury[*], Ashok Kumar, and Wareendra Nath

Weapons and Electronic Systems Engineering Establishment, New Delhi-110 066, India
[*]*E-mail: kishor92@gmail.com*

ABSTRACT

A typical warship consists of weapons and sensors that are of diverse origins and are generally based on different design standards and philosophies. But to enhance the operation capability of the platform, it is very important that many of the equipment work in tandem. This paper discusses the design of an integration unit that integrates various sensors and equipment onboard a naval platform. It takes a strictly modular approach and is therefore, adaptable to any size and mission requirement. The proposed solution uses commercial off-the-shelf (COTS) hardware and relevant software, to provide the required Quality of Service (QoS) data to the end equipment and systems. This solution provides an efficient and seamless integration of various sensors, weapons and equipment onboard a naval platform.

Keywords: Systems integration, weapons and sensors, data communication

1. INTRODUCTION

A naval ship has a large gamut of systems designed for navigation and weaponry. These systems aid in the safe navigation of the ship and provide critical sea power capability to the naval forces to defend against threats. In order to achieve the goal of navigation and to provide the required combat efficiency, many sensors, equipment and weapons fitted onboard are required to work in tandem[1-3]. This requires flow of information between various equipment and systems. For example, ship's house holding data (SHHD), consisting of ship's course, speed, water depth and geographical position, are required by many navigational and tactical systems. Similarly, combat management system (CMS) is an important consumer for various navigational and tactical data to provide common tactical picture, track management, threat evaluation, and decision engagement capability to the commanding officer of the platform[4]. Thus, the parameters acquired by relevant sensors are required to be made available to various consumer equipment and systems.

This paper highlights the design of an 'Integration Unit' that integrates various sensors and equipment onboard a naval platform. It takes a strictly modular approach, and is therefore adaptable to any size and mission requirement. The proposed solution uses commercial off-the-shelf (COTS) hardware and relevant software, to distribute pertinent information to end equipment and systems as per the required format.

2. BACKGROUND

Systems integration deals with the process of linking together different systems/sub-systems physically or functionally to act as a coordinated whole[5]. Typical sensors and equipment fitted onboard a naval ship is shown in Fig. 1. In the context of a naval platform the concept of systems integration amounts to integrating sensors, equipment and weapons to increase overall system performance and capability[6]. This is a challenging task considering the fact that these sensors and equipment generally include very sophisticated electronic components and computers. Thus, the tools and techniques for preparing, mixing and matching various sub-systems are also critical technologies as these are the key ingredients to achieving the desired qualities. The problem is multiplied many times as equipment fitted onboard a warship are often procured from different countries or different manufacturers that conform to dissimilar technical standards.

While deciding on the equipment fit for a new warship, the choice of particular equipment is governed by its cost implication and/ or its expected features and capabilities. Thus, the final list of equipment often contains various equipment

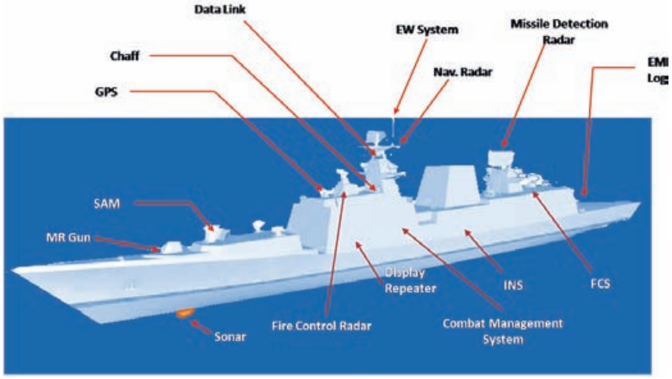


Figure 1. Typical equipment fit onboard naval ships.

Received 09 October 2013, Revised 14 March 2014, Online published 22 September 2014

that are not directly compatible with each other as far as their interfaces, data formats or protocols are concerned. Similar situation prevails during mid-life-upgrade (MLU) of ships. During MLU many equipment are replaced by their newer versions having enhanced features and functionalities, while many legacy equipment that are considered very reliable and are in perfect state of health, are retained. In this case too, it ends up with an assortment of onboard equipment that cannot interact with each other, thus hampering the need for better decision-making and improved performance.

