



BSI Standards Publication

Ships and marine technology — Publish-subscribe architecture on ship-shore data communication — General requirements

National foreword

This British Standard is the UK implementation of ISO 18131:2025.

The UK participation in its preparation was entrusted to Technical Committee SME/32, Ships and marine technology - Steering committee.

A list of organizations represented on this committee can be obtained on request to its committee manager.

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

© The British Standards Institution 2025
Published by BSI Standards Limited 2025

ISBN 978 0 539 24614 8

ICS 47.020.70

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 30 November 2025.

Amendments/corrigenda issued since publication

Date	Text affected
------	---------------



**International
Standard**

ISO 18131

**Ships and marine technology —
Publish-subscribe architecture on
ship-shore data communication —
General requirements**

Navires et technologie maritime — Architecture Publish-subscribe (publier/s'abonner) pour la communication de données entre les navires et les autorités à terre — Exigences générales

**First edition
2025-11**

Reference number
ISO 18131:2025(en)

Full document details: BS ISO 18131:2025

© ISO 2025  **Documents**
Document Sharing Hub



COPYRIGHT PROTECTED DOCUMENT

© ISO 2025

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword	iv
Introduction	v
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Abbreviated terms	3
5 General requirements for publish-subscribe architecture on ship-shore data communication	4
5.1 General	4
5.2 Sequence diagram	5
6 Requirements for security management	6
6.1 General requirements	6
6.2 Application layer authentication	6
6.3 Authorization for topic permission	8
6.4 Data encryption by transport layer security	8
7 General requirements for message format	8
7.1 General requirements	8
7.2 Topic name	9
7.2.1 General	9
7.2.2 Topic naming definition	9
7.2.3 Topic name of data type	10
7.3 Data model	14
7.3.1 General	14
7.3.2 Header of data model	15
7.3.3 Payload of data model	15
8 Functional requirements for data management	29
8.1 Data backup	29
8.2 Data recovery	29
8.3 Log and event storage management	29
Annex A (informative) Examples of data taxonomy	30
Bibliography	136

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

ISO draws attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO takes no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents. ISO shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 8 *Ships and marine technology*, Subcommittee SC 26, *Smart shipping*.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

Information and Communication Technology (ICT) plays an increasingly important role in assuring effective and secure data communication in the marine industry. In this context, there is a growing need for an architecture that not only enables interoperability between applications and systems onboard vessels, but also supports real-time streaming services within ships. Moreover, data communication between ship and shore is provided in real time to support decision-making as integrated information by collecting and analysing data.

In order to expand the usage range between relative entities and to improve the ship's performance, it is important to standardize the message format transmitted from the ship side server to the shore side server.

Existing International Standards on managing data that are derived from ship or shore have been used in the industry such as:

- the IEC 61162 series for the digital interfaces of navigational equipment within a ship;
- ISO 19847 for shipboard data servers;
- ISO 19848 for standard data for shipboard machinery and equipment;
- ISO 23807 for asynchronous time-insensitive ship-shore data transmission.

However, it is recommended to establish a message communication system that shall respond flexibly to the various message formats derived from relevant sectors such as smart shipping, smart ports, smart logistics, etc. Furthermore, utilizing a communication protocol optimized for satellite communication can be considered, even in a situation where communication is intermittently disconnected. Furthermore, it should be highlighted that the data model is designed with scalability to accommodate future expansions related to eco-friendly technologies and environmental regulations.

This document defines a publish-subscribe architecture to facilitate widespread data utilization. It enables many-to-many communication by allowing data to be transferred through registered topics. It provides a dynamic, batch-based messaging structure.

This document provides general requirements for publish-subscribe communication, including security requirements, topic naming definitions, message format, and functional requirements for data management.

Ships and marine technology — Publish-subscribe architecture on ship-shore data communication — General requirements

1 Scope

This document specifies requirements on publish-subscribe architecture on ship-shore data communication. It is intended for use by system providers and users, including software vendors, platform developers, fleet managers, application users in the field of the shipbuilding industry, shipping companies, equipment manufacturers, and port and on-shore service providers, to communicate in the publish-subscribe architecture between ship and shore.

This document focuses on enabling interoperability of shipboard and shore-based systems for the exchange of operational data using publish-subscribe methods.

This document describes:

- the role of a broker, publisher and subscriber;
- the multitenancy-based data management system in the cloud environment;
- general requirements for publish-subscribe architecture;
- security requirements for ensuring data confidentiality, integrity and availability;
- the message format for client id and topic name, and data model;
- functional requirements for data management.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 24060, *Ships and marine technology — Ship software logging system for operational technology*

IEC 61162-460, *Maritime navigation and radiocommunication equipment and systems — Digital interfaces — Part 460: Multiple talkers and multiple listeners — Ethernet interconnection — Safety and security*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org/>

3.1

publish-subscribe architecture

system in which *publishers* (3.5) post messages to an intermediary *message broker* (3.4) and *subscribers* (3.6) register subscriptions with that broker, allowing the broker to filter, classify the client and designate the priority of messages

Note 1 to entry: The message broker performs a store-and-forward function to route messages from publishers to subscribers, with optional prioritization of messages in a queue before routing.

3.2

quality of service

QoS

level that is a deal between two parties, the *publisher* (3.5) and *subscriber* (3.6), of a message with respect to the assurance of distribution of data

3.3

topic naming definition

hierarchical structure in which clients publish messages on designated topics and receive messages that match those topics

Note 1 to entry: This structure comprises multiple topic levels separated by slash ("/") characters and allows the use of wildcards such as "+" and "#".

3.4

message broker

intermediary system responsible for controlling the distribution of data, and storing, forwarding, filtering and prioritizing public requests from the *publisher* (3.5) to the *subscriber* (3.6)

Note 1 to entry: Clients switch between publisher role and subscriber role depending on the functionalities desired.

3.5

publisher

client of ship side or shore side who sends messages with a topic to a *message broker* (3.4) and the topic matches with a subscription created by the client

3.6

subscriber

client on the ship side or shore side who receives published messages on a specific topic

Note 1 to entry: The subscriber obtains data by registering a subscription with the *message broker* (3.4) according to a defined topic structure.

3.7

multitenancy

architecture in which many tenants (users) simultaneously share the same software instance

Note 1 to entry: It allows a single *message broker* (3.4) to serve multiple tenants or customers, while keeping their data and communication separated from each other. It stores metadata about each tenant and uses this data to alter the software instance at runtime to fit each tenant's needs. The tenants are isolated from each other via permissions. Even though the tenants share the same software instance, each one uses and experiences the software differently. It consists of handling multiple IDs on a single server in cloud environments.

3.8

ship-shore data communication

exchange of information between a shipboard system and a shore-based system through a communication network

Note 1 to entry: It enables real-time sharing of operational, navigational, safety, environmental, and maintenance data, supporting enhanced monitoring, management, decision-making, and regulatory compliance for vessel operations. This communication typically utilizes a publish-subscribe architecture depending on the specific operational and technical requirements.

4 Abbreviated terms

API	application programming interface
CSV	comma-separated values
ICT	information and communication technology
IoT	internet of things
JSON	javascript object notation
QoS	quality of service
URL	uniform resource locator
UUID	universally unique identifier
VPN	virtual private network
XML	extensible markup language
XSD	XML schema definition language
MQTT	Message Queuing Telemetry Transport
AMQP	Advanced Message Queue Protocol
pub/sub	publish-subscribe
NO _x	Nitrogen Oxides
SO _x	Sulfur Oxides
LNG	liquefied natural gas
HT	high temperature
LT	low temperature
ECR	engine control room
LAN	local area network
UV	ultraviolet
TBN	total base number
BOG	boil-off gas
CCTV	closed-circuit television
RDF	radio direction finder
GPS	Global Positioning System
EPIRB	emergency position-indicating radio beacon

5 General requirements for publish-subscribe architecture on ship-shore data communication

5.1 General

Data for communication between ship and shore can comprise a variety of messages to navigation, machinery, cargo, administration etc. As a communication method, the message is exchanged between the server on the ship and the server on the shore, in which the most beneficial architecture is the publish-subscribe (pub/sub) architecture.

Known to be well-suited for IoT applications, the pub/sub architecture is designed to have a superior flexibility and scalability, and to be suitable for bandwidth saving and service expansion. The pub/sub architecture also enables bi-directional communication between ship and shore.

