**D2.11 - Specification of Protocols for ship-to-shore communication**

Project no. 636329

Project acronym: EfficienSea2

Project full title: EFFICIENSEA2 – efficient, safe and sustainable traffic at sea

Funding scheme: Innovation Action (IA)

Start date of project: 1 May 2015

End date of project: 30 April 2018

Duration: 36 months

Due date of deliverable: 25 April 2017

Actual submission date:

Revised submission date: -

Organisation in charge of deliverable: Partner 21, DANELEC

**DOCUMENT STATUS**

**Authors and contributors**

|  |  |
| --- | --- |
| Name | Organisation |
| Henrik Bech Helnæs (editor) | Danelec Marine |
| Erik Styhr Petersen | Wärtsilä |
| Anton Karelskiy | Transas |
| Timo Kostiainen | Furuno |
| Peter Andersen | Cobham |

**Document History**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Initials | Description |
| 0.01 | 2016-09-19 | HBH | First Draft |
| 0.02 | 2016-09-23 | HBH | Update after Hamburg meeting |
| 0.03 | 2016-09-17 | HBH | 50% review |
| 0.04 | 2017-03-03 | HBH | 75% review |
|  |  |  |  |
|  |  |  |  |

**Reviewers**

|  |  |
| --- | --- |
| Name | Organisation |
| Andy Winbow | CIRM |
| Krzysztof Bronk | NIT |
| Soeren Schweigert | OFFIS |
| Timo Kostiainen | FURUNO |
| Peter Andersen | COBHAM |
|  |  |

# 

Contents

[1 Summary 5](#_Toc476305838)

[2 Introduction 6](#_Toc476305839)

[3 Definitions and Acronyms 6](#_Toc476305840)

[4 Scope 7](#_Toc476305841)

[5 Logical Architecture and design 8](#_Toc476305842)

[5.1 Simple implementation 8](#_Toc476305843)

[5.2 Roaming function 10](#_Toc476305844)

[5.3 Cyber Security and IEC-61162-460 Firewall/Gateway 11](#_Toc476305845)

[5.3.1 HTTPS Proxy Server Requirements 12](#_Toc476305846)

[5.4 Helper Components 13](#_Toc476305847)

[5.4.1 Almanac 13](#_Toc476305848)

[5.4.2 Reaching on-board services from shore 15](#_Toc476305849)

[5.4.3 Transport of large amounts of data 18](#_Toc476305850)

[5.4.4 Large Data Transport Service 19](#_Toc476305851)

[5.4.5 Replication between Databases on-board and on-shore 20](#_Toc476305852)

[6 Interfaces 22](#_Toc476305853)

[6.1 REST 23](#_Toc476305854)

[6.2 SOAP 23](#_Toc476305855)

[6.3 Conclusion 23](#_Toc476305856)

[7 Non IP Media Gateways 25](#_Toc476305857)

[7.1 MMS Helper Component using non IP Media Gateway 27](#_Toc476305858)

[7.2 IP Traffic over VDES 28](#_Toc476305859)

[8 Efficient Protocols vessel/shore 29](#_Toc476305860)

[8.1 TCP and FTP over high latency, low bandwidth communication channels. 29](#_Toc476305861)

[8.2 TLS over high latency, low bandwidth communication channels (hereunder HTTPS). 29](#_Toc476305862)

[8.3 UDP over high latency, low bandwidth communication channels. 30](#_Toc476305863)

[8.4 Privacy v.s. compression 30](#_Toc476305864)

[8.5 Vessel Public or Local IP Address and IP v4 v.s. IP v6 30](#_Toc476305865)

[8.6 IP v4 v.s. IP v6 31](#_Toc476305866)

[9 Maritime Cloud Demonstrator Component 32](#_Toc476305867)

[10 Deployment considerations and Architecture Update 33](#_Toc476305868)

[11 Proposed Demonstrator 34](#_Toc476305869)

[12 Identification of potential Areas for standardization 35](#_Toc476305870)

[13 Conclusion 36](#_Toc476305871)

[Bibliography 37](#_Toc476305872)

[14 Appendix A –Review Report 38](#_Toc476305873)

# Summary

<TBD: Summary of report>

# Introduction

This document is a deliverable produced by WP2 Task 2.4 – shipboard system integration and on-board networks.

In short, the Task 2.4 is about shipboard system integration in terms of cyber security, on-board network integration and firewalling as well as integration of systems with the shipboard component of the maritime cloud concept.

This document is to specify protocols to be used for vessel-to-shore communication for the different types of Equipment on the vessel.

<TBD: More introduction when rest of chapters has more contents>

# Definitions and Acronyms

|  |  |
| --- | --- |
| AE | Architectural Element |
| AIS | Automatic Identification System (IEC 62320:2008) |
| ASM | Application Specific Messaging |
| DMZ | Demilitarised Zone |
| DOS | Denial Of Service |
| E2 | EfficienSea2 |
| ECDIS | Electronic Chart Display and Information System (IEC 61174:2015) |
| FW | Firewall |
| GPS | Global Positioning System |
| GW | Gateway |
| ICS | Integrated Communication System |
| LOS | Loss of Service |
| MC | Maritime Cloud |
| MCC | Maritime Cloud Client Component |
| MMS | Maritime Messaging Service |
| MSP | Maritime Service Portfolio |
| QoS | Quality of Service |
| RADAR | RAdio Detection And Ranging (IEC 62388:2013) |
| SOLAS | Safety of Life at Sea (SOLAS Convention, 1974, with amendments) |
| VDE | VHF Data Exchange |
| VDES | VHF Data Exchange System |
| VDR | Voyage Data Recorder (IEC 61996:2013) |
| VTS | Vessel Traffic Service |
| Wi-Fi | a trademark of the Wi-Fi alliance (WLAN or Wireless Local Area Network) |

# Scope

The deliverable D2.11 as described in the E2 consortium agreement is to provide specification of protocols for vessel-to shore communication for the different type of equipment on the vessel.

