Establishing Low Cost Aquatic Monitoring Networks for Developing Countries

Jarrod Trevathan[1], Ian Atkinson[1], Wayne Read[1], Ron Johnstone[2],
Nigel Bajema[1], and James McGeachin[1]

[1] eResearch Centre, James Cook University, Australia
eResearch@jcu.edu.au

[2] School of Geography, Planning & Environmental Management/Centre for Marine Studies,
University of Queensland, Australia
rnje@uq.edu.au

Abstract. Effective monitoring of natural resources in developing countries is vital to ensuring the sustainability of the environment and major industries such as aquaculture. Sensor networks are a promising technology that can be employed to remotely gather real-time data on important environmental parameters. This data can be exploited by operators/policy makers to better manage environments and maximize the production from aquaculture ventures. However, developing countries face numerous problems in deploying and using sensor network technology. The main issues include cost, limited technological infrastructure, and inexperience with collecting, storing and analyzing data. This paper examines these issues and defines the level of quality that would be sufficient for providing developing countries with usable environmental data. We describe how a country's existing infrastructure can be combined with scalable middleware (SAL) to integrate disparate technologies. We also present results from a test sensor network featuring heterogeneous technologies that is being used for environmental monitoring.

Keywords: Sensor networks, aquaculture, enabling technologies, middleware integration platform, communications infrastructure.

1 Introduction

The commodity of the natural environment is a precious reserve and means of income for many *developing countries* (DCs). Conserving the natural environment is vital to ensure the sustainability and economic progress of such countries. However, it is all too common that environmental degradation is the first thing to occur when these economies move along a path towards industrialization. For many South-East Asian countries, the harnessing of the natural environment to support aquaculture is an important economic resource and provides much needed protein for their populations. As such, these ventures are extremely sensitive to pollution and human activity.

The main reason for environmental damage in DCs is the lack of government policies, limited means to monitor resource usage, and the inability to act on such information if it was available. *First world countries* (FWC) are now starting to

employ sophisticated *sensor network* (SN) systems to augment traditional environmental monitoring programs used to monitor and analyze environmental phenomena. However, there are many issues facing DCs which presently has made these potentially more effective approaches unviable. Significant obstacles are associated with the costs related with setting up a monitoring system, and also the availability of IT and engineering expertise. Many examples of systems in FWCs have been overly expensive, and for the most part ineffective. Furthermore, generally DCs have very limited ICT infrastructure. Such infrastructure is required to collect, collate and communicate the information to, and perform analysis on the behalf of policy makers/end-users. Furthermore, whatever infrastructure policy makers have access to can be disparate and tends not to be designed to smoothly integrate.

In recent years, middleware software solutions have emerged that can be used to integrate heterogeneous IT [1, 2, 3, 4, 5]. The effect of this middleware approach is that the existing IT infrastructure in a DC, no matter how basic or incompatible can be used to establish basic SNs for environmental monitoring. Despite only possibly having access to older and/or obsolete hardware/software platforms, SN middleware technology can bridge the gaps between hardware systems and allow DCs to use low cost and pervasive sensors. When combined with emerging standards such as Sensor Web Enablement [6, 7], data collected from SNs can be archived so that external monitoring agencies can view and analyze the data. Such end-to-end monitoring systems will be of great value for remotely monitoring aquatic environments and assist, for example, in managing the sustainability of the aquaculture industry.

This paper examines the issues facing DCs in deploying SNs for remote aquatic monitoring, and contrasts the approaches taken in more developed countries. This analysis allows us to define the level of quality that would be sufficient for providing DCs with valuable data for the purposes of environmental monitoring and management. We describe how software mechanisms can be used to integrate disparate hardware and communications technologies existing in DCs. The software middleware we have developed (referred to as *SAL* [1, 2]) is scalable and can support a SN of almost any complexity. Finally, we present real world examples of how the proposed software is being used for environmental monitoring, and how it is being tailored specifically for aquatic settings.

This paper is organized as follows: Section 2 discusses the issues facing DCs and draws comparisons with FWCs in terms of establishing viable and sustainable SN systems. Section 3 presents the software technologies we are developing in order to address some of these issues. Section 4 gives a series of test deployments to illustrate its scalability and resourcefulness to handle SNs containing dissimilar technologies. Section 5 examines the SEMAT project and our contribution to the development of its technologies. Section 6 provides some concluding remarks.