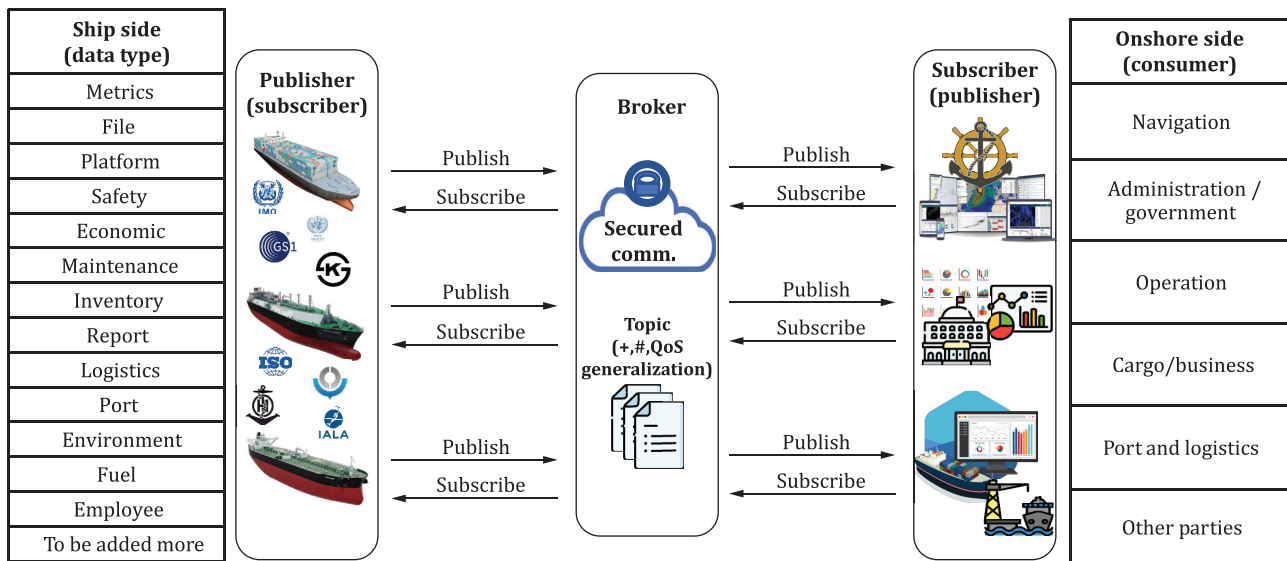


Figure 1 — Schematic of ship-shore data communication

For ship-shore data communication using the pub/sub architecture, data shall be published and subscribed in accordance with the topic naming definition specified in 7.2.2, whereby it is possible to enable communication between a single data source and a single recipient (one-to-one), or between multiple sources and multiple recipients (many-to-many), by registering data on a topic, as illustrated in Figure 1. Data streaming is continuously enabled as long as the registration of the topic is not cancelled.

For example, in the case where it is required to monitor the temperature of an engine air cooler in a specified vessel, the temperature can continuously be communicated without cancelling from the topic subscription.

The configuration of data streaming between ship and shore is based on a publisher, subscriber and message broker approach. Normally message brokers are located in onshore data centers, and publisher and subscribers are located to both ship and shore side. The flow of the data communication is as follows. First, data obtained from equipment in each vessel at sea is sent to the broker in the form of a configured topic. Subsequently, various clients including the onshore operation centre acquire the data through the broker, and vice versa.

In this document, it is necessary to consider communicating not only navigation and machinery equipment data on the ship side, but also data generated from shore-based systems. Thus, the pub/sub architecture can be useful in ship-shore data communication.

Since there are various data categories beyond navigation and machinery equipment, new data structures shall be established on the basis of the pub/sub architecture.

Therefore, the publish-subscribe architecture should be compatible with the standardized existing data structure for navigation and machinery equipment, and even be able to communicate between ship and shore by mapping the existing data structure into the pub/sub data patterns.

By viewing a semantic perspective, the architecture can be also compatible with a data quality index that consists of accuracy, completeness, consistency, credibility and timeliness, since the architecture has the benefit of qualifying the data by real-time integrity. This architecture is also applicable for data communication between ship-to-ship.

Accordingly, the configuration of the data communication between ship and shore is presented in [Figure 2](#).

5.2 Sequence diagram

In a pub/sub model, there is a broker in the middle as an intermediary. Clients can subscribe to receive messages from the broker and also publish messages to the broker which are then transmitted to all subscribed clients. A sequence diagram for the data communication flow is shown in [Figure 2](#).

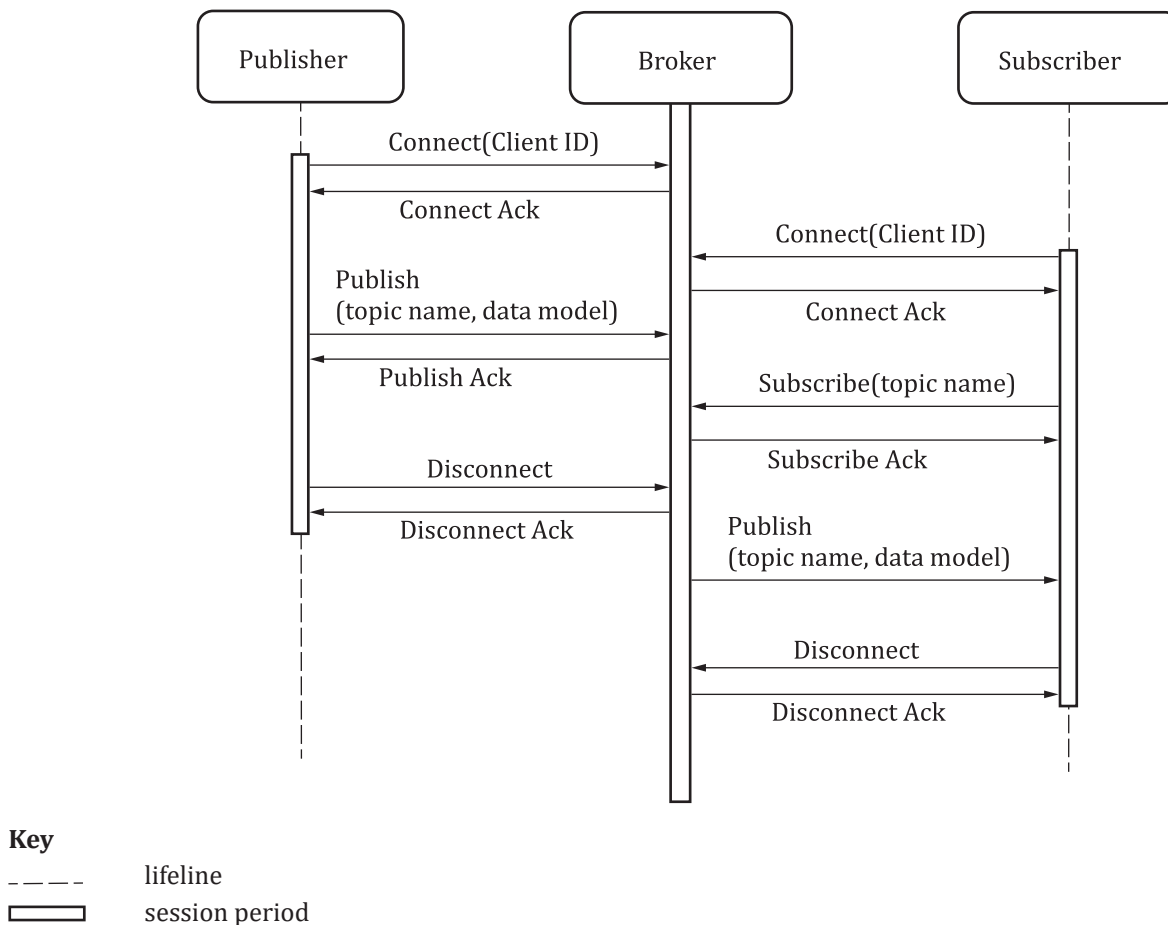


Figure 2 — Sequence diagram for data communication flow

A detailed explanation for topic name and data schema is given in [Clause 7](#).

From a broker's perspective, data publishing and subscribing occur simultaneously, so it can be viewed as bidirectional. For example, the package update or feedback command can be transferred by the pub/sub from ships, and also from shore.

The security against vulnerability caused by the direct opening of the data server for data communication shall be improved. A method shall be implemented that can compensate for the restriction of data communication due to the ship's dynamic IP address and the quality of data communication according to the state of the satellite. Thus, all these issues regarding security and unstable network conditions that can occur during ocean navigation can be managed while the broker acts as an intermediary.

[Figure 2](#) shows the functionalities on the high-level layer when sending a message by the pub/sub pattern, which can be also used on virtual private network (VPN). The diagram includes “Ack” messages, which indicate acknowledgements confirming message receipt.

When connecting to the broker, WebSocket can be used for real-time message delivery using the pub/sub pattern. The broker can support WebSocket communication by adjusting its configuration settings.

6 Requirements for security management

6.1 General requirements

To effectively address security issues that can arise in the communication between ship and shore, all data assets shall be protected by multiple security management methods such as authentication on the application layer, authorization for topic permission, data encryption and detection log management. Some of these methods are defined as follows.

- Authentication for application layer: The application layer belonging to the top layer of the protocol is easily attacked by hackers among other layers. The application layer is exposed to more network threats than other layers, thus the secure authentication method is required.
- Authorization for topic permission: Assuming that an open broker is not used, an access permission of the topic is required. Therefore, only authorized clients can publish and subscribe, and the broker should have a method for the access scheme towards the topic.
- Data encryption: When sensitive data are transmitted through network communication, data shall be encrypted so that it cannot be manipulated or tampered by a man-in-the-middle attack.

NOTE To enhance cyber security, other relevant rules, regulations and standards can apply.

6.2 Application layer authentication

In general, if the validity of the client certificate is proven, it can be used as an authentication for the client to access the server, and the server provides a digital certificate issued by a third party that can confirm the client's identity.

The primary purpose of a client ID is to serve as a unique identifier for each client that connects to a broker. This unique identifier is crucial for managing the connection and session state between the client and the broker.

An example of the naming definitions for the client ID is shown in [Table 1](#).

