# Software Defined Open Architecture Modem Development at CMRE

John Potter, João Alves, Thomas Furfaro, Arjan Vermeij, Nicolas Jourden,
Giovanni Zappa, Alessandro Berni, and Diego Merani
NATO STO Centre for Maritime Research and Experimentation
La Spezia, Italy
{potter,alves,furfaro,vermeij,jourden,zappa,berni,merani}@cmre.nato.int

***Abstract*—This paper covers the first steps in creating a Software Defined Open Architecture Modem (SDOAM). Potentially useful operating environments, platforms and approaches are reviewed as well as relevant work on underwater digital communications at CMRE, including JANUS. A high-level architectural structure, based on a generalisation of the classic OSI communications stack, is proposed, identifying the modules that will make up the system and taking care to include all the features that the major stakeholders will want to see while minimising the investment that will need to be made to migrate from existing systems to SDOAMs. A key new element is the provision of policy engines that will negotiate the switching between different modules in the OSI-like layers. We also propose to formalise the cross-layer linking that is typically found in practical implementations by providing a fully-connected cross-layer framework for all processes to access. Finally, future developments and research directions are identified. The exact definitions of the interfaces between these modules remain to be specified, as is the software support environment on which they will operate.**

***Index Terms*—underwater communications, acoustic, optical, electromagnetic, modems, OSI, software defined networking, software architecture, physical layer, routing, networking, transport, JANUS, DTN, modulation, coding**

## I. Introduction

Software-defined communication implementations date back to 1990 when the first demonstration of the Advanced Development Module (ADM, [1] ) radio was made. These radios were fitted with multiple waveforms that enabled them to do more than one job. From these early attempts at achieving a re-configurable communications capability in the RF domain, the aim has been to achieve improved functionality and system re-usability by complying with an increasing number of established standards.

In the underwater domain, besides the incredible difficulties presented by the ocean, one of the biggest challenges is interoperability. Due to this, different systems using different techniques and methodologies have arisen in industry and academia which follow no standardisation agreements resulting in a number of solutions which are totally incompatible with one another. At this point it is unrealistic, and not necessarily desirable, that manufacturers should be willing to give up their investments in technology development and discard their proprietary coding schemes. A promising way forward is to instead develop an open architecture modem standard where the different pieces of the protocol stack developed by the various stakeholders can be combined to work together following a OSI communication stack model or similar cross-layer approach. The challenges of the underwater acoustic channel environment for communication dictate the efficient use of the available communication bandwidth with improved performance in terms of the in-service probability and supported data rate while keeping the energy consumption at a minimum. This will require flexible and adaptable platforms based on open standards. Rigid, closed, proprietary all-in-one implementations of acoustic modems are a severe bottleneck to the progress needed to fulfil these needs. Open and modular architectures are an attractive solution since they allow the reuse of already developed modules to enhance new components in a given architecture. To enable the reuse of protocol implementations (from physical coding to routing and application support) and cross-application flexibility across different modems and test beds, we need to move from the existing monolithic integrated modems and communications stacks to Software-Defined Open-Architecture Modems (SDOAMs).

## II. Existing Approaches and Solutions

The concept of a software defined communications system has been around for several decades, with primary interested being related to terrestrial RF applications, [2]. Software-Defined Radio (SDR) for air-borne systems exists as standards like the US's Joint Tactical Radio System (JTRS, [3] ) and NATO's STANAG 5066 [1], and in implementations like the JTRS's Software Communications Architecture (SCA, [4], [5]) or GNURadio. Such SDR systems typically provide software reconfigurability for physical coding and media access control, and then use more traditional OSI-like standards and implementations (TCP/IP for example) on top of the customized, lower layer configurations.

In the underwater regime, providing physical layer compatibility with media access control is not sufficient to adapt traditional terrestrial algorithms and techniques, as the defining characteristics of the media for communications are so

[1]STANAG 5066: Profile for High Frequency (HF) Radio Data Communication

978-1-4799-7578-5/14/$31.00 

drastically different. Existing SDOAM-like solutions and approaches vary between providing physical layer reconfigurability and providing full blown application- and network-layer solutions. SUNSET, [6], and DESERT, [7], provide MAC, networking, and application-level functionality and reconfigurability, but relies on modems themselves to manage modulation and coding. UnetStack [8] goes slightly further, providing interfaces for configuring the modulation and coding scheme, but like SUNSET and DESERT requires host hardware on which to operate. EvoLogics WiSE, [9], [10], [8], and Subnero, [11], modems provide such a host environment, as well as some level of physical layer reconfigurability. Modems like UANT, [12], rModem, [13] and that presented in [14] focus on the provision on low-level interfaces for other higher-level software suites to utilise.