2   Issues Facing Developing Countries

This section describes the issues facing DCs for the deployment and management of SN systems. We contrast these issues with the ICT requirements for SNs and the lessons learned from projects in FWCs. This allows us to define the quality of service and the minimal technological infrastructure needed to deploy a network that will at the very least provide valuable data for end-users.

The underlying core issues for deploying environmental monitoring systems in DCs are basically the same problems as in FWCs. However, there are several features that make key differences. Firstly, FWCs generally have access to some pervasive broadband infrastructure. That is, communications technology is usually high speed, reliable and has wide coverage. Secondly, FWCs have access to an abundant amount of current generation computing hardware and software platforms for collecting, archiving and analyzing data. Finally, most FWCs have established policy making/endorsing bodies that use the collected data to ensure that phenomena under study conforms to the established policies and legal practices.

In contrast, DCs often only have access to basic ICTs. However, there is a very large range – between the ultimate extreme of having absolutely nothing, or being relatively in league with FWCs. In any case, we have to allow for the fact that there will be heterogeneous technologies which might be as basic as ordinary radio communication, or it might be equivalent to FWC broadband and cellular network standards. Computing and software platforms within DCs are often older, or in many cases obsolete compared to the latest technology in use by FWCs. Furthermore, there may be no established policy making mechanism, or this may be primitive with only a mediocre level of effectiveness. The resounding question this raises is, *"what is the minimum technological infrastructure required to establish an environmental monitoring SN system in a DC"?* To answer this question it is prudent to examine similar systems for FWCs.

An example of a system is not particularly suited for use in a DC is being established by the *Australian Institute of Marine Science* (AIMS)[1] as part of the *Australian Integrated Marine Observing System* (IMOS). This project involves the deployment of remote sensing buoys at strategic locations around the Great Barrier Reef[2] [8]. Their purpose is to detect oceanographic features such as salinity, temperature, light, etc. However, the solution falls short for use in by a DC in multiple ways. The first major shortcoming is cost. Each buoy costs an estimated $50,000 AUD, with deployment and maintenance also requiring significant technical infrastructure and expertise. Furthermore, it is not pervasive and gives only localized answers. That is, there are only four sampling points across a reef that spans 344,400 square kilometers. Therefore, the effectiveness of the data for any sort of modeling is limited. As such, this sort of approach is unsuitable for a DC.

With this example in mind, let us examine the features of cheaper alternatives. Regardless of the cost, the system must be able to deploy "enough" sensors to measure with "sufficient" accuracy to present useful information. (Note that less precise assumes sufficient accuracy.) Four million cheap sensors with limited precision are arguably of much greater value than four expensive sensors of extremely high precision. Furthermore, four million sensors will cover a larger geographical area thereby maximizing the coverage for the network and environments sensed. In terms of an aquatic setting, the geographic dimensions are important. Sensor readings are not just required horizontally between locations, but also need to be at multiple points vertically between the surface and the bottom of the water body. This makes the cost of the sensors extremely important in order to take the most measurements.

---

[1] www.aims.gov.au

[2] http://imos.org.au/faimms.html

Fig. 1. The cost spectrum for SN systems

Figure 1 illustrates the cost spectrum for SN systems. As the design choice moves between the extremes, the features of either extreme will become more pronounced. A cheap system will allow for a pervasive network of low-cost sensors and gateways to be deployed on a potentially massive scale. Whereas expensive systems (as in the AIMS example [8]) can only result in a few sensor nodes thereby giving localized readings. Low-cost systems potentially have lower power requirements and the components are physically smaller than in an expensive infrastructure. However, cheap systems typically do not have the same endurance and life span as more expensive setups. Although the pace of improvement in technology means that sensors can be outdated within two/three years in any case. In general, cheap sensor nodes can be easily replaced which means their lifespan may not be a significant limiting factor. Furthermore, expensive systems that are damaged, lost or stolen are quite difficult to replace making less expensive systems more attractive. While expensive systems are highly proficient (and precise) due to their sophistication, cheap systems are usually capable albeit with less precision.