Table 1 — Example of client ID

Contents	Ship platform ID	-	Ship ID	-	Sequence	Remark
Number of digits	Three digits	-	Above 10 digits	-	Ship's randomized unique number	The number of digits is just an example.
Example	Shipboard data management server	-	IMO number (IMO xxxxxxxx), Official number, Ship series number, etc.	-	UUID, encrypted code	Ship platform ID-IMOxxxxxx-UUID or encrypted code

The client ID consists of the following three elements:

- Ship platform ID, which refers to a unique identifier assigned to a specific ship platform to facilitate communication and data exchange.
- Ship ID, a persistent identifier linked to a vessel throughout its lifetime, used to ensure accurate data routing and vessel tracking.

- Sequence, which may include a UUID and encrypted code, used to enhance security, identification, and message integrity.

For example, in a typical communication flow, the ship may first generate a randomized session-specific number. A UUID is then created and attached to the message. The message is encrypted to generate an encrypted code and is published to the broker. The shore-based system then decrypts the message using the shared key and validates the UUID to ensure session integrity and secure identification.

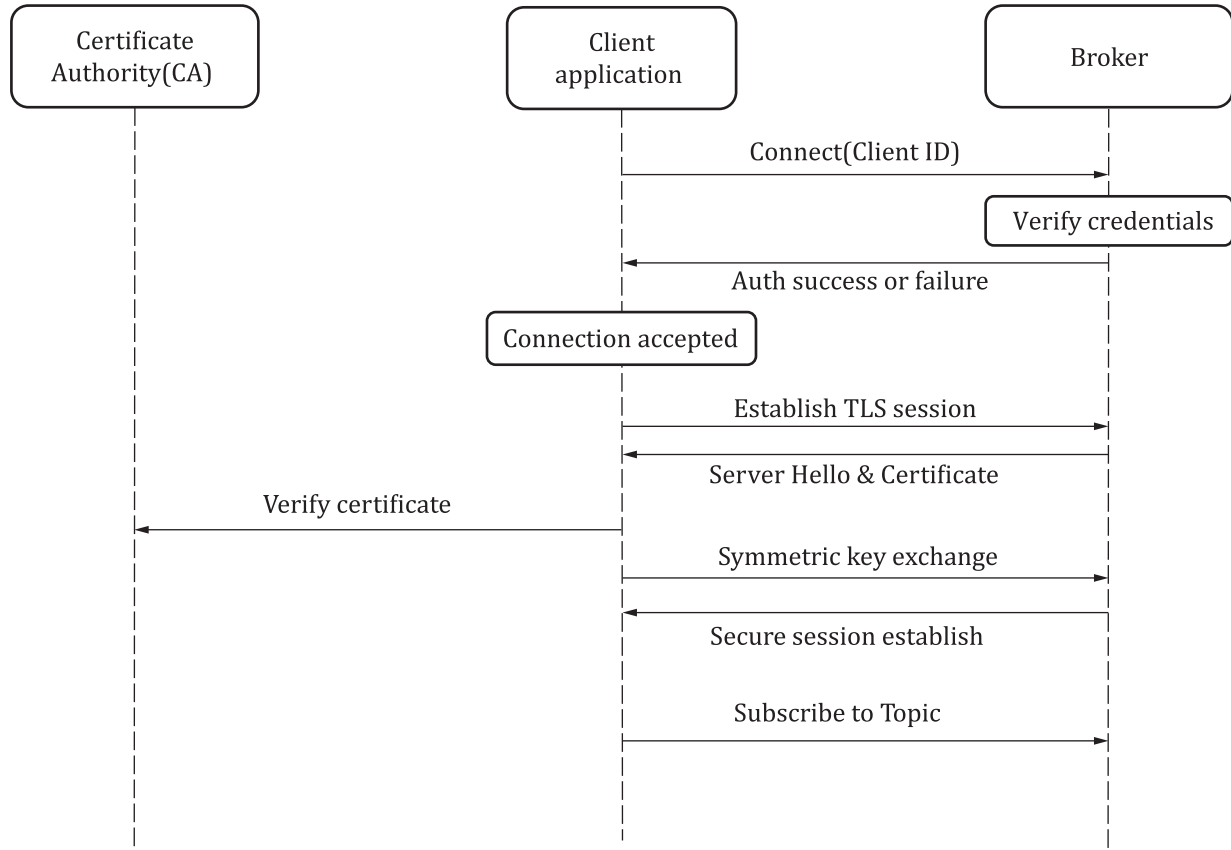


Figure 3 — Sequence diagram for the authentication process

[Figure 3](#) shows a sequence diagram for the authentication process. This diagram outlines the roles and interactions between entities, which are described as follows:

- Connection request: The client application connects to the broker using the client ID.
- Broker verification: The broker verifies the credentials against its user database.
- Response: The broker responds with either an “authentication success” or “authentication failure” message.
- Secure connection initiation: After successful authentication, the client application initiates a TLS handshake with the broker.
- Broker certificate provision: The broker provides its digital certificate to the client during the TLS handshake.
- Certificate verification: The client application verifies the broker’s certificate using a trusted certificate channel.
- Key exchange: A symmetric key exchange occurs to establish a secure communication channel.
- Secure communication: The client securely publishes and subscribes to topics through the broker after the secure session is established.

Since the digital certificate contains the server's public encryption key, the client receives the server's certificate to verify its authenticity. On the application layer, when sending a CONNECT message to the broker, the authentication is proven through the client ID and password. The username of the client ID is defined as a UTF-8 encoded string, and password is defined as binary data.

For example, OAuth2, which is widely used on the internet as an open authentication protocol, can also be used.

6.3 Authorization for topic permission

The authorization for topic permission means granting only authorized clients the right to access an allocated resource. The authorization management for accessing to the topic shall be assigned and managed by a broker. The authorized clients shall be permitted to publish to or subscribe to the topic.

The types of access permissions managed by the broker include:

- access to only a registered topic (intended one by clients);
- the restriction of the client's choice (only publishing, only subscribing or both);
- the selection of the Quality of Service (QoS 0,1,2)

6.4 Data encryption by transport layer security

The entities shall be able to confirm those entities that are trustworthy of each other. The contents of the communication between entities shall be protected from being eavesdropped by a third party. Therefore, a certificate with an electronic signature is used, and communication contents are encrypted. In other words, the transport layer security encrypts data sent over the internet to ensure that eavesdroppers and hackers are unable to see what data are transmitted. This is a method of generating encrypted data through a symmetric and asymmetric key, and of obtaining data using the same key for decryption.

7 General requirements for message format

7.1 General requirements

To communicate data in a pub/sub pattern, the following shall be defined:

- the type of data to communicate;
- how the message format is designed.

The message format is composed of the client ID, topic name and data model. Since the client ID is related to the authentication, see [Table 1](#) for further details. The topic name can be configured by topic naming definitions. The data model has two components: header and payload.

The message format firstly requires the authentication to access the topic, and allows mutual data users through the data model to check the header and the payload configurations.

The broker manages common topics as a tool to send data to multiple clients, and each client creates a response topic and delivers it to the broker, thereby creating a virtual connection between the publisher and the subscriber.

The messages are distributed in the formatted schema and the data model parses and checks the necessary data.

The hierarchical modelling for the message format can be implemented as shown in [Figure 4](#).

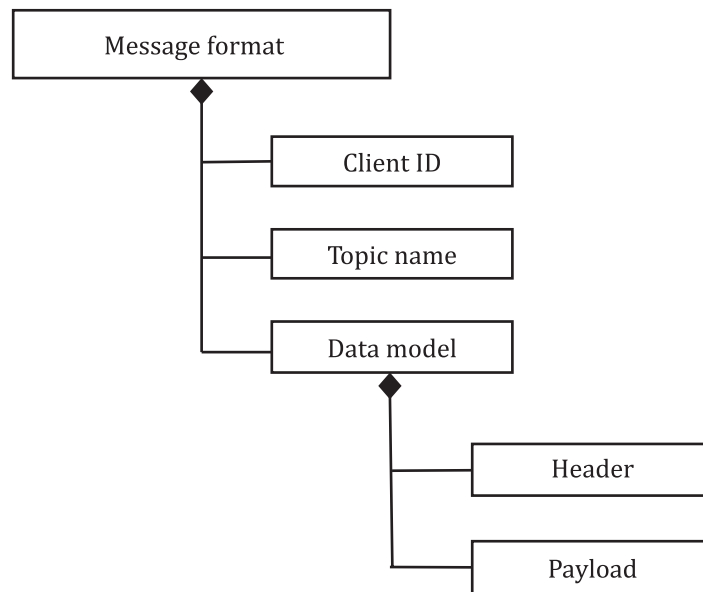


Figure 4 — Model of message format

7.2 Topic name

7.2.1 General

The topic name is generated by the topic naming definition and can be configured according to various data types.

7.2.2 Topic naming definition

The configuration of topic naming definitions uses delimiters to classify topic levels, and means that the broker uses it to filter messages coming from each connected client. As shown in [Table 2](#), the topic naming definition follows a four-level hierarchy.

Table 2 — Example of topic naming definition

Contents	Ship platform ID	/	Ship ID	/	Data type	/	Response
Alarm data	ShipplatformID	/	Vesselnumber (IMO number)	/	metrics/alarm	/	Applicable when sub- scribing
Example	(Publishing) shipplatformID/Vesselnumber/metrics/alarm (Subscribing) shipplatformID/Vesselnumber/metrics/alarm/response						

The contents refers to specific data elements included within a message, encompassing various fields and types of data as defined in [Table 4](#).

The ship platform ID refers to a unique identifier assigned to a specific ship platform to facilitate communication and data exchange.

