

Input paper for the following Committee(s):

check as appropriate

- ☐ ARM      ☐ ENG      ☐ PAP  
☒ ENAV      ☐ VTS

Purpose of paper:

- ☐ Input  
☒ Information

Agenda item <sup>2</sup> (from agenda)

n.n

Workplan Task Number / Technical Domain <sup>2</sup>

.....

Working Group

WG 1 Harmonisation

Author(s) / Submitter(s)

Danish Maritime Authority

## Title of paper

### 1 SUMMARY

This document is an update of the input delivered to the IALA ENAV19 Meeting regarding an NW-NM Service (ENAV19-9.14.6). In a total of three Annexes it provides an example of how to use the Draft IALA Guideline on the specification of e-Navigation technical services (ENAV21-9.2).

The three parts of the document is delivered as three different annexes all representing updates to previously submitted material.

### 2 ACTION REQUESTED OF THE COMMITTEE

The Committee is requested to: Take note of the information provided.

---

<sup>1</sup> Input document number, to be assigned by the Committee Secretary

<sup>2</sup> Input papers should be assigned to a work task as listed in the Committee work plan which is available in input papers. Leave open if uncertain but consider how the paper is to be processed if not relevant to a work task

**ANNEX 1 NW-NM Service Specification V0.4**

**ANNEX 2 NW-NM REST Service Technical Design v0.4**

**ANNEX 3 NW-NM Service Specification v0.4**

IALA Working Document

# ANNEX 1



# NW-NM Service Specification v0.4

---

## Contents

1	Introduction.....	4
1.1	NW-NM .....	4
1.2	Purpose of the document.....	4
1.3	Intended readership.....	4
1.4	Inputs from other projects.....	4
2	Service Identification .....	5
3	Operational Context.....	6
3.1	Present Day Operational Context.....	6
3.2	EfficienSea 2 Envisaged Operational Context.....	7
3.3	Functional and Non-functional Requirements .....	7
3.4	Other Constraints .....	8
3.4.1	Relevant Industrial Standards .....	8
3.4.2	Operational Nodes .....	8
3.4.3	Operational Activities .....	8
4	Service Overview.....	9
4.1	Service Interfaces.....	9
5	Service Data Model.....	10
5.1	MessageSeries.....	12
5.2	Chart.....	12
5.3	Area.....	13
5.4	AreaDesc.....	13
5.5	Category .....	13
5.6	CateogryDesc.....	14
5.7	Reference .....	14
5.8	ReferenceDesc .....	14
5.9	Attachment.....	14
5.10	AttachmentDesc .....	15
5.11	DateInterval.....	15
5.12	MessagePart .....	15
5.13	MessagePartDesc .....	16
5.14	GeoJSON .....	16
5.15	Message.....	17
5.16	MessageDesc.....	18
6	Service Interface Specifications.....	19
6.1	Service Interface MessageService .....	19

6.1.1	Operation get() .....	19
7	References.....	21
8	Acronyms and Terminology .....	22
8.1	Acronyms.....	22
8.2	Terminology.....	22
Appendix A	Service Specification XML.....	25

# 1 Introduction

## 1.1 NW-NM

Navigational Warnings (NW) and temporary / preliminary Notice to Mariners (NM T&P) have many similarities and few differences. They largely serve the same purpose, with the main differences being down to the current speed, quality assurance and methods of promulgation.

**Navigational Warnings (NW)** are part of the Maritime Safety Information (MSI) system. Currently, NW's are promulgated in text via SafetyNET, NAVTEX, and is in some countries accessible on the WWW or as voice broadcasts via coastal radio stations.

**Notices to Mariners (NM)** are promulgated weekly in order to keep nautical charts and publications, as far as possible, up to date. Temporary (T) and Preliminary (P) NMs advise mariners of important matters affecting navigational safety, including new hydrographic information (in advance of new editions or chart updates), changes to routing measures and aids to navigation, and other important categories of data. NM T&P's are today promulgated on paper weekly, fortnightly or monthly and are often accessible on the WWW in PDF format. Not all ENC's include T&P information currently.

As part of EfficienSea 2, a combined NW-NM model, and promulgation thereof, is being developed and tested. This involves the specification of an NW-NM service and implementation of an NW-NM service instance that can be integrated with the Maritime Cloud eco system.

The requirements for the NW-NM service is further detailed in the the project document "NW-NM service description and requirements gathering" [2].

## 1.2 Purpose of the document

The purpose of this service specification document is to provide a holistic overview of the NW-NM service and its building blocks in a technology-independent way, according to the guidelines given in the Service Description Guidelines [1].