Hence, there is a need to have an integration unit that can understand the 'languages' of the source equipment/ sensors, extract the required information and deliver it to the consumer systems in the required format. It should support all the interfaces and protocols that are generally used by the ship equipment. The integration solution should also be flexible enough to accommodate new additions of equipment with ease.

3. DESIGN CONSIDERATIONS

While deciding on the design of the integration unit, the following features were taken into consideration.

- *Scalability*: The platform used for the integration unit must support growing workloads so that data from newly inducted sensors/ equipment can be seamlessly integrated. This feature is important as it would increase the longevity of deployment of the integration unit, thus reducing the cost of the system in the long run.
- *Serviceability*: Modular system design should be employed. This would allow fast replacement and upgradation and would minimise operator time devoted to maintenance.
- *Fault Detection and Fault Isolation*: The integration unit should assist the operator to detect and isolate fault in the hardware or software. Health monitoring software modules should be part of the system and should continuously report the health of the integration Unit.
- *Availability*: The integration unit must embrace high availability features as it would be used for mission-critical applications. Single point of failures should be addressed.
- *Quality of Service (QoS)*: Processing power and I/O hardware should be chosen in such a way as to satisfy the QoS that is dictated by the end equipment and systems**.**
- *Fault Tolerance*: The integration unit must have some level of fault tolerance capability so as to enable the unit to continue operation in the event of a fault, possibly at a reduced level, rather than failing completely.

4. ARCHITECTURAL DESIGN OF THE INTEGRATION UNIT

4.1 Hardware Platform

For the design of the integration unit, COTS compact peripheral component interconnect (cPCI)-based platform was chosen to house the processors and I/O cards. The platform consists of two 8-slot sub-systems, or domains. Each domain has one slot for the host processor, one slot for the hot swap controller (HSC) board and six slots for non-host compact PCI boards. The hardware platform is fully compliant with the compact PCI hot swap specification developed by the PCI Industrial Computers Manufacturing Group. The core of this architecture is the hot swap controller and bridge module that allows the system processors to access both I/O domains. Redundant power supply units were used for increasing the availability of the system. Figure 2 illustrates the architecture of the interface unit showing its main components.

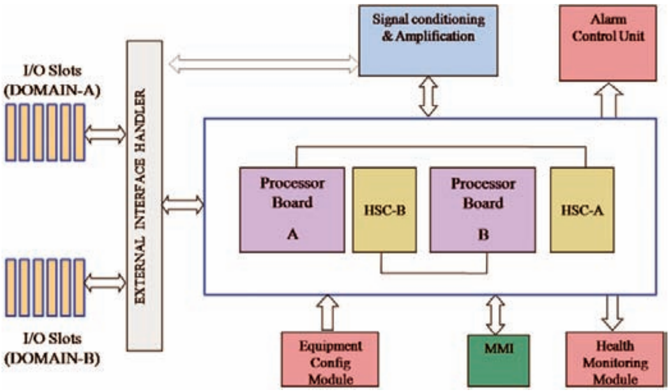


Figure 2. Architecture of the integration unit.

4.2 OS Kernel Layer

Linux Operating System (Kernel 2.6) was chosen for the design of the integration unit. The O(1) scheduler of Linux 2.6 Kernel provides greatly increased scheduling scalability with a very low interrupt response time. Device drivers of the I/O cards were enhanced to support hot swap functionality. Drivers were modified so that the I/O card could cease all activity when it is about to be hot swapped. Further, high availability device drivers were required to be able to enter a standby mode while bus control was being passed from one CPU to another, without crashing the whole system.

4.3 Application Layer

The application layer of the interface unit consists of three software modules that work together to provide a high available environment. The core application module was responsible for delivering relevant data to the respective consumer equipment. Health check module continuously checked the health of the processor in the other domain and reported any anomaly to the core application module. Event management module interacted with the I/O board drivers and reported the event of board insertion/extraction in the live system to the core module for taking necessary actions.