The metrics/alarm refers to data used for monitoring and analysis; alarms serve as critical notifications prompting operators or system administrators to take immediate action to prevent potential issues or failures.

The response refers to the message sent from a server in reply to a client request, containing either the requested information or the result of the operation performed.

The data type for sending the alarm message derived from a component of the main engine is defined as metrics and the sub-element is designated as an alarm coming from the component.

[Table 2](#) shows that when publishing alarm data, the classification of topic level is defined by using slashes in the order of the ship platform ID, the ship's unique number (IMO number), and the data type (metrics/alarm).

[Table 2](#) also shows that when subscribing to the data, a response is added at the end of the publishing topic syntax. For this reason, a subscribing client can receive all the alarm data generated by the vessel, but if the subscriber wants to receive data only about the alarm of a specific component of the main engine, it can be parsed into more readable data formats. The component model structure is shown in [Annex A](#).

When subscribing to a topic, it can use wildcards to subscribe to multiple topics simultaneously. A wildcard can only be used to subscribe to topics, not to publish a message.

The Quality of Service (QoS levels) can be properly decided according to the data type or the priority.

It supports three levels of Quality of Service which are described below.

- QoS0 (At most once): The message is sent at most once and does not provide delivery guarantee of a message. It means the message is not stored and can be lost if the client is disconnected, or if the server fails. In the maritime industry, the architecture does not recommend servers to forward publications at QoS0 to a client.
- QoS1 (At least once): The message is sent at least once and it is possible to deliver a message more than once by setting the value of duplicate flag by 1. In maritime industry, the architecture does require QoS1 as the default mode of transfer in accordance with the [Table 3](#), because reliable transfer should be implemented where the communication failure occurred by the shadowing area or the satellite unstable condition.
- QoS2 (Exactly once): The message is sent exactly once by using four-way handshaking. Where the acknowledgement for the message transmission is required between publisher and subscriber, QoS2 should be required specially for the message of a ship control, query service, and portcall.

7.2.3 Topic name of data type

[Table 3](#) lists the data types that shall be referred to in the topic naming definitions specified in [Table 2](#).

Table 3 — Topic name of data type

Type	Subelement	Categorical data	Topic name	QoS ^a
Metrics	Alarm	metrics/alarm	ShipplatformID/Vesselnumber/metrics/alarm, ShipplatformID/Vesselnumber/metrics/alarm/ response	1
	Sensor	metrics/sensor	ShipplatformID/Vesselnumber/metrics/sensor, ShipplatformID/Vesselnumber/metrics/sensor/ response	1
File	Image	file/image	ShipplatformID/Vesselnumber/file/image, ShipplatformID/Vesselnumber/file/image/ response	1
	Voice	file/voice	ShipplatformID/Vesselnumber/file/voice, ShipplatformID/Vesselnumber/file/voice/ response	1
	Alarm	file/alarm	ShipplatformID/Vesselnumber/file/alarm, ShipplatformID/Vesselnumber/file/alarm/ response	1
	Sensor	file/sensor	ShipplatformID/Vesselnumber/file/sensor, ShipplatformID/Vesselnumber/file/sensor/ response	1

^a See [7.2.2](#) for an explanation of the QoS levels.

Table 3 (continued)

Type	Subelement	Categorical data	Topic name	QoS ^a
Platform	Monitoring	Platform /monitoring /platformlog	ShipplatformID/Vesselnumber/platform /monitoring /platformlog, ShipplatformID/Vesselnumber/platform /monitoring/platformlog/response	1
		Platform /monitoring /servicelog	ShipplatformID/Vesselnumber/platform/monitoring/servicelog, ShipplatformID/Vesselnumber/platform/monitoring/servicelog/response	1
	Ship particular	platform/ shipparticular	ShipplatformID/Vesselnumber/platform/ shipparticular, ShipplatformID/Vesselnumber/platform/ shipparticular/response	1
	Feedback command	Platform /feedbackcommand	ShipplatformID/Vesselnumber/platform/ feedbackcommand, ShipplatformID/Vesselnumber/platform/ feedbackcommand/response	2
	Query service	platform/query	ShipplatformID/Vesselnumber/platform/query, ShipplatformID/Vesselnumber/platform/query/ response	2
	Firmware update	Platform /firmware	ShipplatformID/Vesselnumber/platform/ firmware, ShipplatformID/Vesselnumber/platform/ firmware/response	1
	Platform management	Platform /management	*Ship to Shore ShipplatformID/Vesselnumber/platform/ management, ShipplatformID/Vesselnumber/platform/ management/response *Shore to Ship ShipplatformID/Vesselnumber/platform/ management/onshore, ShipplatformID/Vesselnumber/platform/ management/onshore/response	1
Safety	Hull stress	safety/hullstress	ShipplatformID/Vesselnumber/safety/hullstress, ShipplatformID/Vesselnumber/safety/hullstress/ response	1
	Fire	safety/riskfire	ShipplatformID/Vesselnumber/safety/riskfire, ShipplatformID/Vesselnumber/safety/riskfire/ response	1
	Explosion	Safety /riskexplosion	ShipplatformID/Vesselnumber/safety/ riskexplosion, ShipplatformID/Vesselnumber/safety/ riskexplosion /response	1
	Collision	Safety /riskcollision	ShipplatformID/Vesselnumber/safety/ riskcollision, ShipplatformID/Vesselnumber/safety/ riskcollision/response	1
	Sailing path	safety/sailingpath	ShipplatformID/Vesselnumber/safety/ sailingpath, ShipplatformID/Vesselnumber/safety/ sailingpath/response	1

^a See 7.2.2 for an explanation of the QoS levels.

Table 3 (continued)

Type	Subelement	Categorical data	Topic name	QoS ^a
Economic	Routing rule	Economic /routerule	ShipplatformID/Vesselnumber/economic/routerule, ShipplatformID/Vesselnumber/economic/routerule /response	1
	Insurance	Economic /insurance	ShipplatformID/Vesselnumber/economic/insurance, ShipplatformID/Vesselnumber/economic/insurance /response	1
Operation maintenance	Equipment	om/equipment	ShipplatformID/Vesselnumber/om/equipment, ShipplatformID/Vesselnumber/om/equipment /response	1
Inventory	Equipment	Inventory /equipment	ShipplatformID/Vesselnumber/inventory/equipment, ShipplatformID/Vesselnumber/inventory/equipment /response	1
	Safety device	Inventory /safetydevice	ShipplatformID/Vesselnumber/inventory/safetydevice, ShipplatformID/Vesselnumber/inventory/safetydevice /response	1
	Medicine	Inventory /medicine	ShipplatformID/Vesselnumber/inventory/medicine, ShipplatformID/Vesselnumber/inventory/medicine/response	1
	Grocery	inventory/grocery	ShipplatformID/Vesselnumber/inventory/grocery, ShipplatformID/Vesselnumber/inventory/grocery/response	1
Report	Noon report	report/noon	ShipplatformID/Vesselnumber/report/noon, ShipplatformID/Vesselnumber/report/noon /response	1
	Abstract report	report/ablog	ShipplatformID/Vesselnumber/report/ablog, ShipplatformID/Vesselnumber/report/ablog /response	1
	MRV	report/mrv	ShipplatformID/Vesselnumber/report/mrv, ShipplatformID/Vesselnumber/report/mrv /response	1
	CII	report/cii	ShipplatformID/Vesselnumber/report/cii, ShipplatformID/Vesselnumber/report/cii /response	1
	EEXI	report/eexi	ShipplatformID/Vesselnumber/report/eexi, ShipplatformID/Vesselnumber/report/eexi /response	1
Logistics	Cargo	logistics/cargo	ShipplatformID/Vesselnumber/logistics/cargo, ShipplatformID/Vesselnumber/logistics/cargo /response	1
	Container	logistics/container	ShipplatformID/Vesselnumber/logistics/container, ShipplatformID/Vesselnumber/logistics/container/response	1

^a See 7.2.2 for an explanation of the QoS levels.

Table 3 (continued)

Type	Subelement	Categorical data	Topic name	QoS ^a
Port	Portcall optimization	port/portcall	ShipplatformID/Vesselnumber/port/portcall, ShipplatformID/Vesselnumber/port/portcall/response	2
	Immigration	port/immigration	ShipplatformID/Vesselnumber/port/immigration, ShipplatformID/Vesselnumber/port/immigration/response	1
Environment	Ballast	Environment /ballast	ShipplatformID/Vesselnumber/environment/ballast, ShipplatformID/Vesselnumber/environment/ballast /response	1
	NoxSox	Environment /noxsox	ShipplatformID/Vesselnumber/environment/noxsox, ShipplatformID/Vesselnumber/environment/noxsox/response	1
	Emission	Environment /co2emission	ShipplatformID/Vesselnumber/environment/co2emission, ShipplatformID/Vesselnumber/environment/co2emission/response	1
	Weather	Environment /weather	ShipplatformID/Vesselnumber/environment/weather, ShipplatformID/Vesselnumber/environment/weather/response	1
Fuel	Volume	fuel/volume	ShipplatformID/Vesselnumber/fuel/volume, ShipplatformID/Vesselnumber/fuel/volume/response	1
	Charge	fuel/charge	ShipplatformID/Vesselnumber/fuel/charge, ShipplatformID/Vesselnumber/fuel/charge/response	1
Employee	Health	employee/health	ShipplatformID/Vesselnumber/employee/health, ShipplatformID/Vesselnumber/employee/health/response	1
	Information	Employee /information	ShipplatformID/Vesselnumber/employee/information, ShipplatformID/Vesselnumber/employee/information/response	1

^a See 7.2.2 for an explanation of the QoS levels.