In light of the cost structure and requirements, we define the parameters for a suitable/feasible solution for remote SNs in DCs as follows:

- The system must be cost effective, to provide the most coverage and is affordable for a DC. (Sensors embedded in packaging can be as cheap as $10-$15 each.)
- It must use existing hardware and communication infrastructure. Even though solutions may interconnect to high technology front ends when they are available, there must be basic, low cost solutions that do not depend on any advanced technology. For example, a SN deployment using second hand/obsolete computers with very low cost sensors communicating via early technology mobile networks. There is also scope to capitalize on initiatives that telecommunications companies are offering DCs. For example, Ericsson in collaboration with the *Swedish Program of ICTs in Developing Regions* (SPIDER) has initiated investigation of wireless SNs in developing regions, including the allowance for bandwidth on GSM mobile phone networks to be used for environmental monitoring.[3]
- The data collected must be valuable to the end users and accessible to all interested parties. External bodies such as a government or a United Nations agency can in theory access the collected data. This would aid in the establishment of sound, or in the very least accountable environmental management policies.

---

[3] http://www.spidercenter.org/project/national-research-
and-development-center-wireless-ad-hoc-and-sensor-networks-wasn

It is interesting to note that low technology solutions may also be viable in some locations and not others. For example, deploying and manually cleaning sensors on a daily basis (as would be required with some very low-cost infrastructure) would be cost prohibitive for most FWCs. However, this solution may in fact fit very well with the structure of a DC where labor can be both effective and relatively inexpensive. This may be a consideration that affects any SN deployment in a DC.

Much of the hardware that is present in a DC typically will come at least in part *via* aide from FWCs. It might be in the form of money to purchase the equipment, but it is more likely to come in the form of donated second hand equipment. Whatever, the method for acquiring the hardware, the underlying point is that there will be a menagerie of heterogeneous and incompatible technologies to work with.

To achieve a system that satisfies the aforementioned constraints requires the use of a software stack to handle the range of communication and hardware technologies. That is, a middleware solution to bring together disparate technologies which range in computational power and cost.

3   Scalable Software Solutions for Abstracting Hardware and Communications Technologies

Lack of standards makes it difficult to integrate heterogeneous sensors in a single SN. Therefore it is important to have middleware which is able to manage all types of sensors. This middleware is often implemented specifically for one SN, which makes it very hardware dependent. Changes in the network, such as adding a new sensor, lead to the need for manipulating the code of the middleware. This section describes the *Sensor Abstraction Layer (SAL)* which is designed to address these issues.
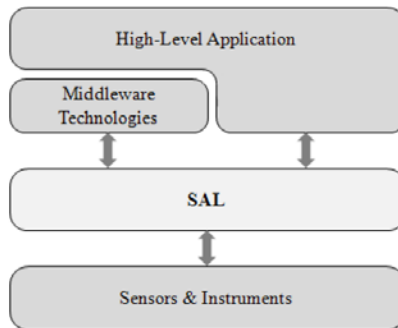
Fig. 2. Architectural overview of a system stack using SAL

SAL is a middleware which provides a plug-in-based model where support for new types of sensors can be loaded to the running system via plug-in. The system automatically detects and configures new sensors, if permitted by the hardware and underlying OS. Furthermore, it provides a unified interface to all sensors by abstracting the sensor specific features. With this it aims at simplifying the access to a SN as well as the management and control of its sensors [1, 2].

SAL can be seen as a low-level software layer as it bridges a network of sensors with high-level applications or further middleware technologies, e.g., a ring buffer. Figure 2 presents an architectural overview of a system stack using SAL.

SAL is implemented as a client-server application and thus consists of two components, the *SAL client* and the *SAL agent*. The *SAL client* represents an interface for SAL to either the user via a user interface or to other applications. It implements the SAL agent *Application Programming Interface* (API) in order to provide the functionality of the SAL agent. The API is grouped into the following categories:

***Sensor Management* –** Methods to manage the pool of sensors and including operations for enumerating, adding and removing sensors.
***Sensor Control* -** Methods to report on a sensor's capabilities and control the streaming of the data.
***Platform Configuration* -** Methods to adjust the platform, e.g., add support for a new sensor type.