5. FUNCTIONALITIES PROVIDED BY THE INTERFACE UNIT

5.1 System-level Functionalities

5.1.1 Dual Redundancy

As the integration unit has been designed for military operation, it is required to eliminate single points of failure. Hardware duplication and network redundancy are common techniques utilised for improving the reliability and availability of such systems[7]. For the integration unit the active-standby model of redundancy handling was used. In this configuration, one CPU manages all the twelve I/O slots. In addition, the second CPU serves as a warm standby, ready to run the system in the event of a failure on the active system. Redundancy for processor boards, power supply units and inter-system communication channels were catered, as these are the elements whose failure would bring down the whole system. Redundancy for I/O boards was not provided in the current version, but the failure of an I/O board was reported to the operator. The operator could then replace the faulty board while the system is up and running without hampering the ongoing system operations.

5.1.2 *Hot Swap Capability*

The system components namely, processor boards, I/O boards, and power supply units in the integration unit are hot swappable, allowing the system to continue operating in the face of component failures and could be repaired while application service continues. For making this possible, the drivers for these boards were modified to include hot swap-enabled features. The complete process of hot swap involves physical, hardware, and software connection processes. The physical connection process is the basic process of putting a board into a live system. This powers up the board and enables it for access by a PCI bus transaction in PCI configuration space. The board then runs its power up diagnostics, initialises itself, loads EEPROM data, and then becomes operable from the hardware perspective. Now, the system initialises the board's PCI configuration space registers with I/O space, memory space, interrupts and PCI bus numbers and makes it ready to be accessed by a device driver. After the device driver is loaded, it is ready for use by the application. Similarly, extraction was initiated when the operator opens the board ejector handle, which activates a mechanical switch. The hot plug system driver senses this and notifies software that board activity must be stopped and that software device drivers should be unloaded. The application, that is using the board, is informed that the resource is no longer available. After extraction, all system resources previously assigned to that board are made available for other uses.

5.1.3 *Inter-System Communications Services*

Inter-system communications services (ISCS) provide a mechanism for data communication between the two domains of the integration unit. It allows applications to communicate between the two domains. It provides a robust interface between the two domains with redundant serial connections and a network interface.

5.1.4 *Real-time Operation*

In the design of the integration unit, not only the correctness of the data matters, but it is also important to deliver the processed data to its destination within a predefined time[8]. This type of situation is best handled using a real-time operating system (RTOS). In the current version of the system, Linux Operating System (kernel version 2.6) was used. Though it is not a hard core RTOS, but remarkable improvement in the performance of the interface unit was observed after shutting down unnecessary daemons. This greatly improved the response time, thereby serving our purpose.

5.2 Application-level Functionalities

The proposed integration unit has the capability to handle almost all types of interfaces that are present on a typical naval ship. The application for the interface unit has been implemented using C language. The core application has been developed using multi-threading concept as it involves a mix of multiple I/Os and CPU intensive operations. Broadly, the core application is composed of two modules, viz., equipment interface module and processing module.

For each transmit and receive channel of the equipment connected to the interface unit, a thread (equipment thread) was created in the equipment interface module. The purpose of an Equipment thread is to open the required device/ port, configure it as per the associated interface type and then write or read data from it. Processing module consists of processing threads, whose typical functions involve extraction of particular bytes from an incoming packet, forming a new packet from multiple packets, padding of application-level checksum to an incoming packet and other types of processing as required by the end equipment. The processing module also handles the rate at which data is required by the destination equipment. The equipment interface module and the processing module communicate through message queues.

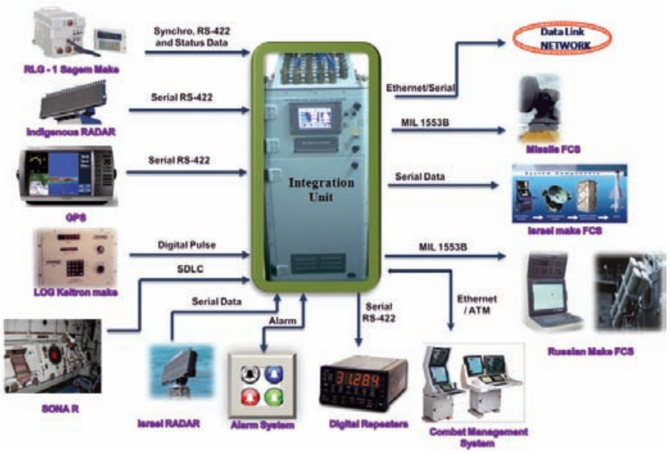


Figure 3. Connectivity of integration unit with various weapons and sensors.