The type refers to the classification of data as represented in [Table 4](#).

The subelement refers to a component or child element within a data type that contains properties or values nested under that data structure.

The categorical data refers to a type of data that can take on a set of distinct values with meaningful groupings or orderings enabling structured analysis and reporting.

The topic name refers to hierarchical string used to organize and route messages between publishers and subscribers, acting as an address that allows subscribers to receive relevant messages based on their subscriptions.

The QoS1 level should be applied to most data types to prevent communication failures in coverage-limited (blanket) areas. It is recommended to apply the QoS 2 level to data that required formal approval or confirmation between ship and shore systems, such as feedback command, operational acknowledgements, or regulatory reporting.

Depending on the system configuration, the topic level can be subdivided down to the equipment layer to support extended usability and detailed data mapping.

Details of each data type are described in [Table 4](#).

Table 4 — Description of data types

Type	Description
Metrics	Measures of quantitative value such as alarm or sensor commonly used for tracking and comparing performance or operation. i.e. analog (sensor measuring), digital (alarm, control)
File	Large amount of data such as alarm, sensor, image or voice.
Platform	An integrated set that collectively meets client's end-to-end data needs. It is necessary information for the efficient operation of the platform. i.e. monitoring, ship particular, feedback command, query service, firmware update, ship platform management
Safety	It is necessary information for the safe operation of ship. i.e. hull stress, fire and explosive information, collision risk, route information
Economic	Key information for economic management of the ship's operation. i.e. routing rule, insurance
Operation maintenance	Information related to the operation and maintenance service of equipment. i.e. equipment information, running hour
Inventory	Stocks related to equipment, medicines, and provision. i.e. equipment, safety device, first aids, food and drink
Report	Vessel's operation information and environmental data required by IMO. i.e. noon report, abstract report, Monitoring, Reporting and Verification (MRV), Carbon Intensity Indicator (CII), Energy Efficiency Existing Ship Index (EEXI)
Logistics	Logistics information linked to land-distribution. i.e. ship logistic information, container information
Port	Port arrival and departure information and immigration procedures. i.e. port call optimization, import and export immigration data
Environment	Environmental information on pollutants generated from ships and their surroundings. i.e. ballast, emission (NO _x /SO _x /CO ₂), weather information
Fuel	Fuel oil information for consumption, storage or supply on a ship. i.e. bunkering
Employee	Seafarer's role, work service and health information.
LO	Lubricating oil (LO) for consumption, storage or supply on a ship.
Water	Drinking water for consumption, storage or supply on a ship.
Chemical	Chemical contents (refrigerant, water treatment, cleaning, etc) in a ship.
Cargo (container)	Cargo information that can be counted, such as containers.
Cargo (liquid)	Cargo information that can be measured by weight or volume such as oil, liquid gas etc.
Cargo (bulk)	Cargo information that can be measured by weight or volume such as ore, coal, grain etc.

7.3 Data model

7.3.1 General

The data model has the two components: the header and the payload.

The header of data model defines when data communication occurs and who the parties of data communication are, which is the default value that shall be included whenever data communication occurs.

The payload of data model is the actual value of data communication, which can be used on the data parsing.

7.3.2 Header of data model

The header is composed of various parameters which include transactionId, tenantId, timestamp, shipId, companyId, dataFormat, contentType, version, compressor and timeunit.

The details of each parameter applied in a header are described in [Table 5](#).

The parameters listed in [Table 5](#) shall be used.

Table 5 — Description of headers

Parameter	Type	Description	Mandatory (M)/ optional (O)	Example
transactionId	String	UUID format, Any integer value for which a client manages requests/responses.	M	756ddc9b-62dc-49a9-b16c-fe7c10bb4879
tenantId	String	Tenant ID for onshore-management.	M	tenant_1
timestamp	String	Time for API creation. yyyy-MM-dd'T'HH:mm:ss.SSSSSS'Z' All timestamps refers to the ISO 8601 series UTC data format.	M	2023-02-10T13:49:11.000317Z
shipId	String	Vessel ID	M	IMO number
companyId	String	Shipowner ID	M	Onshore1
dataFormat	String	Data format for payload.	M	ISO 19848, GS1, IEC 61162-450, OPC UA etc
contentType	String	Content type for payload.	M	Application/json, Application/xml, text/xml, Text/csv, Application/<binary> using well-defined formats/elements, Image/jpeg, png
version	String	Version for data model.	M	1.0
compressor	String	Compressor algorithm for payload.	M	7z, zip, lz4, snappy, zlib
timeunit	String	Time for data management.	M	s, ds, cd, ms, us

Headers shall be fixed as stated in [Table 5](#). The business logic can be put into the payload because a tool such as tcpdump should be used to check the header while exchanging data.

7.3.3 Payload of data model

There are eight payload configurations according to the following parameters in [Table 6](#):

a) alarm/sensor (ship to onshore)

Although the following example has been defined according to the format in ISO 19848, the alarm/ sensor data can be taken flexibly according to each configuration. Thus, in this parameter, it becomes N/A.

Table 6 — Header description for alarm/sensor

Parameter	Type	Description	Mandatory (M)/ optional (O)
N/A	Object	Time Series Data Object	M

EXAMPLE 1

An example for a response payload is:

```

{
  "Package": {
    "ShipID": "IMOxxxxxxx"
    "TimeSeriesData": [
      {
        "DataConfiguration": {
          "ID": "IMOxxxxxxx",
          "TimeStamp": "2022-12-14T03:17:13.755000Z"
        },
        "TabularData": [
          {
            "NumberOfDataSet": 1,
            "NumberOfDataChannel": 2,
            "DataChannelID": [
              "/VDR/HEROT/HEROT_CH1",
              "/VDR/HEROT/HEROT_CH2"
            ],
            "DataSet": [
              {
                "TimeStamp": "2023-02-06T06:24:58.953141Z",
                "Value": [
                  "0.2",
                  "A"
                ],
                "Quality": [
                  "",
                  ""
                ]
              }
            ]
          }
        ]
      }
    ]
  }
}

```

b) Package (Firmware/Config) update request (onshore to ship)

When it is necessary to request a package update of firmware or configuration onshore, package update request should be sent, composed of the parameters specified in [Table 7](#), from onshore to ship.

Table 7 — Header description for package update request

Parameter		Type	Description	Mandatory/optional
package	type	object	Description of the package file itself	M
		string	Package Type 'P': Platform Package 'S': Service Package 'D': Data Analyser 'F': Firmware 'C': Config	M
	name	string	Package name	M
	version	string	Package version	M
	description	string	Package description	O
	vendor	object	Information of developer	O
	developer	string	Developer name	O
	contact	string	Developer contact	O
deployment	type	object	Description of how to deploy	M
		string	Deployment Type 'C': container 'T': tarbal	M
	origin	object	Source information for deployment	M
	harbor_url	string	Harbor URL (Mandatory, if 'C' Type)	MC
	filebase_url	string	File base URL (Mandatory, if 'T' Type)	MC
	target	object	Deployment target information	M
	equipment	string	Server type	O
file_list	Filename	array of objects	File List. (If tarbal type or single file type)	O
		String	File name	M
	checksum	string	sha1 hash	M
Key M Mandatory O Optional MC Mandatory, if container type				

EXAMPLE 2

An example for a request payload is:

```
{
  "package": {
    "type": "P",
    "name": "onboard-datap",
    "version": "1.0",
    "description": "Data processor",
    "vendor": {
      "developer": "developer",
      "contact": "iotdev@xxxxxx.com"
    }
  },
}
```

```

"deployment": {
  "type": "T",
  "origin": {
    "harbor_url": "http://192.168.7.154/onboard-dp/1.0/",
    "filebase_url": "ftp://192.168.7.154/onboard-dp/1.0/"
  },
  "target": {
    "equipment": "Generic Server"
  }
},
"file_list": [
  {"filename": "lib/aopalliance.aopalliance-1.0.jar", "checksum": "0235ba8b489512805ac13a8f9
ea77alca5ebe3e8"},
  ...
  {"filename": "lib/xml-apis.xml-apis-1.4.01.jar", "checksum": "3789d9fada2d3d458c4ba2de349d
48780f381ee3"}
]
}

```

c) Package (Firmware/Config) update response (ship to onshore)

When the firmware package or configuration has been successfully updated on the ship, a package update response composed of the parameters specified in [Table 8](#) should be sent from ship to shore.

Table 8 — Header description for package update response

Parameter	Type	Description	Mandatory (M)/ optional (O)	Example
resultCode	Integer	Result code on the success or fail — 0: Success — 1: Fail	M	0
resultDesc	String	Message for communication failure where the result code is 1.	O	

EXAMPLE 3

An example for response payload is:

success

```

{
  "resultCode": 0,
}

```

fail

```

{
  "resultCode": 1,
  "resultDesc": "unauthorized"
}

```

The shipboard data server shall manage software version logging and onboard software maintenance requirements in accordance with ISO 24060 and IEC 61162-460.

d) Feedback command request (onshore to ship)

When a feedback command request is required onshore, a feedback command request composed of the parameters specified in [Table 9](#) should be sent from shore to ship.