Each category uses a different markup language. The *Sensor Management* methods use *SensorML* [6], which describes a sensor's configuration. The methods in the category *Sensor Control* use *CommandML*. The CommandML documents contain a list of commands which are supported by a sensor. The last category, *Platform Configuration,* uses *Platform Capabilities and Configuration Markup Language (PCML)*. Documents of the type PCML contain information on how the platform is to be configured in order to support a certain type of sensor technology.
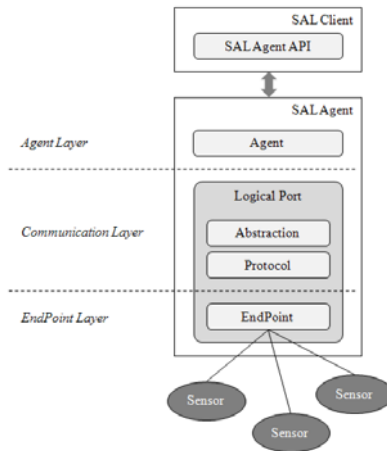

Fig. 3. SAL components and software layers

The *SAL agent* implements the various features of SAL. It runs on a platform which is connected to the sensors and therefore is regarded as a *sensor gateway*. The connection between the platform and the sensors can be either direct using platform specific I/O ports like USB or indirect by using wireless technology. The *SAL agent* manages the sensors which are directly connected and are found by the agent and the

indirectly connected sensors it has been told of. The *SAL agent* consists of three layers: *Agent Layer, Communication Layer,* and *EndPoint Layer* (see Figure 3).

The *Agent Layer* is responsible for the communication with the *SAL client*. It receives messages, parses and forwards them to the underlying *Communication Layer* and sends the response back to the client.

The *Communication Layer* provides methods for managing and controlling sensors. The managing methods are used to configure and set up hardware while the controlling methods translate a generic command into a sensor native command, which then can be transmitted to the sensor. The generic commands are provided in the SAL API. For translating the generic commands the two sub-layers *Abstraction Layer* and *Protocol Layer* are used. The *Abstraction Layer* is an adapter layer where the generic commands are implemented. From here the sensor-specific methods are called, which are implemented in the *Protocol Layer*.

The *EndPoint Layer* is tightly coupled to the I/O ports available on the sensor gateway. It is responsible for transmitting native sensor commands produced by the sub-layer *Protocol* to the sensor. In the other direction, data from the sensor is transmitted to the SAL agent. The software code for this layer is normally included in the operating system. SAL ensures it is available and configured correctly.

The two sub-layers *Abstraction* and *Protocol* and the *EndPoint Layer* together form a *Logical Port*. Each sensor is connected to a Logical Port. SAL allows a specific client to control multiple agents (*horizontal scalability*). Furthermore, the logical port allows SAL to scale vertically (*vertical scalability*) in that an agent can control other agents in a hierarchical manner. This is due to the logical port treating the attached device as a data source, independent of whether it is a sensor or another agent. The ability to scale allows SNs of almost any topology or configuration.

SAL is also integrated with the RBNB Data Turbine developed by the San Diego Super Computing Group [9]. The Data Turbine allows for near real-time streaming of data from multiple sources (as is common in a SN). All incoming data is temporarily stored in a time-stamped buffer prior to being consumed by the end-user. The end-user can essentially fast-forward and rewind the buffered data which is extremely useful for time series analysis and presentation.

SAL has been trialed on the Davies Reef SN project [10] whereby it controls a sensor array via a low cost/power computer in a remote reef. Of greater importance to this paper's objectives is SAL's role in the SEMAT project [11] which aims to develop low cost pervasive aquatic SN systems (see Section 5).

4   A Test Implementation

To show SAL's feasibility for deployment in a low cost heterogeneous environment, we created a series of sample SNs of varying scales. This section describes the test setup and SAL's performance at increasing levels of complexity.

One Client, One Agent – The first test involved using one agent that had multiple heterogeneous sensors attached. There was a single client controlling the agent. The aim for this test was to illustrate SAL's ability to connect, manage and collect data from numerous heterogeneous sensor types simultaneously from different vendors.

The sensors included a one wire dongle sensing temperature and humidity, a Web camera streaming live data, fifteen Java sun SPOTs[4] (which contains a package of multiple sensor types), and a one wire bus containing ten independent thermistors.