To clarify the above-mentioned software architecture, let us consider a hypothetical scenario where only one source equipment-ring laser gyro (RLG) and one sink equipment–analog dial display, are connected to the interface unit. Consider that the RLG gives its output in asynchronous serial format on RS422 signal level, and that the packet consists of information about multiple fields. Let us consider that the sink equipment needs information about the course of the ship in synchro format. In this particular scenario, as two equipment/ channels are involved, so two message queues would be created. In the equipment interface module, two threads would be created. The first equipment thread would open the port (source equipment) of the interface card on which output of RLG is terminated, it would then configure itself with all the required serial parameters such as baud rate, number of stop bits, etc., and then would wait to receive any data from the port. When data is received, it is sent to the first message queue. The second equipment thread would open the port of the interface card on which the dial display is connected, and then configures the card as per the required format, say 90 V_{l-l}, 400 Hz Synchro format. Now this thread would wait for any data that is available on the second message queue. When data is available on the second message queue, it is picked up by this thread and then written to the destination port. In the processing module, one thread would be created. The processing thread receives RLG packet from the first message queue, extracts the bytes that contain course information, performs weighted multiplication as per the end equipment's requirement and then sends the result to the second message queue.

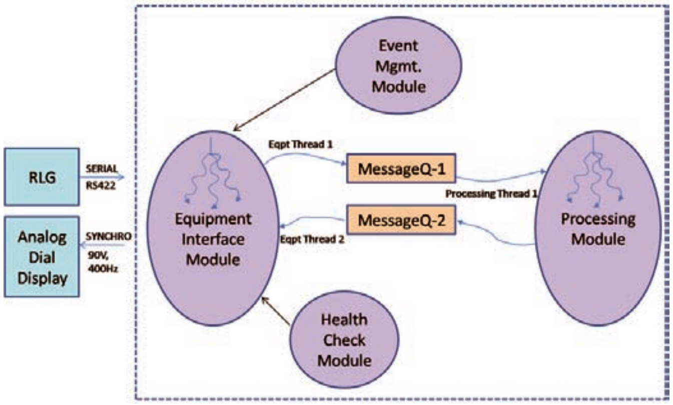


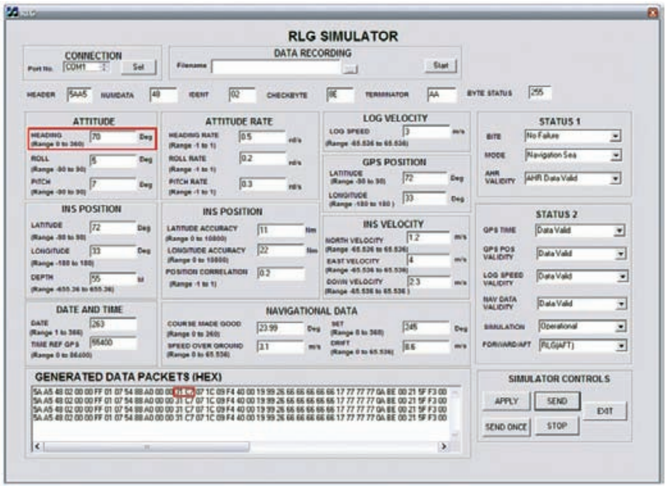
Figure 4. Data acquisition and processing mechanism in the integration unit.

6. IN-HOUSE VALIDATION

6.1 Methodology

The Unit has been tested under full load by feeding data from in-house developed equipment interface simulators prior to deployment onboard ships. These simulators are either PC-based or microcontroller-based units that mimic the signals of the original equipment on their Input/Output lines. For example, RLG interface simulator is a PC-based interface simulator that has been implemented using Microsoft Visual C++. A snapshot of the RLG interface simulator screen is shown in Fig. 5. This simulator generates navigational and stabilisation data in the same format as that generated by the actual Sigma-40 RLG system fitted on a naval ship. A commercial USB to RS422 converter was used in this simulator to generate signals on RS422 signal levels.