Since other Work Packages in E2 is developing service architecture and some service descriptions, including interface specifications, the scope of the T2.4 D2.11 has been changed to include a more elaborate on-board logical architecture/design with discussion of various types of helper components and what services they could provide for on-board MC clients.

Work done in WP2 and WP3 has shown that there is a need to elaborate more on the on-board architecture and that the definition of the MC, in particular with respect to the Maritime Cloud Demonstrator Component needed to be refined.

<TBD: more scoping rationale>

The D2.11 scope is thus:

* Develop and describe the on-board logical architecture/design
* Describe requirements for standard interface to the MC Identity and Service Registry
* Describe requirements for standard interface to the T2.3 Roaming Device
* Discuss and recommend efficient communication protocols (in IP domain) to be used between clients and services in the MC
* Review and update the T2.4 D2.10 Recommended on-board network architecture
* Describe proposed demonstrator implementation D2.12 to be delivered in M36

# Logical Architecture and design

This chapter describes the proposed logical architecture and top level design on the components of the maritime cloud with focus on the on-board components and how they interface/communicate to/from shore components of the maritime cloud.

The various diagrams show relations on the transport level only and hence the components and communication done to enable establishment of transport connections on lower OSI layers are not shown.

In summary, the proposed logical architecture and design, will only allow transport layer endpoints. E.g. implementation of messaging via VDES in AIS, NAVTEX similar way must be done so it adheres to the OSI model and that endpoints in this type of messaging are always on transport level.

## Simple implementation

The simplest implementation of a client of the maritime cloud using a service that has been discovered using Service Registry and authenticating using the identity registry is shown in Figure 1.



Figure 1 Simplest implementation of a maritime cloud client

Note that the diagram does not implement any cyber security using firewall, nor does it contain a roaming device. A roaming device does NOT need to be active or existing in a communication between client and a service.

The following chapters will add roaming device and firewall to the architecture.

The diagram depicts the simplest use of the MC as described in (E2-T3.1-D3.2, 2015).

Also note that the diagram also defines that protocol and interface to the IR and SR are HTTPS and SOAP. The rationale for that is described in chapter 0.

Since the Conceptual model of maritime cloud does not specify how particular services are implemented – i.e. it does not require for example HTTPS and SOAP, the protocol and interface used to communicate with the service could be any IP based.

## Roaming function

To accommodate for seamless roaming of traffic between various communication channels, Figure 2 illustrates that component as well as how a client can use the roaming components interface to setup Quality of Service for the communication that it needs to make to services/ components on shore.

The Roaming device protocol and interface to the Maritime Cloud i.e. the on-board clients are required to be SOAP using HTTPS as application layer protocol. The rationale is described in chapter 0.

The EfficienSea2 deliverable (E2-T2.3-D2.8, 2016) D2.8 Interface to Maritime Cloud, amongst other, specify that interface.



Figure 2 Roaming Device to manage multiple communication Channels and QoS

Apart from the Interface to setup QoS for communication, the Roaming device acts transparently to any IP based communication to/from the vessel. This includes the communication with IR and SR.

## Cyber Security and IEC-61162-460 Firewall/Gateway

The Cyber Security considerations in (E2-T2.4-D2.10, 2016) D2.10 Recommended On-Board Network Architecture, found that use of IEC62216-460 Firewall/Gateways to ensure Cyber Secure protection of network zones was needed. Since clients often reside in a secure zone, Figure 3 shows how such a Firewall/Gateway can be applied.

Clients that are cyber secure can of course exist on the unprotected side.



Figure 3 Client Communication thru -460 Firewall

The -460 firewall is a dual firewall, partly due to the reason to deny direct IP based connections thru the firewall. This is done so that same IP ports are not open on inside FW and outside FW. The -460 standard are very specific on this topic when it comes to e.g. VPN connections thru the firewall – which would violate above IP port opening restriction. -460 then describes that such VPN connections e.g. should be time limited.

On top of the basic Packet Filtering function of a dual firewall, it traditionally also contains Application Gateways that applies security mechanisms to specific applications such as FTP and Telnet servers.

Further, it contains Circuit-level gateways that applies security for TCP or UDP connections that is established. Circuit-level gateways also include state-full inspection functionality that tracks the state of each connection and its protocol, and only allows “legal” state transitions.

The latest in traditional firewalls are threat management that also includes abilities to apply filters for malware and virus as well as detection and blocking of denial of service attacks.

Next-generation firewalls or, one should rather say present day firewalls also include:

* Integrated intrusion prevention
* Application awareness and control to see and block risky apps
* Upgrade paths to include future information feeds
* Techniques to address evolving security threats

Two of the basic techniques in a firewall are network address translation (NAT) and proxy servers.

The following will describe the requirements for a HTTPS proxy server, in a -460 Gateway.

### HTTPS Proxy Server Requirements

A proxy can keep the internal network structure of a network secret by using network address translation, which can help the security of the internal network. This makes requests from machines and users on the local network anonymous.