Table 9 — Header description for feedback command request

Parameter	Type	Description	Mandatory (M)/ optional (O)
command	String	Command name(ex, 'ls -al')	M

EXAMPLE 4

An example for request payload is:

```
{
  "command": "cat tempfile.txt"
}
```

e) Feedback command response (ship to shore)

When the feedback command request finishes on the ship, a feedback command response composed the parameters specified in [Table 10](#) should be sent from ship to shore.

Table 10 — Header description for feedback command response

Parameter	Type	Description	Mandatory (M)/ optional (O)	Example
resultCode	Integer	Result code on the success or fail — 0: Success — 1: Fail	M	0
resultDesc	String	Message for communication Success: result of command Failure: error desc if the result code is 1.	O	

EXAMPLE 5

An example for response payload is:

```
success
{
  "resultCode": 0,
  "resultDesc": "Example of control data"
}

fail
{
  "resultCode": 1,
  "resultDesc": "unauthorized"
}
```

f) Query service (shore to ship)

When a query service is required onshore, a query service composed of the parameters specified in [Table 11](#) should be sent from shore to ship.

Table 11 — Header description for query service (shore to ship)

Parameter	Type	Description	Mandatory (M)/ optional (O)
command	String	Command name(ex, 'queryData')	M
dataChannelId	List<String>	Retrieving the list of data channel ID	M
offset	String	A time for retrieving data, expressed in accordance with the ISO 8601 series	O
before	Boolean	A flag that selects whether to retrieve the previous value (true) or the subsequent value (false) based on the offset when retrieving data.	O
header	Boolean	A flag that determines whether to display header values as specified in ISO 19848 in data responses.	O
limit	Integer	Limit the number of data responses.	O
orderBy	String	Sort criteria for data response. Descending, or ascending	O
idType	String	A value indicating for ID type. LocalID, ShortID, Alias	O
label	String	A value determining data collation priority. LocalID, ShortID	O

EXAMPLE 6

An example for request payload is:

```
{
  "command": "queryData",
  "dataChannelId": [
    "/jsmea_mac/IAS/read/Content1/Outlet/BG003PC.PIDA.SP",
    "/jsmea_mac/IAS/read/Content1/Outlet/BG013APC.PIDA.SP"
  ],
  "offset": "2023-02-13T09:10:05Z",
  "before": true,
  "header": true,
  "limit": 1,
  "orderBy": "desc",
  "idType": "LocalID",
  "label": "LocalID"
}
```

The request parameters for a query service regarding fuel consumption may include fuel consumption data, associated metadata, and additional detailed information specified in example 7.

EXAMPLE 7

An example for request payload is:

```

{
  "FuelConsumptionData": {
    "VesselID": {
      "Type": "IMO",
      "Value": "IMO1234567"
    },
    "Metadata": {
      "RequestID": "REQ-2025-001",
      "ResponseDate": "2025-02-17T12:05:00Z",
      "Status": {
        "Code": "200",
        "Description": "Successfully processed"
      }
    },
    "Timestamp": "2025-02-17T12:00:00Z",
    "FuelData": {
      "Type": "Diesel",
      "QuantityConsumed": {
        "Amount": {
          "Value": 120,
          "Unit": "Liters"
        }
      },
      "DistanceTravelled": {
        "Amount": {
          "Value": 250,
          "Unit": "NauticalMiles"
        }
      },
      "AverageConsumption": {
        "Value": {
          "Amount": 0.48,
          "Unit": "Liters/NM"
        }
      }
    }
  }
}

```

An example XML schema for the fuel consumption query service response is provided in example 8.

EXAMPLE 8

An example for XML schema is:

```

< xs:schema xmlns:xs = "https://www.w3.org/2001/XMLSchema" >

  < xs:element name = "FuelConsumptionResponse" >
    <xs:complexType>
      <xs:sequence>
        < xs:element name = "RequestID" type = "xs:string"/ >
        < xs:element name = "Vessel" >
          <xs:complexType>
            <xs:sequence>
              < xs:element name = "IMO" type = "xs:string"/ >
              < xs:element name = "Name" type = "xs:string"/ >
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        < xs:element name = "FuelData" >
          <xs:complexType>
            <xs:sequence>
              < xs:element name = "Timestamp" type = "xs:dateTime"/ >
              < xs:element name = "FuelType" type = "xs:string"/ >
              < xs:element name = "QuantityConsumed" >
                <xs:complexType>
                  <xs:sequence>
                    < xs:element name = "Value" type = "xs:float"/ >
                    < xs:element name = "Unit" type = "xs:string"/ >
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
              < xs:element name = "DistanceTravelled" >
                <xs:complexType>
                  <xs:sequence>
                    < xs:element name = "Value" type = "xs:float"/ >
                    < xs:element name = "Unit" type = "xs:string"/ >
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
              < xs:element name = "AverageConsumption" >
                <xs:complexType>
                  <xs:sequence>
                    < xs:element name = "Value" type = "xs:float"/ >
                    < xs:element name = "Unit" type = "xs:string"/ >
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs>

```

g) Query service (ship to shore)

When the query service from shore on ship finishes, a query service composed the parameters specified in [Table 12](#) should be sent from ship to shore.

Table 12 — Header description for query service (ship to shore)

Parameter	Type	Description	Mandatory (M)/ optional (O)
result	String	Response result (Success or Failure)	M
body	Object	— Where the result indicates "success": time series data object, — Where the result indicates "failure": errorCode and Object involving an error message	M
command	String	Command name, "queryData"	M

EXAMPLE 9

A success example for response payload is:

```
{
  "result": "success",
  "command": "queryData",
  "body": {
    "Package": {
      "Header": {
        "ShipID": "Ship"
      },
      "TimeSeriesData": [
        {
          "DataConfiguration": {
            "ID": "IAS-read",
            "TimeStamp": "2021-08-12T00:00:00.000000Z"
          },
          "TabularData": [
            {
              "NumberOfDataSet": 1,
              "NumberOfDataChannel": 2,
              "DataChannelID": [
                "/jsmea_mac/IAS/read/Content1/Outlet/BG003PC.PIDA.SP",
                "/jsmea_mac/IAS/read/Content1/Outlet/BG013APC.PIDA.SP"
              ],
              "DataSet": [
                {
                  "TimeStamp": "2021-09-23T07:07:45.271356Z",
                  "Value": [
                    "1334.0",
                    "442.0"
                  ]
                }
              ]
            }
          ]
        }
      ]
    }
  }
}
```

```

        "Quality": [
            "0",
            "0"
        ]
    }
]
}
]
}
]
}
}
}
}
}
}
}

```

EXAMPLE 10

A failure example for response payload is:

```

{
  "result": "failure",
  "command": "queryData",
  "body": {
    "message": "platform connection error: api",
    "errorCode": "504"
  }
}

```

h) Custom declarations (ship to shore)

When a custom declaration is required in GS1 dataformat, the following parameters specified in [Table 13](#) should be sent from ship to shore.

Table 13 — Header description for custom declarations (ship to shore)

Parameter	Type	Description	Mandatory (M)/ optional (O)
DocumentType	String	Type of the document (e.g. CustomsDeclaration)	M
ReferenceNumber	String	Unique reference number for the declaration	M
Date	String	Date when the declaration is made	M
Time	String	Time when the declaration is made	M
PortOfEntry	String	Port where the goods are entering	M
Vessel IMO	String	IMO number of the vessel	M
VoyageNumber	String	Voyage number of the transport	M
Consignor ID	String	Unique identifier for the consignor	M
Consignor Name	String	Name of the consignor	M
Consignor Address	Object	Address information for the consignor	M
Consignee ID	String	Unique identifier for the consignee	M
Consignee Name	String	Name of the consignee	M

Table 13 (continued)

Parameter	Type	Description	Mandatory (M)/ optional (O)
Consignee Address	Object	Address information for the consignee	M
GoodsItems	Array	List of goods items within the declaration	M
DeploymentID	String	Unique identifier for the deployment	M
Method	String	Method of deployment (e.g. API, FTP)	M
ServerURL	String	URL of the server for data deployment	M
AccessCredentials	Object	Access credentials for the server	M
Format	String	Format of the data being sent (e.g. JSON)	M

EXAMPLE 11

A example for request payload is:

```
{
"@context": ["https://gs1.github.io/EPCIS/epcis-context.jsonld",
{
"@vocab": "https://ns.gs1.org/epcis-masterdata/",
"cbvmda": "urn:epcglobal:cbv:mda":
},
{
"ext1": "https://ext.com/ext1",
"ext2": "https://ext.com/ext2",
"Header": {
"ShipID": "IMO1234567",
"DataChannelListID": {
"ID": "DataChannelList.xml",
"TimeStamp": "2016-01-01T00:00:00Z"
},
}
},
"isa": "EPCISDocument",
"schemaVersion": "2.0",
"Author": "Author1",
"creationDate": "2015-12-01T00:00:00Z",
"epcisHeader": {
"epcisMasterData": {
"vocabularyList": [
{
{
"CustomsDeclaration": {
"DocumentType": {
"Value": "CustomsDeclaration"
},
"DocumentNumber": {
```

```

    "Value": "DECL-2025-0001"
  },
  "Date": {
    "Value": "2025-02-17"
  },
  "Time": {
    "Value": "08:30:00"
  },
  "PortOfEntry": {
    "Value": "Busan"
  },
  "Vessel": {
    "IMO": {
      "Value": "IMO1234567"
    },
    "VoyageNumber": {
      "Value": "VN123"
    }
  },
  "Consignor": {
    "Name": {
      "Value": "Global Goods Corp."
    },
    "Address": {
      "Street": {
        "Value": "456 Export Ave"
      },
      "City": {
        "Value": "Bigcity"
      },
      "State": {
        "Value": "CA"
      },
      "PostalCode": {
        "Value": "67890"
      },
      "Country": {
        "Value": "USA"
      }
    },
    "TaxIdentifier": {
      "Value": "987-65-4321"
    }
  },
  "Consignee": {