The common theme in existing SDOAM-like solutions is their software-defined nature, while the interface with that software-defined infrastructure is disparate and varying among implementations. It is in this gap that CMRE's SDOAM exists, in the future providing a standard interface between the constituent component that together make a complete communications stack.

## III. Proposed Architecture

The CMRE-proposed SDOAM solution is designed to maximise flexibility for stack developers and providers, by provisioning a simple overall architecture with similarly simple interfaces and cross layer communications support.

The proposed software-defined open architecture modem follows a modular cross-layer approach, in contrast with the typical rigidly-layered OSI implementations. The OSI model is a conceptual framework and the gap between OSI-driven implementations and the model is sometimes significant. The OSI model may in fact be too complex or overly structured for the UW communications environment. Some technical points related to security are difficult to categorise in the OSI model and some redundancies, namely in regards to flow control, can be found.

The choice for departure from the OSI layered modem was motivated by some specific needs of underwater communications such as:

- Limiting the amount of overhead imposed by each layer;
- Exploring direct "crossing" links between layers to make "smart" decisions;
- Skipping the higher layers as they are usually not applicable in underwater networks or can simply exist in the application domain.

In Fig. 1 we depict CMRE's proposed SDOAM model. Conceptually, the SDOAM is composed of a series of modules with an internal communication infrastructure used by the different entities to exchange information. The modules each contain a Policy Engine (PE) and at least one Protocol Module (PM). For analogy with the OSI, we retain the name "layer" for the Physical, MAC, Network and Transport modules. These modules depicted on the left side of the cross-layer framework are directly responsible for packet processing. New information aggregators proposed by CMRE are to the right. These provide networking services that run on information collected by the PEs. Examples of such services are shown, such as network trust and reputation models, based on neighbour nodes activity.

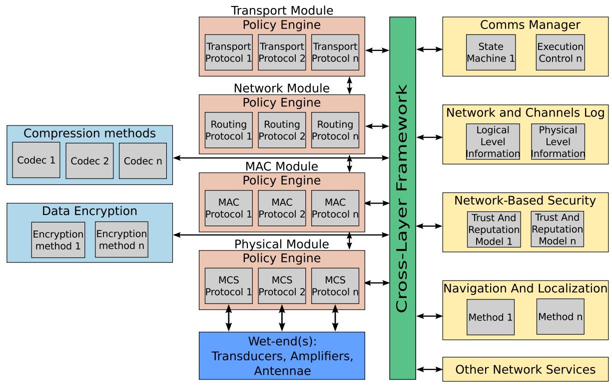


Fig. 1. The proposed block architecture. Note that Compression and Encryption exist as available methods that can be called at any point during packet processing.

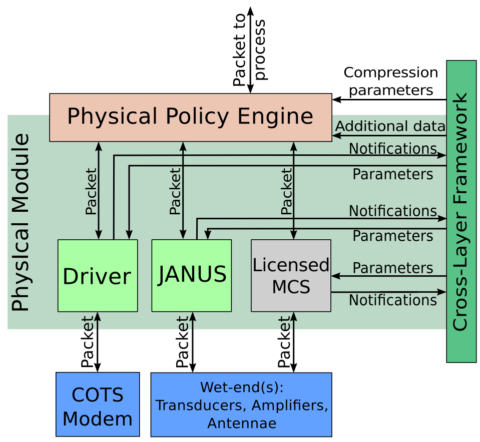


Fig. 2. The physical layer with three different MCS modules.

The main objectives of the concept are to provide a model that accommodates protocol switching within a layer, grouped by the PE, and that allow this engine and individual protocol implementations to access available cross-layer information. The proposed architecture has the potential to facilitate research and experimental solutions while promoting commercial deployments, as some of the protocols may be linked as proprietary black boxes. An advantage of such approach is that it can accommodate different transmission media (acoustic, optical, electromagnetic) simply by providing the respective "wet-ends" and suitable protocols.

*Packets:* The basic information unit, known to every module composing the SDOAM, is the packet. Each protocol module is capable of processing and emitting a packet up or down the processing chain. Packets that are being transmitted (i.e., moving *down* the stack) are successively constructed in each

PM, with more information being coded to the packet itself for delivery when the packet is emitted. Packets that are received (i.e., moving *up* the stack) are successively deconstructed from the originally received packet into a derived class of packet that contains additional information. Conceptually, the composition of a packet is similar to that of the TCP/IP protocol – as each packet progresses through the stack vertically, combinations of headers and/or footers are appended or removed from the raw packet. When the packet is being received each layers' meta-data is stripped off from the header and footer, and then published to the cross-layer framework. The core payload is then passed up to the next layer. Conversely, when a packet is being composed, the handling protocol appends a header and possibly footer before the packet is pushed further down the stack.