There are several types of proxy servers.

* A proxy server that passes requests and responses unmodified is usually called a gateway or sometimes a tunneling proxy.
* A forward proxy is an internet facing proxy used to retrieve from a wide range of sources.
* A reverse proxy uses a front-end to control and protect access to a server on a private network. A reverse proxy amongst others takes care of authentication and en/decryption.

For the purpose of vessel to shore initiated communication, a reverse proxy is of little use. A basic network address translating gateway is sufficient. Application of a reverse proxy that terminates HTTPS sessions, just to forward the payload into other established HTTPS sessions to shore would be a security risk and not needed.

The network address translation is however needed to obtain cyber security.

For the purpose of shore to vessel initiated communication, where services on vessel are to be contacted by shore based clients, the reverse proxy type in combination with VPN tunnelling will be required, as described later in this document.

## Helper Components

This chapter discuss various possible helper components that might be beneficial to have in the architecture for various reasons.

One reason could be that if clients would need to implement the same functionality and that functionality would be so significant that a shared component implementing the functionality would make sense.

Another reason could be that if clients would need to communicate the same information across the sparse communication links to shore, it would make sense that clients share a component that communicate and hold that information.

### Almanac

One helper component that has been described in Pre-EfficienSea2 work and in the (E2-T3.1-D3.2, 2015) MC Conceptual Model is the Almanac.

The almanac could be as simple as an off-shore version of the IR and the SR, but it could also offer a more sophisticated service discovery and authentication methods.

Almanac could also expand set of services offered via shore based SR with on-board services (or local services).

Almanac could also implement a service availability policy and hence require clients to be restricted to that policy on a vessel or a fleet of vessels.

It should be noted here that clients are not required to use the Almanac, and could still perfectly well communicate directly with the shore based IR and SR.

In Figure 4 the recommended placement in the logical architecture is shown. Please note that in this example, the Client only uses the Almanac to get IR and SR information. It could as well communicate directly with the IR and SR on shore.



Figure 4 Almanac to simplify Clients and minimize communication

Since a protocol/interface to the almanac for the clients will be shared in a similar fashion as IR and SR, the protocol/interface needs to be standardized. Hence requirement for HTTPS and SOAP (ref. chapter 0)

### Reaching on-board services from shore

For majority of installations with IP communication over SATCOM and even 3g/4g, the vessel and services on the vessel does not have an endpoint reachable from shore or other vessels.

Whenever such a communication scenario is needed, non-standardized and temporary VPN solutions are applied. One example is remote management of on-board equipment for maintenance or monitoring purposes.

The IEC61162-460 standard does not allow permanent VPN connections thru the -460 Gateway/Firewall. To overcome that limitation in a cyber secure way, a helper component could be a VPN Service e.g. in the simplest form of a VPN Router in the -460 DMZ as shown in Figure 5.



Figure 5 Reaching an on-board Service from shore

Since the VPN service component is in the DMZ, it would be allowed to create a permanent VPN tunnel to a shore based entity. It could then allow for shore initiated transport sessions to on-board services with a proper cyber secure authentication as shown using the shore based “Vessel Connector”.

With the VPN Service setup in Figure 5, the Vessel Connecter need to be sufficiently Cyber Secure, since access to a secure zone on the vessel is provided, even though the in-bound Firewall only provides access to the on-board Service to be reachable from shore. Alternatively a higher level of cyber security can be achieved by adding a reverse proxy server in the DMZ so that shore-based clients is forced to authenticate with this server, before the on-board service can be used.

Figure 16 Show how a VPN Service can be implemented securely by combining a VPN Client with a Reverse HTTPS Proxy Server.

Initially, the VPN Client establish a VPN Tunnel to the Vessel Connector, allowing shore based clients to initiate connections to the on-board services. The VPN Client may also want to establish appropriate Quality of service for the VPN connection by setting this up with the Roaming Device.

The Reverse Proxy may requires authentication of incoming connections before allowing traffic to on-board services. This, however depends on security level of the zone that Vessel Connector and the shore based client is in.

<TBD: authentication using IR and SR)



Figure 6 VPN Service Implementation Example

The Sea Traffic Management (STM) validation project (STM Communications, 2017) has defined a component similar to the Vessel Connector and has named it SeaSwim Connector. At time of writing, the specification of the SeaSwim Connector was not yet finalized, but here a reference to the draft specification (Fabio Renda, 2016).

#### Recommended VPN standard

A virtual private network (VPN) extends a private network across a public network, such as the Internet. It enables users to send and receive data across shared or public networks as if their computing devices were directly connected to the private network. Applications running across the VPN may therefore benefit from the functionality, security, and management of the private network.

A VPN is created by establishing a virtual point-to-point connection through the use of dedicated connections, virtual tunneling protocols, or traffic encryption.

VPN systems may be classified by:

* The protocols used to tunnel the traffic
* The tunnel's termination point location
* The type of topology of connections, such as site-to-site or network-to-network
* The levels of security provided
* The OSI layer they present to the connecting network, such as Layer 2 circuits or Layer 3 network connectivity
* The number of simultaneous connections

There exist many variants of VPN on both the Layer2 and Layer3 types.

Many router manufacturers, including Asus, Cisco, Draytek, Linksys, Netgear, and Yamaha, supply routers with built-in VPN clients. Some use open-source firmware such as DD-WRT, [OpenWRT](https://en.wikipedia.org/wiki/OpenWRT) and Tomato, in order to support additional protocols such as [OpenVPN](https://en.wikipedia.org/wiki/OpenVPN).