The graphical user interface (GUI) of an interface simulator allows a user to set different values for all the parameters relevant to the equipment. For instance, in case of RLG Interface Simulator, a user can set valid values for parameters such as heading, roll, pitch, speed over ground, etc. as shown in Fig. 5. When the 'Apply' button in the simulator is clicked, the program constructs a data packet from the entered values, taking into consideration the multiplying factors as defined in the equipment interface protocol document (EIPD). The content of this data packet is shown at the bottom part of the RLG interface simulator. When 'Send' button is clicked, the generated data packet is sent out periodically at a refresh rate of 10Hz. If we consider Heading information of ship from the designated RLG, EPID says that it should be represented in 2 bytes and weight of most significant bit (MSB) represents heading angle of 180 degree. Thus, if a user enters a value of 70 degree in the RLG Interface simulator as the Heading angle, then the corresponding 2 bytes of data would be (2^{15} x 70)/ 180, that is , 12743 or 31C7 in Hexadecimal as shown in Fig. 5.



Port No. COM1

Filename

HEADER 5AA5 NUMDATA 48 IDENT 02 CHECKBYTE 8E TERMINATOR AA BYTE STATUS 255

HEADING (Range 0 to 360) 70 Deg
ROLL (Range -90 to 90) 5 Deg
PITCH (Range -90 to 90) 7 Deg

HEADING RATE (Range -1 to 1) 0.5 rd/s
ROLL RATE (Range -1 to 1) 0.2 rd/s
PITCH RATE (Range -1 to 1) 0.3 rd/s

LOG SPEED (Range -65.536 to 65.536) 3 m/s

LATITUDE (Range -90 to 90) 72 Deg
LONGITUDE (Range -180 to 180) 33 Deg

BITE No Failure
MODE Navigation Sea
AHR VALIDITY AHR Data Valid

LATITUDE (Range -90 to 90) 72 Deg
LONGITUDE (Range -180 to 180) 33 Deg
DEPTH (Range -655.36 to 655.36) 55 M

LATITUDE ACCURACY (Range 0 to 10800) 11 Nm
LONGITUDE ACCURACY (Range 0 to 10800) 22 Nm
POSITION CORRELATION (Range -1 to 1) 0.2

NORTH VELOCITY (Range -65.536 to 65.536) 1.2 m/s
EAST VELOCITY (Range -65.536 to 65.536) 4 m/s
DOWN VELOCITY (Range -65.536 to 65.536) 2.3 m/s

GPS TIME Data Valid
GPS POS VALIDITY Data Valid
LOG SPEED VALIDITY Data Valid
NAV DATA VALIDITY Data Valid
SIMULATION Operational
FORWARD/AFT RLG(AFT)

DATE (Range 1 to 366) 263
TIME REF GPS (Range 0 to 86400) 86400

COURSE MADE GOOD (Range 0 to 360) 23.99 Deg
SPEED OVER GROUND (Range 0 to 65.536) 3.1 m/s
SET (Range 0 to 360) 345 Deg
DRIFT (Range 0 to 65.536) 8.6 m/s

```
5A A5 48 02 00 00 FF 01 07 54 88 A0 00 00 31 C7 07 1C 09 F4 40 00 19 99 26 66 66 66 66 66 17 77 77 77 0A BE 00 21 5F F3 00
5A A5 48 02 00 00 FF 01 07 54 88 A0 00 00 31 C7 07 1C 09 F4 40 00 19 99 26 66 66 66 66 66 17 77 77 77 0A BE 00 21 5F F3 00
5A A5 48 02 00 00 FF 01 07 54 88 A0 00 00 31 C7 07 1C 09 F4 40 00 19 99 26 66 66 66 66 66 17 77 77 77 0A BE 00 21 5F F3 00
5A A5 48 02 00 00 FF 01 07 54 88 A0 00 00 31 C7 07 1C 09 F4 40 00 19 99 26 66 66 66 66 66 17 77 77 77 0A BE 00 21 5F F3 00
```

Figure 5. Snapshot of an equipment interface simulator.