```

```

    "Name": {
      "Value": "John Doe Enterprises"
    },
    "Address": {
      "Street": {
        "Value": "123 Main St"
      },
      "City": {
        "Value": "Anytown"
      },
      "State": {
        "Value": "NY"
      },
      "PostalCode": {
        "Value": "12345"
      },
      "Country": {
        "Value": "USA"
      }
    },
    "TaxIdentifier": {
      "Value": "123-45-6789"
    }
  },
  "GoodsItems": [
    {
      "ItemID": {
        "Value": "ITEM001"
      },
      "Description": {
        "Value": "Electronics - Smartphone"
      },
      "Quantity": {
        "Value": 100,
        "Unit": "PCS"
      },
      "CustomsValue": {
        "Currency": {
          "Value": "USD"
        },
        "Amount": {
          "Value": 50000
        }
      },
      "TariffCode": {

```

```

        "Value": "851712"
    },
    {
        "ItemID": {
            "Value": "ITEM002"
        },
        "Description": {
            "Value": "Electronics - Laptop"
        },
        "Quantity": {
            "Value": 50,
            "Unit": "PCS"
        },
        "CustomsValue": {
            "Currency": {
                "Value": "USD"
            },
            "Amount": {
                "Value": 75000
            }
        },
        "TariffCode": {
            "Value": "847130"
        }
    }
],
"Deployment": {
    "DeploymentMethod": {
        "Value": "API"
    },
    "ServerURL": {
        "Value": "https://api.port.com/customs/declaration"
    },
    "AccessCredentials": {
        "Username": {
            "Value": "user123"
        },
        "Password": {
            "Value": "securepassword"
        }
    },
    "Format": {
        "Value": "JSON"
    }
},

```

```

    },
  }
}

```

8 Functional requirements for data management

8.1 Data backup

The data server shall be able to implement a backup of the system function in accordance with a general process.

8.2 Data recovery

There shall be data recovery settings and capabilities, such as a buffering and recovery mechanism, to avoid data loss in the data server.

8.3 Log and event storage management

The data server shall have a functionality for the management of log and event storage to historically manage, access and resuse event data.

Annex A (informative)

Examples of data taxonomy

A.1 Component model

As shown in [Table 3](#), the example of topic naming definitions is organized down to the subelement level. Nevertheless, a more detailed topic structure should be considered to enable finer-grained event filtering, depending on system requirements.

The component model consists of the eight kinds of systems as follows:

- Propulsion system: comprising a model for main engine which includes a diesel engine, dual fuel engine, turbine, propulsion motor etc.
- Auxiliary system: comprising a model for auxiliary equipment which includes fuel and lubricating oil, refrigerant system, boiler, bilge system etc.
- Electric system: comprising a model for electric system, which includes a generator, distribution system, lighting system, navigation and communication equipment etc.
- Hull and deck system: comprising a model for hull and deck equipment, which includes manoeuvring machinery, anchoring, mooring equipment, ballast, lifting, a fire-fighting system etc.
- Cargo system: comprising a model which includes cargo handling, cargo hold and deck equipment etc.
- Accommodation: comprising a model for accommodation equipment, which includes heating, ventilation, air conditioning system etc.
- Monitoring and control system: comprising a model for a ship automation system, which includes monitoring and control equipment.
- Navigation and communication system: comprising a model for navigation and communication equipment, which includes navigation and radar system, data transmission system etc.

Additionally, the component model to be added after the subelement shall be defined.

A.2 Model structure

The component model structure is defined as a hierarchical structure with five levels:

- top-level group (level 1),
- group (level 2),
- equipment (level 3),
- component (level 4).
- measuring value (level 5).

The component model structure can be used in the publish-subscribe model for data communication, such as MQTT and AMQP.

The top-level group (level 1) is divided into two main categories:

- main system, and
- auxiliary system.

Together, these categories comprise the following systems:

- propulsion system,
- electric system,
- hull and deck system,
- cargo system,
- accommodation,
- monitoring and control system,
- navigation and communication system,
- miscellaneous systems.

The group (level 2) is a sub-group of the top-level group (level 1).

The equipment (level 3) is made of specific equipment and a system, which is part of the group (level 2).

The component (level 4) is the lowest hardware level, which is part of the equipment (level 3).

The measuring value (level 5) refers to data obtained from sensors, including physical quantities such as temperature, pressure, and flow rate, as well as operational status indicators such as trip and alarm states.

In order to support data communication for eco-friendly ships, this annex provides a reference data model.

An example structure covering dual-fuel engines, LNG systems, and fuel gas supply systems is provided in [Table A.1](#).

Table A.1 — Configuration example of model structure

Level	Elements
1) Top-level group	Auxiliary system
2) Group	Fuel system
3) Equipment	Fuel gas supply system for main engine
4) Component	Fuel gas supply system(room) Gas valve unit(gas regulating train, gas regulating unit) Gas valve unit extraction fan Gas valve unit double block breed valve Gas valve unit gas shut off valve (ESD) Gas valve unit gas venting valve Gas valve unit N ₂ purge valve Gas valve unit gas flowmeter Gas valve unit gas supply line Gas valve unit N ₂ purge line Gas valve unit annular space Master gas valve (MGV) Gas valve unit filter Control panel
5) Measuring value	Sensing and status

An example of a model structure for a fuel gas supply system is defined in [Table A.2](#).

Table A.2 — Model structure for levels of fuel gas supply system

Level 1	Level 2	Level 3	Level 4	Level 5
Auxiliary system	Fuel system	Fuel gas supply system	Fuel gas supply system (room)	Sensing (room temperature, gas concentration, leak detection)
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit (gas regulating train, gas regulating unit)	Sensing (gas pressure, gas temperature, gas flow rate), Status (valve open/close, pressure normal/alarm)
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit extraction fan	Sensing (pressure), Status (running or stop, remote/abnormal/trip)
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit double block bleed valve	Status (valve open/close)
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit gas shut off valve (ESD)	Status (valve open/close, trip)
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit gas venting valve	Status (valve open/close, ESD trigger logs)
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit N ₂ purge valve	Status (valve open/close, purge operation)
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit gas flowmeter	Sensing (gas flow rate (kg/h), massflow (kg), Status (calibration status))
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit gas supply line	Sensing (gas pressure, gas temperature, gas leakage),
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit N ₂ purge line	Sensing (line pressure, N ₂ purge flowrate [N m ³ /h]), Status (valve open/close, purging completion)
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit annular space	Sensing (gas pressure, gas detection and gas leakage) Status (N ₂ purge flowrate [Nm ³ /h])
Auxiliary system	Fuel system	Fuel gas supply system	Master gas valve (MGV)	Status (valve open/close, ESD trip)
Auxiliary system	Fuel system	Fuel gas supply system	Gas valve unit filter	Status (clogging alarm, valve bypassing)
Auxiliary system	Fuel system	Fuel gas supply system	Control panel	Status (remote/manual, ESD trip, running/standby or maintenance mode)

Details of the model structure for each system are defined in [Tables A.3](#) to [A.11](#).

Table A.3 — Model structure for levels of propulsion system

Level 1	Level 2	Level 3	Level 4
Propulsion system	Engine	Diesel engine	Main diesel engine
Propulsion system	Engine	Diesel engine	Main engine cylinder
Propulsion system	Engine	Diesel engine	Main engine cylinder head
Propulsion system	Engine	Diesel engine	Main engine cylinder cover
Propulsion system	Engine	Diesel engine	Main engine cylinder liner
Propulsion system	Engine	Diesel engine	Main engine cylinder oil line
Propulsion system	Engine	Diesel engine	Main engine piston

Table A.3 (continued)