The requirement as set out by the scope of EfficienSea2 is to provide standardized and preferably open-source to enable easy implementation, integration with existing IT infrastructures and wide acceptance of the MC. Further the VPN standard must be able to provide the right levels of cyber security.

Only OpenVPN fulfills these requirements.

OpenVPN is an open-source software application that implements virtual private network (VPN) techniques for creating secure point-to-point or site-to-site connections in routed or bridged configurations and remote access facilities. It uses a custom security protocol that utilizes SSL/TLS for key exchange. It is capable of traversing network address translators (NATs) and firewalls. It was written by James Yonan and is published under the GNU General Public License (GPL).

### Transport of large amounts of data

<TBD: Describe at least one solution using helper component(s) for transport of large amounts of data>

Many Satcom providers are offering services for efficient transfer of larger data amounts.

#### Data mirror services

Some of the services are offered as a file repository mirror service.

A file server is located on-board the vessel with standardized SMB or NAS interface accessible by clients on the vessel. Similarly a file server is located on shore with similar standardized SMB or NAS interfaces towards clients on shore. The two file repositories are synchronized, so that any changed content on either side is synchronized to the remote side. The synchronization service is proprietary and highly optimized to reduce sat-com bandwidth usage.

The mirror services may offer multiple shore instances. E.g. the file repository on-board the vessel may contain separate folders that gets synchronized with different shore file servers.

The synchronization traffic is of course encrypted and the client use standardized authorization with the file servers to gain access to the synchronized content.

The advantage of this type of service is that control of QoS is quite straight forward and can be given lower priority so that large data transfer can occur in background in a similar manner as e-mail traffic.

#### FTP Proxy servers

Other services are offered using instances of FTP proxy servers.

The FTP proxy servers terminate the FTP connections from the clients, but are transporting the FTP data and commands in an optimized manner to the shore based FTP proxy counterparts.

The traffic between two FTP proxy servers are then highly optimized to reduce sat-com bandwidth and introduce higher reliability for the FTP connections, given the latency and dropout characteristics of sat-com connections.

The advantage of this latter service is that it has a standardized FTP interface and is able to transport media to any shore based ftp server.

The disadvantage is that using plain FTP provides no privacy and have cyber security risks.

If the service is implemented as FTP over SSH (SFTP), the proxy servers involved would need to terminate the SSH sessions locally if any optimizations providing higher efficiency than plain SFTP is to be done.

### Large Data Transport Service

This chapter describe and demonstrate how a Large Data Transport Service, dedicated to move large amounts of data to/from shore efficiently, can be constructed.

The idea with the service is that large data amounts can be posted for transfer and that the transfer can be executed asynchronously to the client / service communication (A service similar to an UPS service, just for digital payloads).

This type of service could be centralized and standardised.

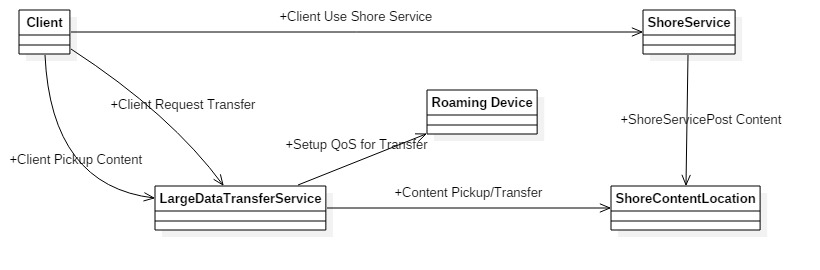


Figure 7 Classes and Associations for a Large Data Transport Service

Figure 7 illustrates a Client that is in the process of using a shore service (assuming IR and SR already used to establish the association). As result of the Clients use of the shore service is that a large data amount has to be delivered to the client. The Shore Service provides information to the Client on where the data content is located (URL) alongside with metrics and credentials to use to get access to the data. The client then use a local on-board service (Large Data Transfer Service) that is specialized in handling transfer of data to/from shore efficiently.

The Transfer Service provides the client with estimated time of arrival of the large data and transfer session identification/handle for later reference.

The Large Data Transfer Service is of course able to setup the Quality of Service required for the particular Transfer by using the Roaming Device Configuration Interface.

Figure 8 show a sequence diagram of the communication involved in an asynchronous transfer of large data content. Note that the diagram does not include details on service discovery, authentication etc. These details can be found in (E2-T3.3-D3.7, 2016) Technical Specification of the Maritime Cloud. Also it does not describe any particulars of the Shore Service. One can imagine that the Shore Service could be a sea chart provider and the sea charts then constitute the large data to transfer.