6.2 Results and Discussions

To test the performance of the integration unit in lab, eight equipment interface simulators were connected to the integration unit. List of these simulators with relevant information is given in Table 1.

For evaluating the performance of integration unit, it was fully loaded by connecting all the eight equipment interface simulators for a period of 48 h and various performance parameters were monitored. Result of this setup is given in Table 2.

The results show that the average latency for data from source-to-sink equipment/systems through the integration unit is well within the permissible limit for use in navigation and tactical applications onboard naval platforms. Nil packet drop observed over the experimental period ascertain high reliability of the unit to handle data delivery. By analysing the results after loading the unit with multiple types of equipment simulators, it is evident that the unit is capable of catering equipment with diverse electrical interfaces, update rates and data transfer rates. Very low memory utilisation and negligible CPU usage observed in our experimental setup show that the unit is scalable to handle data from more number of equipment/systems. Based on these findings it can be seen that the integration unit has the potential to be used for reliable transfer of information among various navigational and tactical equipment/systems.

7. CONCLUSION

The proposed integration unit allows equipment, sensors and weapon systems from diverse origin to work in tandem to enhance operational capability of a naval platform. This approach of systems integration is different from that used in the past where hardwiring of individual systems was the only viable approach. These traditional method led to large cabling

Table 1. List of equipment interface simulators used

Equipment simulator	Source/ Sink	Refresh rate	Data rate	Important parameters	Signal type
RLG (navigational/tactical)	Source	10 Hz	38.4 Kbps	Heading, roll, pitch, velocities, deck inclination angle	Serial (RS 422)
EM Log (navigational)	Source	-	100/ 200 Pulses per Nautical Mile	Ship speed and distance covered	Pulse
GPS (navigational)	Source	1 Hz	4800 bps	Latitude, longitude and associated information	Serial (RS 422)
Surveillance radar (tactical)	Source	10 Hz	1 Mbps	Target tracks	MIL 1553B
Analog dial (navigational)	Sink	-	-	Ship heading	Synchro
Digital display (navigational)	Sink	1 Hz	38.4 Kbps	Ship heading, speed and distance covered	Serial (RS 232)
Missile decoy system (tactical)	Sink	10 Hz	38.4 Kbps	Wind speed, wind direction, heading	SDLC (RS422)
Combat management system (tactical)	Sink	Event based	1000 Mbps	Track data, heading, roll, pitch, wind speed/ direction	Ethernet

Table 2. Experimental results

Performance parameter	Average value
Latency	23 ms
Packet drop	0
Memory utilisation	39%
CPU usage	4%

overheads and required cumbersome system integration and maintenance activities by operators. Our approach uses cPCI-based COTS I/O cards and open source software and supports all types of interfaces that are used on a typical naval platform. The main strength of this design is its scalability feature in terms of software and hardware and the ease of maintainability. Depending upon a naval platform, suitable I/O cards can be fitted in the integration unit and the application software is accordingly configured to suit the requirement of the chosen platform. The integration unit provides additional features like health monitoring, soft real-time operations, dual-redundancy and hot-swap capability, thus making the integration unit a state-of-the art system for various naval platforms.

8. FUTURE WORK

As the integration unit requires extensive handling of data, it is envisaged to use a data-centric software tool in the future for efficient handling of data. Cheng and Liao[9] used cloud computing platform to perform fusion of huge volume of military intelligence information from various sources in an efficient manner. Ye[10], *et al.* have used publish-subscribe-based technique for data fusion and data dissemination in a sensor grid. In our future work, we intend to use data distribution service (DDS)[11]. Publisher-subscriber paradigm of DDS will allow the developers to focus on processing of data, leaving all the complexities involved in transmission, reception and QoS maintenance to the DDS software. In the DDS scenario, the environment would consist of a set of publishers and a set of subscribers. A publisher publishes its data (also called topics) with the configured QoS parameters. A subscriber that subscribes to a particular topic receives it for its consumption. For example, Nav Data, consisting of Ship's course, speed, latitude and longitude, can be considered as a topic for publishing. Any equipment that wants this Nav data needs to subscribe to it. The QoS parameters of DDS ensure deterministic delivery. This approach is also ideal for safety-critical fault-tolerant systems as it is capable of supporting many-to-many connectivity with redundant nodes.