Level 1	Level 2	Level 3	Level 4
Propulsion system	Engine	Diesel engine	Main engine connecting rod
Propulsion system	Engine	Diesel engine	Main engine bottom end bearing
Propulsion system	Engine	Diesel engine	Main engine main bearing
Propulsion system	Engine	Diesel engine	Main engine crosshead bearing
Propulsion system	Engine	Diesel engine	Main engine bedplate
Propulsion system	Engine	Diesel engine	Main engine crankcase
Propulsion system	Engine	Diesel engine	Main engine camshaft
Propulsion system	Engine	Diesel engine	Main engine crankshaft
Propulsion system	Engine	Diesel engine	Main engine turbocharger
Propulsion system	Engine	Diesel engine	Main engine servo oil pump
Propulsion system	Engine	Diesel engine	Main engine servo oil rail
Propulsion system	Engine	Diesel engine	Main engine fuel rail
Propulsion system	Engine	Diesel engine	Main engine fuel pipe
Propulsion system	Engine	Diesel engine	Main engine fuel pump
Propulsion system	Engine	Diesel engine	Main engine fuel oil injector
Propulsion system	Engine	Diesel engine	Main engine starting air distributor
Propulsion system	Engine	Diesel engine	Main engine start air line
Propulsion system	Engine	Diesel engine	Main engine scavenging air cooler
Propulsion system	Engine	Diesel engine	Main engine scavenging air filter
Propulsion system	Engine	Diesel engine	Main engine scavenging air pump
Propulsion system	Engine	Diesel engine	Main engine scavenging air space
Propulsion system	Engine	Diesel engine	Main engine scavenging air receiver
Propulsion system	Engine	Diesel engine	Main engine auxiliary blower
Propulsion system	Engine	Diesel engine	Main engine exhaust valve
Propulsion system	Engine	Diesel engine	Main engine exhaust valve actuator
Propulsion system	Engine	Diesel engine	Main engine exhaust valve spring air
Propulsion system	Engine	Diesel engine	Main engine lub oil line
Propulsion system	Engine	Diesel engine	Main engine lub oil sump tank
Propulsion system	Engine	Diesel engine	Main engine vibration damper
Propulsion system	Engine	Diesel engine	Main engine coupling
Propulsion system	Engine	Diesel engine	Main engine control air line
Propulsion system	Engine	Diesel engine	Main engine cooling water line
Propulsion system	Engine	Diesel engine	Main engine turning gear
Propulsion system	Engine	Diesel engine	Main engine reversing/starting gear
Propulsion system	Engine	Diesel engine	Main engine special tool
Propulsion system	Engine	Diesel engine	Main engine governor
Propulsion system	Engine	Diesel engine	Main engine fuel index limiter
Propulsion system	Engine	Diesel engine	Hydraulic top bracing
Propulsion system	Engine	Diesel engine	Oil mist detector
Propulsion system	Engine	Diesel engine	Water mist catcher
Propulsion system	Engine	Diesel engine	Main engine exhaust gas receiver

Table A.3 (continued)

Level 1	Level 2	Level 3	Level 4
Propulsion system	Engine	Diesel engine	Main engine turbocharger turbine
Propulsion system	Engine	Diesel engine	Main engine turbocharger compressor
Propulsion system	Engine	Dual fuel engine	Main dual fuel engine
Propulsion system	Engine	Dual fuel engine	Main engine cylinder
Propulsion system	Engine	Dual fuel engine	Main engine cylinder head
Propulsion system	Engine	Dual fuel engine	Main engine cylinder cover
Propulsion system	Engine	Dual fuel engine	Main engine cylinder liner
Propulsion system	Engine	Dual fuel engine	Main engine cylinder oil line
Propulsion system	Engine	Dual fuel engine	Main engine piston
Propulsion system	Engine	Dual fuel engine	Main engine connecting rod
Propulsion system	Engine	Dual fuel engine	Main engine bottom end bearing
Propulsion system	Engine	Dual fuel engine	Main engine main bearing
Propulsion system	Engine	Dual fuel engine	Main engine crosshead bearing
Propulsion system	Engine	Dual fuel engine	Main engine bedplate
Propulsion system	Engine	Dual fuel engine	Main engine crankcase
Propulsion system	Engine	Dual fuel engine	Main engine camshaft
Propulsion system	Engine	Dual fuel engine	Main engine crankshaft
Propulsion system	Engine	Dual fuel engine	Main engine turbocharger
Propulsion system	Engine	Dual fuel engine	Main engine servo oil pump
Propulsion system	Engine	Dual fuel engine	Main engine servo oil rail
Propulsion system	Engine	Dual fuel engine	Main engine fuel rail
Propulsion system	Engine	Dual fuel engine	Main engine fuel pipe
Propulsion system	Engine	Dual fuel engine	Main engine fuel pump
Propulsion system	Engine	Dual fuel engine	Main engine fuel oil injector
Propulsion system	Engine	Dual fuel engine	Main engine starting air distributor
Propulsion system	Engine	Dual fuel engine	Main engine start air line
Propulsion system	Engine	Dual fuel engine	Main engine scavenging air cooler
Propulsion system	Engine	Dual fuel engine	Main engine scavenging air filter
Propulsion system	Engine	Dual fuel engine	Main engine scavenging air pump
Propulsion system	Engine	Dual fuel engine	Main engine scavenging air space
Propulsion system	Engine	Dual fuel engine	Main engine scavenging air receiver
Propulsion system	Engine	Dual fuel engine	Main engine auxiliary blower
Propulsion system	Engine	Dual fuel engine	Main engine exhaust valve
Propulsion system	Engine	Dual fuel engine	Main engine exhaust valve actuator
Propulsion system	Engine	Dual fuel engine	Main engine exhaust valve spring air
Propulsion system	Engine	Dual fuel engine	Main engine lub oil line
Propulsion system	Engine	Dual fuel engine	Main engine lub oil sump tank
Propulsion system	Engine	Dual fuel engine	Main engine vibration damper
Propulsion system	Engine	Dual fuel engine	Main engine coupling
Propulsion system	Engine	Dual fuel engine	Main engine control air line

Table A.3 (continued)

Level 1	Level 2	Level 3	Level 4
Propulsion system	Engine	Dual fuel engine	Main engine cooling water line
Propulsion system	Engine	Dual fuel engine	Main engine turning gear
Propulsion system	Engine	Dual fuel engine	Main engine reversing/starting gear
Propulsion system	Engine	Dual fuel engine	Main engine special tool
Propulsion system	Engine	Dual fuel engine	Gas admission valve
Propulsion system	Engine	Dual fuel engine	Gas pipe
Propulsion system	Engine	Dual fuel engine	Pilot fuel injector
Propulsion system	Engine	Dual fuel engine	Pressure relief valve
Propulsion system	Engine	Dual fuel engine	Exhaust line rupture disc
Propulsion system	Engine	Dual fuel engine	Main engine N ₂ supply valve
Propulsion system	Engine	Dual fuel engine	Hydraulic top bracing
Propulsion system	Engine	Electronically controlled engine system	Engine control system (ECS)
Propulsion system	Engine	Electronically controlled engine system	Auxiliaries control unit (ACU)
Propulsion system	Engine	Electronically controlled engine system	Cylinder control unit (CCU)
Propulsion system	Engine	Electronically controlled engine system	Engine control unit (ECU)
Propulsion system	Engine	Electronically controlled engine system	Engine interface control unit (EICU)
Propulsion system	Engine	Electronically controlled engine system	Scavenge air control unit (SCU)
Propulsion system	Engine	Electronically controlled engine system	Cooling water control unit (CWCU)
Propulsion system	Engine	Electronically controlled engine system	Local display unit (LDU)
Propulsion system	Engine	Electronically controlled engine system	Main operating panel (MOP)
Propulsion system	Engine	Electronically controlled engine system	Local operating panel (LOP)
Propulsion system	Engine	Electronically controlled engine system	Hydraulic lubricant oil pump
Propulsion system	Engine	Electronically controlled engine system	Hydraulic lubricant oil filter
Propulsion system	Engine	Electronically controlled engine system	Hydraulic lubricant oil system
Propulsion system	Engine	Electronically controlled engine system	Cylinder control module (CCM)
Propulsion system	Engine	Electronically controlled engine system	Main control module (MCM)
Propulsion system	Engine	Electronically controlled engine system	Input output module (IOM)
Propulsion system	Engine	Electronically controlled engine system	Fuel oil rail unit
Propulsion system	Engine	Electronically controlled engine system	Fuel flow limit valve (FLV)
Propulsion system	Engine	Electronically controlled engine system	Valve control unit (VCU)

Table A.3 (continued)

Level 1	Level 2	Level 3	Level 4
Propulsion system	Engine	Electronically controlled engine system	Crank angle sensor unit (CAS)
Propulsion system	Turbine	Steam turbine	Main steam turbine
Propulsion system	Turbine	Steam turbine	Speed governor
Propulsion system	Turbine	Steam turbine	Turning gear
Propulsion system	Turbine	Steam turbine	Vacuum pump
Propulsion system	Turbine	Steam turbine	Vacuum ejector
Propulsion system	Turbine	Steam turbine	High pressure turbine
Propulsion system	Turbine	Steam turbine	High pressure turbine casing
Propulsion system	Turbine	Steam turbine	High pressure turbine diaphragm
Propulsion system	Turbine	Steam turbine	High pressure turbine rotor
Propulsion system	Turbine	Steam turbine	High pressure turbine nozzle
Propulsion system	Turbine	Steam turbine	High pressure turbine main bearing
Propulsion system	Turbine	Steam turbine	High pressure turbine thrust bearing
Propulsion system	Turbine	Steam turbine	High pressure turbine coupling
Propulsion system	Turbine	Steam turbine	High pressure turbine manoeuvre valve
Propulsion system	Turbine	Steam turbine	High pressure turbine drain valve
Propulsion system	Turbine	Steam turbine	High pressure turbine governor
Propulsion system	Turbine	Steam turbine	High pressure steam line
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine casing
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine diaphragm
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine rotor
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine nozzle
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine main bearing
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine thrust bearing
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine coupling
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine manoeuvre valve
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine drain valve
Propulsion system	Turbine	Steam turbine	Intermediate pressure turbine governor
Propulsion system	Turbine	Steam turbine	Intermediate pressure steam line
Propulsion system	Turbine	Steam turbine	Low pressure turbine
Propulsion system	Turbine	Steam turbine	Low pressure turbine casing
Propulsion system	Turbine	Steam turbine	Low pressure turbine diaphragm
Propulsion system	Turbine	Steam turbine	Low pressure turbine rotor