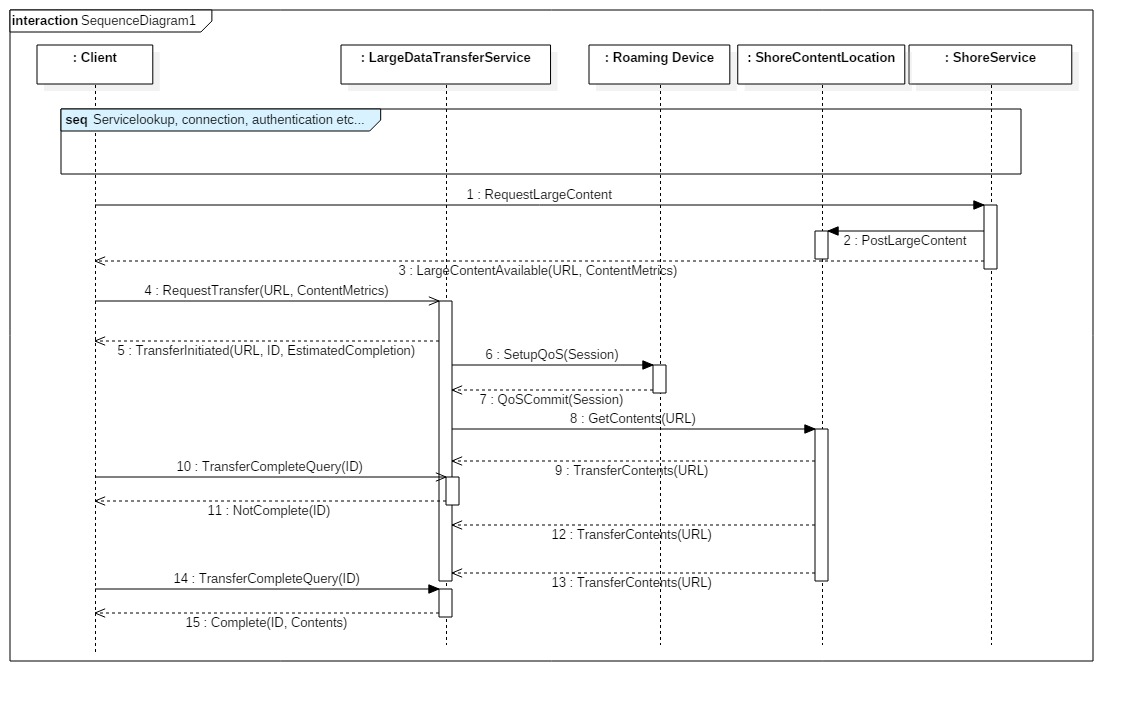


Figure 8 Sequence Diagram for the communication involved for a Large Data Transfer Job

A large data transfer service as shown in Figure 7 could very well be implemented with available open source based SFTP client and servers with additional simple service applications.

### Replication between Databases on-board and on-shore

A use-case derived from work in E2 Task 5.1 and 2 Development of a new common port database concept and structure (E2-T5.1-D5.3, 2016) and identified in liaison work between WP5 and WP2 T2.4 is the need for replication of databases on-board and on-shore.

It is imagined that a general structure of databases containing information needed to generate a large range of documents to be submitted in relation to port reporting as well as information needed to operate and plan vessels port of call efficiently.

Much of the information is confidential and owned by the vessel owner/operator and other information is publicly available.

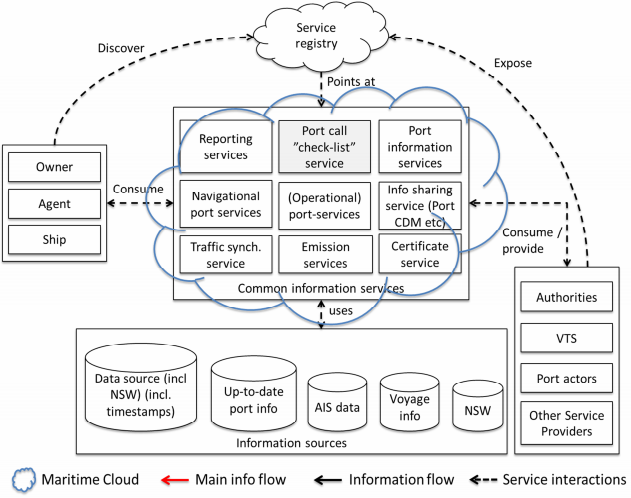


Figure 9 An architecture as illustrated in E2 D5.3

In relation to ship/shore communication and protocols, the Box containing the Owner, Agent and Ship becomes the focus.

In discussion with WP5, the owner, agent, ship architecture could be constructed as a set of replicated databases containing partly confidential information and partly information available from the various services/sources in the maritime cloud.

To discover the basic communication and protocol need, a setup with one shore based DB in owner secure zone, replicating with vessel databases, also in owner controlled secure zones as shown in Figure 10



Figure 10 Vessel DataBase synchronisation with shore DB

Question is then, is the suggested on-board architecture and design able to support the use-case and what recommendations can be deduced for efficient protocols supporting the synchronisation.

The size of data cannot be characterized as large and the requirement for speed of migration of data updates are low (latencies in the 10min range are accepted).

Database replication is a topic that have been researched heavily and still are. There are many solutions on the market both as open source and proprietary. There are different architectures, each providing different benefits.

Various replication methods is implemented in database replication middleware. Each method is using different protocols to exchange information.

All the replication methods seen so far is based on transport level communication (TCP), either based on TLS or using VPN tunnelling. The format used to exchange data or replication records/statements relies completely on the specific replication middleware.

The proposed architecture/design from T2.4 is considered open and ready to use for implementation of replication middleware, tailored to the needs and the available communication media.

With the architecture and active use of the roaming device, it should be considered that replication middleware can utilize the knowledge of available communication channels to make intelligent decisions on when and what data to replicate.

# Interfaces

This chapter describe requirements to the interface provided by the generic services of the MC, namely the Identity Registry, the Service Registry, the on-board Roaming Device and the various on-board helper components.