REFERENCES

1. Lenzerini, Maurizio. Data integration: A theoretical perspective. *In* Proceedings of the 21st ACM Symposium on Principles of Database Systems, New York, 2002. pp. 233-246. doi: 10.1145/543643.543644
2. Waltz, Edward L. Information warfare principles and operations. Artech House, MA, USA, 1998.
3. Zhu, Q.; Aldridge, Stuart L. & Resha, Thomas N. Hierarchical collective agent network (HCAN) for efficient fusion and management of multiple networked sensors. *Information Fusion*, 2007, **8**(3), 266-280. doi: 10.1016/j.inffus.2005.04.002
4. Sycara, K.; Glinton, R.; Yu, Bin; Giampapa, J.; Owens, S.; Lewis, M. & Grindle, L. An integrated approach to high-level information fusion. *Information Fusion*, 2009, **10**(1), 25-50. doi: 10.1016/j.inffus.2007.04.001
5. Hoang, N.; Jenkins, M. & Karangelen, N. Data integration for military systems engineering. *In* Proceedings of the IEEE Symposium and Workshop on Engineering of Computer Based Systems, 1996. pp. 11-15. doi:10.1109/ecbs.1996.494504
6. Khaleghi, B.; Khamis, A.; Karry, F. & Razavi, S. Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, 2013, **14**(1), 28-44. doi: 10.1016/j.inffus.2011.08.001
7. Okamoto, S. & Sycara, K. Augmenting ad hoc networks for data aggregation and dissemination. *In* Proceedings of the 28th IEEE Conference on Military Communications, 2009, pp. 2346-2352. doi:10.1109/milcom.2009.5379977

8. Bhagiratharao, E. Real time information fusion in military systems. *Def. Sci. J.*, 1990, **40**(1), 71-82.
9. Cheng, X. & Liao, X. The application of cloud computing in military intelligence fusion. *In* Proceedings of the International Conference of Information Technology, Computer Engineering and Management Sciences, 2011, pp. 241-244. doi:10.1109/icm.2011.217
10. Ye, Z.; Chen, H. & Wu, Z. Dart-Dataflow: Towards communicating data semantics in sensor grid. *In* Lecture Notes in Computer Science, 2005. pp. 517-522. doi: 10.1007/11590354_65
11. Tambe, S.; Aranda, F.G. & Schlesselman, J. An extensible architecture for avionics sensor health assessment using data distribution service. *In* Proceedings of the AIAA Infotech@Aerospace Conference, Boston, August 2013. pp.1434-1444. doi: 10.2514/6.2013-5139

CONTRIBUTORS

Mr Kishor Choudhury received his MTech (Computer Technology) from IIT Delhi, in 2012. He is presently working as Scientist 'D' at the Weapons and Electronics Systems Engineering Establishment (WESEE), New Delhi. He has received CNS Commendation in 2006 and DRDO Technology Group Award in 2008. His field of interest includes algorithms, embedded systems, and computer vision. He has contributed in the design and development of interface units for various ships and submarines.

Mr Ashok Kumar obtained his BTech (Electronics and Communication Engineering) from UP Technical University, Lucknow. Presently he is working as Scientist 'C' at Weapons and Electronics Systems Engineering Establishment (WESEE), New Delhi. His significant contributions are in interfacing of weapons, sensors, combat management systems and other ship systems for Indian naval warships. He is also involved in development of interface simulators to carry out testing and validation onboard ships and submarines. His area of research includes: Weapons, sensors and navigation systems integration, data communication and embedded systems.

Mr Wareendra Nath received his BTech (Electronics and Communication Engineering) from National Institute of Technology, Patna, in 2009. He is presently working as Scientist 'B' at the Combat System Integration Group of Weapons and Electronics Systems Engg. Establishment (WESEE), New Delhi. His areas of work include system integration, data communication, opto-electronics and networking. He has contributed in the design and development of interface units for various ships and submarines.