For this test the SAL agent and client were run on separate single board computers (WAFER-945GSE SBC from ICP Electronics[5]) which communicated wirelessly. Each computer had a processor speed of 1.6 Ghz and 2 GB of RAM. The price for each computer was $427 AUD. Once registered, each sensor type could be plugged and unplugged without affecting other devices. The number of devices we could attach and the amount of data transferred essentially was only limited by number of USB connections and the network bandwidth respectively.

The tests were then repeated with a Gumstix[6] device. These are miniature computers (approximately the size of a stick of chewing gum) designed to run in purpose built applications. These devices are indicative of what would be deployed in the field for a wireless SN. The processor speed is 400 MHz with 128 MB RAM. Our tests used the Vertex series package which retails for approximately $300 AUD. These tests were designed to examine how SAL performs on lower powered devices as what might occur in a DC.

There was no notable deviation in performance from the tests run on the single board computers. The entire price for the test SN was approximately $3,000 AUD depending on the computers used and the number of desired sensors.

Multiple Agents at the Same Level (Horizontal Scalability) – The second test involved testing SAL's ability to scale horizontally. There was one client controlling multiple agents. Each agent was independent to and peer with each of the other agents. Each agent had a varying array of sensors attached.

This test involved the single board computers and also some relatively old desktop computers (which ranged in processing power and memory capacity). This test was used to examine SALs performance by combining some equipment that might be representative of what is available to DCs.

As per the previous tests, the system performed well. We theorized that SAL can be used on any computer that has the processing power equivalent of the early Pentium range (i.e., 1.6 MHz). The memory requirements are only constrained by the ability to support the Linux operating system and the Java run-time environment.

Multiple Agents at Differing Levels (Vertical Scalability) – The final test determined how well SAL scales vertically where one client that controls one agent. This agent then controlled two agents directly below it, each of which had differing arrays of sensors (refer to the discussion of a logical port in Section 3). The results showed that this worked seamlessly in line with how SAL handles data sources.

In conclusion, regardless of the technology employed, SAL was able to seamlessly integrate the entire SN. Almost any computer from the mid 90s onwards can run SAL. The price for an entire SN can be as little as $500 - which only increases with the number of agents/clients used and the number of and costs of the sensors desired.

---

[4] http://www.sunspotworld.com/
[5] http://www.icp-australia.com.au/
[6] http://www.gumstix.com/

5   Pervasive, Low Cost and Intelligent Aquatic SNs

This section describes the *Smart Environment Monitoring and Analysis Technologies* (SEMAT) project as an example of an end system that could be used in the context of a DC [11]. SEMAT is largely driven by the need to create a low cost intelligent SN system for monitoring aquatic and coastal environments, and importantly the analysis of that data into information which can be used for management and planning. The specific goals for SEMAT are as follows:

1) Underwater wireless communications – As aquatic environments are remote and vast, it is not economically viable or practical to have sensors wired together. Often in such an environment the positioning of the cables represents a significant practical problem. The cable itself is also vulnerable to breakage or degradation over time.

2) Short-range wireless and power transmission – Interconnecting cables for data communications and/or power *in-situ* is complex in a marine environment. SEMAT aims to develop technologies whereby neighboring cables can be connected underwater and inductive methods used transfer power and data between nodes.

3) Plug and Play – A major problem facing the deployment of wireless SNs is the disparate technologies used as equipment must be combined from different manufacturers. Even the simple case of adding a new type of sensor usually involves reconfiguring the entire system so that the end user can view the sensor's output. SEMAT will use SAL to allow new equipment to be added to the network such that it is instantly recognized and configured for use. This is analogous to plugging in a new peripheral device for a computer such as a printer or mouse, which the operating system automatically detects and allows instant use. Making a wireless SN plug and play removes much of the technical overhead for managing the network by novices.

4) Minimal deployment expertise – SEMAT will offer end users a complete package. The end user will only need to choose what sensors they require and SEMAT will auto-configure the necessary parameters. Essentially once deployed, the user could take a laptop down to the beach (for example) and can begin to view the sensed data.