The E2 WP3 have had early discussion on the pro’s and con’s of various choices and there have been a strategic decision on HTTP/HTTPS based communication where choices landed between SOAP and REST.

## REST

Representational state transfer (REST) or RESTful web services are one way of providing interoperability between computer systems on the internet. REST-compliant web services allow requesting systems to access and manipulate textual representations of web resources using a uniform and predefined set of stateless operations.

By making use of a stateless protocol and standard operations REST systems aim for fast performance, reliability, and the ability to grow, by using reusable components that can be managed and updated without affecting the system as a whole, even while it is running.

REST is an architectural form, and is not an international standard. Interface/implementation has to live up to certain criteria’s before it can be called RESTfull.

## SOAP

SOAP (Simple Object Access Protocol) is a protocol specification for exchanging structured information in the implementation of web services in computer networks. Its purpose is to induce extensibility, neutrality and independence. It uses XML Information Set for its message format, and relies on application layer protocols, most often Hypertext Transfer Protocol (HTTP) or Simple Mail Transfer Protocol (SMTP), for message negotiation and transmission.

SOAP allows processes running on disparate operating systems (such as Windows and Linux) to communicate using Extensible Markup Language (XML). Since Web protocols like HTTP are installed and running on all Operating systems, SOAP allows clients to invoke web services and receive responses independent of language and platforms.

SOAP is a W3C Recommendation and as such, precise and standardized.

## Conclusion

Since one of the goals of E2 is to deliver inputs to standardization, and it is obvious that the interfaces to generic entities of the MC are subject for standardization. Hence it is recommended that the following interfaces must be specified using the SOAP standard:

* Interface to Identity Registry
* Interface to Service Registry
* Interface to Roaming Device
* Interfaces to generic on-board Helper Components

This conclusion does not prevent the existence of MC services that are REST based only. In fact, it is believed that a majority of services developed, will be REST based initially. As services mature and some move into navigation and safety related services, and become standardized, it is believed that these will be required to provide SOAP based interfaces.

<TBD: Describe requirements to interfaces

* Roaming Device QoS
* IR
* SR
* ….

….>

# Non IP Media Gateways

In the E2 project, there has been quite some debate on end-points and discussions to solve the problem of clients wanting to both communicate with IP based services as well as over non IP radio links, such as VDES, AIS/ASM. Especially the worries have been how to integrate the legacy NAVTEX and AIS with the IP domain of the maritime cloud.

The outcome has led to confusion and mixing layers of the OSI model as well as giving a hard time defining the parameters when specifying Quality Of Service in the Roaming Device interface to MC.

This chapter will attempt to provide a solution to this problem.

Until now, the list of possible communication endpoints as defined in D2.8 (E2-T2.3-D2.8, 2016) is a mix between endpoint types in the IP domain and endpoints in the non IP domain.

This means that the client no-matter what, has to distinguish between endpoint types and act differently depending on the nature of the communication media.

The mesh of nodes in the IP domain (the network and the internet), with forwarders and routes, always ensures that traffic gets routed through from one endpoint to another, no matter what type of lower layer OSI model communication media is applied. (Copper, Wireless, fibre, … ).

The routing function in the network is provided by the router nodes in the network.

When dealing with non IP communication links, such as what is used for the VHF AIS/ASM, there is no routing function available. The requirement for communication is that endpoints are within VHF range. The VHF AIS/ASM is also a broadcast media. This means that all radios within range of the transmitter, will receive the communication. VHF AIS endpoints do not relay/transpond messages, and hence it is not directly possible to build a VHF AIS/ASM network/mesh of nodes like nodes in the MC where traffic/messages are routed.

This means that for a client that wants to send/receive messages via VHF AIS/ASM, cannot communicate with VHF AIS/ASM endpoints that are not in VHF Range.

This leads to the conclusion that clients that wants to send messages via VHF AIS/ASM does not need to see other endpoints than VHF AIS/ASM and are aware that this is a very different endpoint than an IP based MC Service. If the Client wants to send a message to another vessel via the VHF AIS/ASM media, it cannot be guaranteed delivery.

The above leads to the idea to structure the on-board architecture, so that the roaming device does not need to roam anything else than IP based traffic. This will help avoiding the definition of the hybrid that is on the drawing board right now, lead to a simpler definition of endpoints and avoid the mixing of QoS from different OSI layers.

Figure 11 shows the first step in the idea. An AIS/ASM Gateway component is introduced. The Gateway is providing a HTTPS, SOAP based interface/service for clients on-board the vessel. The service offers means to transmit AIS/ASM messages as well as means for the client to listen to AIS/ASM messages. The client can setup the Roaming Device so that the communication with the AIS/ASM Gateway, gets the right QoS.

The prioritisation/QoS of messages sent thru the AIS/ASM Gateway are not the responsibility of the Roaming Component, but is managed by the AIS/ASM Gateway.

This then leads to the fact that no matter what, the client both has to specify the QoS for the communication between Client and Gateway AND the QoS for traffic thru the AIS/ASM Gateway.



Figure 11 Client using an AIS/ASM Gateway

This also illustrates that the current specification of the roaming device, are trying to integrate two different functions in a hybrid manner that leads to confusion.

Figure 11 also illustrates that the AIS/ASM Gateway, would then be a local service offered on-board the vessel, and hence it would make no sense that the shore based IR and SR should have anything to do with the clients use of the AIS/ASM Gateway service. The immediate solution to that is the Almanac helper component (chapter 5.4.1), that can provide information on local vessel services.