*Protocol Modules:* Each PM is a self-contained unit that is capable of processing a received or outgoing packet, and can optionally subscribe to the cross-layer framework for configuration and contextual information that may be published by other protocols in the stack. With the proposed architecture CMRE can provide a strong motivation for commercial partners to contribute to the standardisation efforts. Protocols should preferentially be open to the community but when a commercial provider requires intellectual property protection, the model should accept black-box "compiled" blocks and provide an interface for them as is done for open implementations. The is intended to be flexible, allowing developers to create stacks of their own design with no strict ordering or composition of layers.

Fig. 2 presents a physical module as an example of how an SDOAM can accommodate heterogeneous protocols. The physical module is responsible for the Modulation and Coding Scheme (MCS) of the bit stream to be transmitted, and is responsible for switching between MCS solutions.

An important functionality of the physical module is to interface with commercial-off-the-shelf (COTS) underwater modems and provide integration for them alongside the software-defined MCSs. This is accomplished by abstraction drivers as pictured in the left-most module, `Driver`. The `Licensed MCS` depicted is a proprietary MCS binary from a private company that is capable of interfacing with available acoustic hardware. The `JANUS` block is an implementation of JANUS - an open, software-defined MCS proposed by the CMRE to be a commonly available MCS deployed in hardware-heterogeneous networks [15].

Logical link control mechanisms like packet retransmissions, when required, can also provided by the physical module. This is a functionality present in OSI's layer 2 and moved into CMRE's SDOAM's physical module as it is a feature often offered by commercial underwater modems.

*Policy Engine:* The Policy Engine (PE) is a wrapper and abstraction layer responsible for:

- Communicating with other modules;
- Switching between protocol modules inside the layer; and
- Handling of protocol parameters (*e.g.* managing dynamic transmission power).

*Cross-layer Framework:* The cross-layer framework is the centralised mechanism for information exchange between PEs. Every PE may request and provide a number of variables that are contextual in nature, and independent of the packet specifics (i.e., the payload). When providing a variable, the originator of the information also specifies a priority and maximum age. The cross-layer framework then associates a schedule to a variable instance when received, and computes an expiration time. The variable instance is then delivered to the receiving component based on the schedule, with the most urgent data being delivered first.

*1) SDOAM Operation:* Policy Engines do not process packets directly, but instead direct packets to an appropriate PM. The top level PE interfaces the SDOAM with the application layer. If a packet is being transmitted, a PE receives the packet from a PM in the layer above, and passes it to the PE below. In the simplest case there is a single managed protocol, and the PE is largely transparent.

The inverse process occurs when a packet is received. A packet is created at some receiving protocol, typically at a physical layer. After creation, the receiving PM passes the packet up to the managing PE. The PE then serves the packet to the upper module. Only the PMs that can handle the packet will be able to pass the resulting packet up to the managing PE.

The interface between a PE and its constituent PMs is simply defined, consisting of a "`receive()`" call to the library interface of each PM. A message to be transmitted is issued as a packet calling the `receive()` of the required PM. The PM processes the packet, and passes it to the next PE, which makes a determination about the next protocol destination for the particular packet instance, possibly based on configuration or context information from the cross-layer framework. This process continues until the packet is transmitted out of the SDOAM.

There are several key aspects to this design that are worth highlighting:

- Layers are delineated by a PE and a collection of modules, and are independent of the upper and lower layers.
- Successful packet handling from entry to exit points in a stack requires that the receiver have parity with the packets constructing modules on the transmitter.
- As a packet travels through a layer, it can only traverse through a single protocol.
- As a packet is composed or decomposed in a layer, the layer-specific information travels "horizontally" to or from the cross-layer framework, and the layer payload data moves vertically through the stack to the successive layer.

PEs and protocols that respect these guiding design principles preserve the concept of a stack and promote cross-layer intercommunication in a simple and flexible manner. A preliminary implementation has been developed at the CMRE and is completely independent of external, non-standard software sources.

## IV. Future Directions

In parallel to the development of the high-level architectural structure of the Software-Defined Open-Architecture Modem (SDOAM), there have been a number of coordinated efforts, carried out under various projects within the CMRE, that have generated components that can be inserted into an SDOAM implementation. This includes components in the physical, routing and transport contexts. Furthermore, CMRE is developing other modules, focusing on compression, security and covertness, localisation and non-acoustic modalities. These efforts have, as far as possible, been brought into alignment with the proposed SDOAM structure to facilitate their insertion into a practical Digital Communications Stack (DCS) compliant with the SDOAM design.