5) Near real time analysis tools – SEMAT will provide software tools that allow data to be streamed in near real-time from sensors. Users will have the ability to buffer large amounts of data and sift through the data at hand using the RBNB Data Turbine [11]. Data collected is put into a format that is recognized by standards bodies (i.e., Sensor Web Enablement [7]) and therefore can be imported into sophisticated data modeling and visualization tools.

6) Intelligent sensors – Sensor nodes in SEMAT will have a level of intelligence in that they have two way communications with each other. This will allow sensors to have a degree of autonomy from the end user such that if there is a sudden change in a condition which affects the phenomena under study, then sensor nodes can communicate with each other to change their parameters to better study the changes in the environment. For example, if one set of sensors detects that significant rainfall is

occurring, it might communicate with the salinity sensor to increase its sensing rate from daily to hourly. There are almost limitless uses for such intelligence.

SEMAT is an ambitious and large project which draws on expertise from multiple disciplines including marine biology, engineering, mathematics, and computer science. There are several parties involved with the development of SEMAT including: DHI Group[7] (Denmark); University of Queensland (UQ)[8] (Australia); James Cook University (JCU)[9] (Australia); and the Torino Foundation (Italy).

DHI and UQ are responsible for the marine science and high level modeling of marine phenomena. For the purposes of SEMAT, this study is related to the prevalence of *Lymbia algae* blooms at Deception Bay in Queensland Australia.

DHI have developed hydrodynamic modeling software that is being used for the initial studies SEMAT will be conducting. The Torino Foundation specializes in developing and commercializing sensor technology. They are tasked with designing purpose built intelligent sensors for the SEMAT system. The initial sensors under development will be geared towards the studies being undertaken by DHI and UQ.

The Information Technology and Electrical Engineering school at UQ is designing the wireless power transfer and communications systems. They are working in conjunction with the Torino Foundation to decide on the electrical standards for the power transfer. UQ is also working in conjunction with JCU on the communications protocols that will be used between sensor nodes and how the information will be relayed back to the gateway devices.

The eResearch Centre at JCU is responsible for providing the middleware (i.e., SAL) and end user interface for the data acquisition, storage and presentation from the SEMAT system. This is perhaps one of the most challenging components of the SEMAT project. JCU is essentially tasked with making the system plug and play and allowing for near real time analysis of the collected data.

Lymbia Algae Study at Deception Bay

Lymbia Algae is a particularly virulent form of algae that attaches itself to sea grass. As the algae grows, it absorbs nutrients from the sea floor and surrounding areas and physically covers sea grasses preventing light reaching the sea grass – essentially killing it off. Once the algal has finished growing (blooming), it breaks off and typically washes up on the beach/shore line. It leaves scaring along the sea bed where it was present and it usually prevents future sea grass from growing.

While it is uncertain what causes the algae to bloom, it is suspected that it forms mainly in warm conditions, shallow and calm water, and human activity (possibly in the form of nutrient run off) may affect its life cycle. The initial case study for SEMAT will be to study the influences that contribute to the spread of Lymbia algae. Ideally the result will be to suggest a plan for environmental management to control its spread in aquatic environments.

In recent years Deception Bay in Queensland Australia has experienced significant Lymbia algae blooms. The bay is largely protected from the greater ocean and doesn't

---

[7] www.dhigroup.com.au

[8] www.uq.edu.au

[9] www.jcu.edu.au

contain any significant wave action. It is shallow with high tide only raising the water depth to approximately three meters at its peak. Deception Bay is situated between the mainland and the southern region of Bribie Island therefore it is heavily influenced by human activity.

As the initial application for SEMAT is for monitoring aquatic environments for marine science studies, UQ and DHI selected Deception Bay as the premiere site for testing the initial SEMAT system. There were several factors influencing this decision including: proximity to Brisbane; prevalence of algal blooms; subtropical environment; shallow water; and calm conditions

Ultimately, the SEMAT development environment will use six different sensors to examine different potential influences and their interaction on Lymbia algae blooms. The types of sensors include: *temperature*, *total light*, *photo synthetically active radiation* (PAR), *salinity/conductivity*, *turbidity*, *pH*, and *dissolved oxygen*. Several data sinks will be deployed at strategic locations around Deception Bay. Each sink will host the aforementioned sensors.