Figure 12 summarises and includes local on-board non IP Media Gateway Services in to the Architecture/Design. The hybrid client shown are using the Almanac to search for services and then use shore based MC services as well as local non IP Gateway services.

The figure also show with blue dashed lines, what is being discussed right now in E2 as being the Roaming device, where the red dashed line area could as well be a more logic and structured approach.



Figure 12 Architecture/Design including non IP Media Gateway

## MMS Helper Component using non IP Media Gateway

Most of the motivation for including the legacy non IP domains is to support messaging as for example done with ASM. A client of the MMS system should see this service of the Maritime Cloud as being seamless of communication media. Building upon the architecture in Figure 12, the hybrid client could grow into the on-board MMS helper component as discussed many times in E2. Figure 13 show how an MMS Helper component could encapsulate the IP and the non IP domain messaging and create one seamless interface for an on-board MMS Client.

Note that MMS implementation might not make use of Almanac, but use shore based Identity Registry and Service Registry directly.



Figure 13 MMS Helper Component

## IP Traffic over VDES

<TBD: IP based traffic over VDES>

VDES offers a low bandwidth data communication channel between VDES nodes within VHF range. From VDES draft standard, the estimated bandwidth is 150kbps.

It is possible transmit/receive IP data packets over VDES.

TBD: requirement for public IP addresses

TBD: routing function in VDES nodes

TBD: traditional IP routing ARP, RIP, DNS and related overhead, mobility equals larger overhead

TBD: source routing

# Efficient Protocols vessel/shore

<TBD: A general discussion on protocols used in client/service communication vessel/shore

All internet communication is based on (RFC1122, 1989).

## TCP and FTP over high latency, low bandwidth communication channels.

The Transport Connection Protocol is the most widely used transport protocol on the internet. It forms the basis for application protocols, such as HTTP, FTP, …

TCP is accompanied by the Transport Layer Security Protocol (former “Secure Socket Layer” SSL).

When using TCP over SATCOM communication channels offer some problems due to the higher latency and higher error rate compared to high data rate land connections.

Many works and even modification of TCP have been suggested to solve these problems.

Two works are referenced here: (Katz, 1999) “TCP Performance over Satellite Channels” and (Latour, 2000) “Enhancing TCP Performance over Satellite Channels”.

Both works conclude with recommendations on TCP configuration parameters as well as introduction of TCP spoofing methods.

When using FTP for transporting larger files over communication channels with high latency and even have dropouts (packet loss), transport may be disconnected from time to time. It is recommended to utilize the ability to “resume” a file transfer in the FTP protocol, to avoid losing data already transferred.

## TLS over high latency, low bandwidth communication channels (hereunder HTTPS).

Since TLS is an “extension” of TCP, the considerations stated in chapter 8.1 are also valid. It is much more bandwidth consuming to establish a TLS connection than a plain TCP connection. The overhead depends highly of the PKI methods used for the particular TLS session. In many applications it should be considered to keep a TLS session open for longer periods where there is no data transferred between the client and service, since it may require less bandwidth to keep the session open than to tear it down and reopen later.

## UDP over high latency, low bandwidth communication channels.

Universal Datagram Protocol (UDP) can be used to transfer single datagram messages to/from shore. The protocol only guarantees that content in a single datagram is intact. The benefit of UDP is that it has a smaller overhead.

UDP is used heavily in the IEC61162-450 based navigation/vessel network to propagate NMEA sentence, automation system data as well as radar and ecdis imaging.

UDP may be used for streaming applications where data loss is acceptable.

UDP has many benefits in a multiple talker and multiple listener architecture.

Usage of UDP over high latency, low bandwidth communication channels may be considered for very particular applications such live data streaming in special situations.

## Privacy v.s. compression

To obtain privacy, the data to be transferred must be encrypted. To minimize bandwidth usage, the data must be compressed.

Encryption turns your data into high-entropy data, usually indistinguishable from a random stream. Compression relies on patterns in order to gain any size reduction. Since encryption destroys such patterns, the compression algorithm would be unable to give you much (if any) reduction in size if you apply it to encrypted data.

In other words, for any type of compression protocol known today, the efficiency is lowered very significantly if the data is encrypted before the compression. This is due to the nature of encryption, that is practically outputting a scrambled data stream with no data segment repetitions, a property that makes compression inefficient.

Therefore to be able to transfer data efficiently, the compression must be done before data is encrypted.

To obtain complete end-to-end privacy and avoid man in the middle attacks, the encryption must be done at source client and decryption at destination client. To obtain low bandwidth usage, this then requires that compression and de-compression are also done at the source and destination clients respectively.

TLS standard includes methods that support compression before encryption, see .

## Vessel Public or Local IP Address and IP v4 v.s. IP v6

The current infrastructure of various communication providers such as SATCOM, WLAN and 3,4G providers varies. Since the aim of the T2.3 Roaming device is to provide seamless connection from Vessel to Internet, the roaming between the different providers, need to take into account the fact that the vessel communication entry/exit node may be assigned different IP addresses dynamically.

If no action is taken, an established transport session from the vessel to shore cannot be kept alive if roaming device changes to a media where vessel node gets a different IP address than the one used when the transport session was established.



Figure 14 Roaming and IP Address scenarios

The only solution to this problem is to allocate a public IP address for the vessel/roaming device to announce. In the suggested architecture, it is the roaming device that should implement the node having this public IP address.