CMRE's intention with the SDOAM proposal is to reach out over an extended network of international interests, spanning research organisations, commercial manufacturers and users and catalyse consensus towards developing and adopting useful standards that serve the maritime community. The aspiration is that these developments may lead from an SDR-like architecture to a cognitive one.

## V. Conclusion

We have proposed a SDOAM that consists of an OSI-like stack, but with two important new features. Each layer of the traditional OSI stack is to be replaced by a wrapper, within which there are several modules, of equivalent function but that encode the various "languages" used in modems, presided over by a Policy Engine at each level that ensures a coherent choice of modules across all layers and consistency with application requirements. The second feature is that these Policy Engines and encapsulated modules have access to a generic cross-layer linking mechanism possibly via a common database, so that all manner of possible links can be created in a systematic, consistent manner. A third feature in the integration of information aggregators that provide navigation, security and time synchronization services.

It is CMRE's intention to move forward with a digital communications stack in 2014-2015 offering much of the proposed SDOAM capability and structured according to the architecture outlined.

## Acknowledgement

This work has been partially funded by the European Unions Seventh Framework Programme for research, technological development and demonstration under grant agreements no. 611449 and 288704.

## References

[1] F. Torre, "Speakeasy – a new direction in tactical communications for the 21st century," in *Tactical Communications Conference, 1992. Vol. 1 Tactical Communications: Technology in Transition., Proceedings of the*, Apr 1992, pp. 139–142 vol.1.

[2] J. Mitola, III, "Software radios: Survey, critical evaluation and future directions," *Aerospace and Electronic Systems Magazine, IEEE*, vol. 8, no. 4, pp. 25–36, 1993.

[3] ——, "SDR architecture refinement for JTRS," in *MILCOM 2000. 21st Century Military Communications Conference Proceedings*, vol. 1, 2000, pp. 214–218 vol.1.

[4] K. Richardson, C. Jimenez, and D. R. Stephens, "Evolution of the software communication architecture standard," in *Military Communications Conference, MILCOM 2009*. IEEE, 2009, pp. 1–8.

[5] C. M. Jimenez, K. W. Richardson, and D. R. Stephens, "SCA4 – an evolved framework," in *Military Communications Conference, MILCOM 2012*. IEEE, 2012, pp. 1–6.

[6] C. Petrioli, R. Petroccia, and D. Spaccini, "SUNSET version 2.0: enhanced framework for simulation, emulation and real-life testing of underwater wireless sensor networks," in *Proceedings of the Eighth ACM International Conference on UnderWater Networks and Systems*. ACM, 2013, p. 43.

[7] R. Masiero, S. Azad, F. Favaro, M. Petrani, G. Toso, F. Guerra, P. Casari, and M. Zorzi, "DESERT underwater: An NS-Miracle-based framework to design, simulate, emulate and realize test-beds for underwater network protocols," in *OCEANS, 2012 - Yeosu*, May 2012, pp. 1–10.

[8] M. Chitre, R. Bhatnagar, and W.-S. Soh, "UnetStack: an agent-based software stack and simulator for underwater networks," in *Proceedings of IEEE OCEANS14 St. Johns, Newfoundland*, September 2014, to be published.

[9] A. K. Kebkal, K. G. Kebkal, O. G. Kebkal, and M. Komar, "Modeling and experimental validation of basic characteristics of underwater acoustic communication based on signals with sweep-spread spectrum," presented at the IEEE UComms 2012, September 2012.

[10] G. Toso, R. Masiero, P. Casari, O. Kebkal, M. Komar, and M. Zorzi, "Field experiments for dynamic source routing: S2C EvoLogics modems run the SUN protocol using the DESERT underwater libraries," in *Oceans, 2012*, Oct 2012, pp. 1–10.

[11] Subnero Pte Ltd. Software-defined Subnero underwater modem. http://www.subnero.com/technology/.

[12] D. Torres, J. Friedman, T. Schmid, and M. B. Srivastava, "Software-defined underwater acoustic networking platform," in *Proceedings of the Fourth ACM International Workshop on UnderWater Networks*. ACM, 2009, p. 7.

[13] E. M. Sözer and M. Stojanovic, "Reconfigurable acoustic modem for underwater sensor networks," in *Proceedings of the 1st ACM international workshop on Underwater networks*. ACM, 2006, pp. 101–104.

[14] N. Nowsheen, C. Benson, and M. Frater, "A high data-rate, software-defined underwater acoustic modem," in *OCEANS 2010*, Sept 2010, pp. 1–5.

[15] NATO STO CMRE, "JANUS wiki," http://www.januswiki.org/.