6   Conclusions

DCs are heavily reliant on their aquaculture resources for food and exports. Effective management of this vital resource is critical to ensuring the future sustainability and economic progress of a DC. SN technology can be used to actively monitor resource use to provide end-users, policy makers and external third parties with the information they need in order make more informed decisions. However, excessive cost, lack of infrastructure, and limited expertise/experience, hinders the adoption of SN technology in DCs.

This paper examined these issues and provided a framework for how a DC can use low cost sensors in combination with a country's existing infrastructure to deploy a capable SN for aquatic monitoring. We propose that SAL can be used as a viable middleware solution to establish a SN in a DC regardless of conflicting technologies. Preliminary test results and real world deployments indicate that SAL is both scalable and capable of suiting any network configuration and sensor types. The cost is only constrained by the computers used (which can be as old as a Pentium), and the number and expense of the desired sensors. The proposed solution is being integrated into the SEMAT project as the nexus for a fully plug and play system capable of being easily deployed in any aquatic environment.

Future work involves increasing the magnitude and scale of the tests conducted with SAL. That is, we are aiming to test SAL on a large-scale horizontal and vertical level incorporating sensors from numerous vendors. We are also going to perform a series of 'wet tests' whereby SAL is used with submersible sensors and low cost computing devices as a proof of concept. Finally, SAL's versatility will be further developed and tested in conjunction with the aims of the SEMAT project.

References

1. Gigan, G., Atkinson, I.: Towards a uniform software interface to heterogeneous hardware. In: Proceedings of the International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pp. 429–434 (2008)
2. Gigan, G., Atkinson, I.: Sensor Abstraction Layer: a unique software interface to effectively manage SNs. In: Proceedings of the International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pp. 479–484 (2007)
3. Handziski, V., Polastre, J., Hauer, J., Sharp, C., Wolisz, A., Culler, D.: Flexible Hardware Abstraction for Wireless Sensor Networks. In: Proceedings of the 2nd European Workshop on Wireless Sensor Networks (2005)
4. Heinzelman, W., Murphy, A., Carvalho, H., Perillo, M.: Middleware to support Sensor Network applications. IEEE Network Magazine Special Issue, 6–14 (2004)
5. Rmer, K., Kasten, O., Mattern, F.: Middleware challenges for wireless sensor networks. ACM SIGMOBILE Mobile, Communication and Communications Review 6(2) (2002)
6. Aloisio, G., Conte, D., Elefante, C., Marra, G.P., Mastrantonio, G., Quarta, G.: Globus Monitoring and Discovery Service and SensorML for Grid Sensor Networks. In: Proceedings of the 15th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), pp. 201–206 (2006)
7. Botts, M., Percivall, G., Reed, C., Davidson, J.: OGC Sensor Web Enablement: Overview and High Level Architecture. In: Nittel, S., Labrinidis, A., Stefanidis, A. (eds.) GSN 2006. LNCS, vol. 4540, pp. 175–190. Springer, Heidelberg (2008)
8. Bainbridge, S., Rehbein, M. A., Feather, G., Eggeling, D.: Sensor Networks on the Great Barrier Reef - managing marine sensor data. In: Proceedings of the Environmental Information Management Conference, pp. 19–25. University of New Mexico (2008)
9. Tilak, S., Hubbard, P., Miller, M., Fountain, T.: The Ring Buffer Network Bus (RBNB) Data Turbine Streaming Data Middleware for Environmental Observing Systems. In: Proceedings of the Third IEEE International Conference on eScience and Grid Computing (2007)
10. Huddleston-Homes, C., Gigan, G., Woods, G., Ruxton, A.: Infrastructure for a Sensor Network on Davies Reef, Great Barrier Reef. In: Proceedings of the International Conference on Intelligent Sensors, Sensor Networks and Information Processing (2007)
11. Johnstone, R., Caputo, D., Cella, U., Gandelli, A., Alippi, C., Grimaccia, F., Haritos, N., Zich, R.E.: Smart Environmental Measurement & Analysis Technologies (SEMAT): Wireless Sensor Networks in the marine environment. In: Proceedings of Wireless Sensor and Actuator Network Research on Opposite Sides of the Globe, SENSEI (2008)