## IP v4 v.s. IP v6

The amount of available IPv4 addresses is rapidly getting smaller and everything internet is in the process of migrating to IPv6

It is yet to be investigated if IP v6 addressing is possible in relation to SATCOM.

<TBD>

# Maritime Cloud Demonstrator Component

In previous E2 deliverables, such as (E2-T3.1-D3.2, 2015), a Maritime Cloud Client Component (MCC) has been defined. In(E2-T3.3-D3.7, 2016), this component is being further defined and renamed to Maritime Cloud Demonstrator Component.

Quote: “The Maritime Cloud Demonstrator Component serves as a reference implementation on how to use the other Maritime Cloud core components and can further be used by application developer (for example provider of ship equipment) to ease the usage of MC functionality. For this purpose the MCDC realizes some convenience functions, and takes the Roaming Device into account, to ensure a maximum of connectivity. “

There are several purposes to have the MCDC as an architectural element. One is to have a reference implementation of a MC client as a deliverable in E2. Another reason is that the MCDC in an on-board architecture may ease the implementation of MC clients by having the MCDC implementing components/functionality that can be reused amongst clients. The (E2-T3.1-D3.2, 2015) lists two of the obvious areas:

* Secure authentication, signing and encryption support through online use of the Maritime Identity Registry or means to provide the same services offline.
* Service discovery through online use of the Maritime Service Registry or means to provide the same offline.



Figure 15 EfficienSea2 Entities

In Figure 15, the blue dashed boxes identify proposed entities for use in EfficienSea2. Note that the MCDC by this is the collection of helper components identified in previous chapters.

# Deployment considerations and Architecture Update

This chapter reviews the on-board network architecture as described in (E2-T2.4-D2.10, 2016) and discuss considerations on deployment.

The logical architecture/design developed in the previous chapters, does not require existence of neither the MCDC (helper components), nor the roaming device.

Nor does it put restrictions on the number of instances of MCDC and 460 gateways.

This is considered an advantage of the architecture/design, since it allows multiple vendors to build integrated systems with propriety implementations of MCDC and -460 Gateway functionality.

Further, the logical architecture/design allows multiple Roaming Device instances, each tailored for specific needs. One could imagine a roaming device specifically allocated for Navigation related communication and one allocated for engine systems communication.

An example deployment with two separate Vessel Systems (with separate security zones) are shown in Figure 16



Figure 16 Example Network Architecture

# Proposed Demonstrator

<TBD: Propose Prototype demonstration of vessel-to-shore communication roaming.

The M36 deliverable for T2.4 is an Implementation of a prototype which demonstrates vessel-to-shore communication for at least two different types of equipment using the

Protocols defined in D2.4.2 >



# Identification of potential Areas for standardization

<TBD: This chapter is included to provide input to E2 WP1.>

# Conclusion

# Bibliography

**E2-T2.3-D2.8. 2016.** *D2.8 Interface to Maritime Cloud.* s.l. : EfficienSea2, 2016.

**E2-T2.4-D2.10. 2016.** *D2.10 Recommended On-board Network Architecgture.* s.l. : EfficienSea2, 2016.

**E2-T3.1-D3.2. 2015.** *D3.2 Conceptual Model.* s.l. : EfficienSea2, 2015.

**E2-T3.3-D3.7. 2016.** *D3.7 Technical Specification.* s.l. : EfficienSea2, 2016.

**—. 2016.** *Technical Specification of the Maritime Cloud.* s.l. : EfficienSea2, 2016.

**E2-T5.1-D5.3. 2016.** *Development of a new common port database concept and structure.* s.l. : EfficienSea2, 2016.

**Fabio Renda, CIMNE. 2016.** *SeaSWIM Connector Service.* s.l. : STM, 2016.

**Katz, Thomas R. Henderson and Randy H. 1999.** *TCP Performance over Satellite Channels.* Berkeley : EECS, University of Calilfornia, 1999. 94720.

**Latour, Maik Kammermann and Capt Hugues. 2000.** *Enhancing TCP Performance Over Satellite Channels.* Ottawa : Defence Research Establishement Ottawa, 2000. DREO TR 2000-079.

**RFC1122. 1989.** *Requirements for Internet Hosts -- Communication Layers.* s.l. : IETF, 1989.

**RFC3749. 2004.** *Transport Layer Security Protocol Compression Methods.* s.l. : IETF, 2004.

**RFC5246. 2008.** *The Transport Layer Security (TLS) Protocol v1.2.* s.l. : IETF, 2008.

**STM Communications, STM Validation. 2017.** Sea Traffic Management. *http://stmvalidation.eu/.* [Online] 14 02 2017. [Cited: 14 02 2017.] http://stmvalidation.eu/.

# Appendix A –Review Report

| **N°** | **Reviewer**  **Initials** | **Reference in document (General or Paragraph, Figure …)** | **Type (editorial, structural, formulation, error)** | | **Reviewer's Comments, Question and Proposals** | | **Editor’s action on review comment.** |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | |  | |  |
|  |  |  |  | |  | |  |
|  |  |  |  | |  | |  |
|  |  |  |  |  | |  | |
|  |  |  |  |  | |  | |
|  |  |  |  | |  | |  |
|  |  |  |  | |  | |  |
|  |  |  |  | |  | |  |
|  |  |  |  | |  | |  |
|  |  |  |  | |  | |  |
|  |  |  |  | |  | |  |