## 1.3 Intended readership

This service specification is intended to be read by service architects, system engineers and developers in charge of designing and developing an instance of the NW-NM service.

Furthermore, this service specification is intended to be read by enterprise architects, service architects, information architects, system engineers and developers in pursuing architecting, design and development activities of other related services.

## 1.4 Inputs from other projects

An approach to NW-handling and promulgation via AIS was tested during the initial EfficienSea project. The EPD was used to test and evaluate portrayal of NW's on a ECDIS-like device.

This was further developed in the ACCSEAS project, which also developed a combined MSI-NM model and interchange format, see [4], plus an authoring system and promulgation via the Maritime Cloud Messaging Service (please refer to [www.maritimecloud.net](http://www.maritimecloud.net)).

The MSI-NM interchange format devised in the ACCSEAS project was furthermore used as input for IHO, targeting the S-124 NW specification - see [5].

## 2 Service Identification

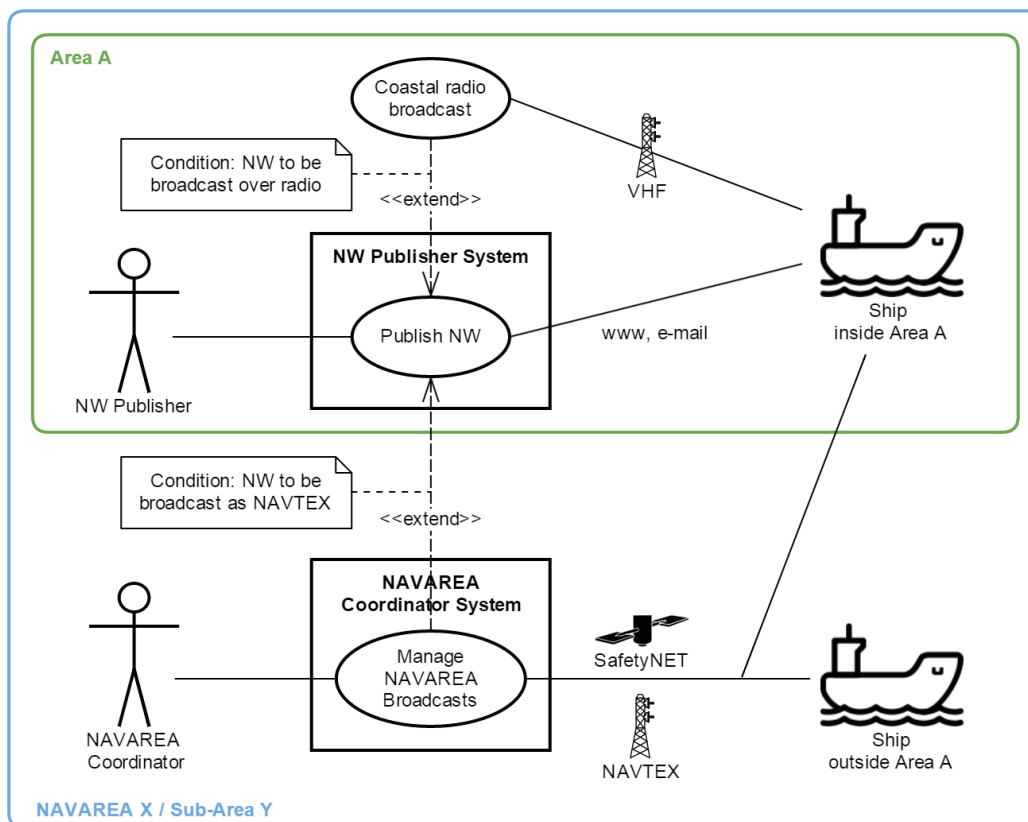
<b>Name</b>	NW-NM T&P Maritime Cloud Service
<b>ID</b>	urn:mrn:mcl:service:specification:dma:nw-nm
<b>Version</b>	0.4
<b>Description</b>	The NW-NM service specification defines a combined NW-NM T&P model along with the actual service API used for accessing NW-NM data, as registered in the Maritime Cloud service catalogue.
<b>Keywords</b>	NW, NM, Navigational Warnings, Notices to Mariners, MSI, Maritime Cloud Service.
<b>Architect(s)</b>	e-Navigation Team Danish Maritime Authority Carl Jacobsens Vej 31 DK-2500 København K Telephone: +45 40 72 61 08 Email: <a href="mailto:mfs@dma.dk">mfs@dma.dk</a>
<b>Status</b>	Released.



## 3 Operational Context

### 3.1 Present Day Operational Context

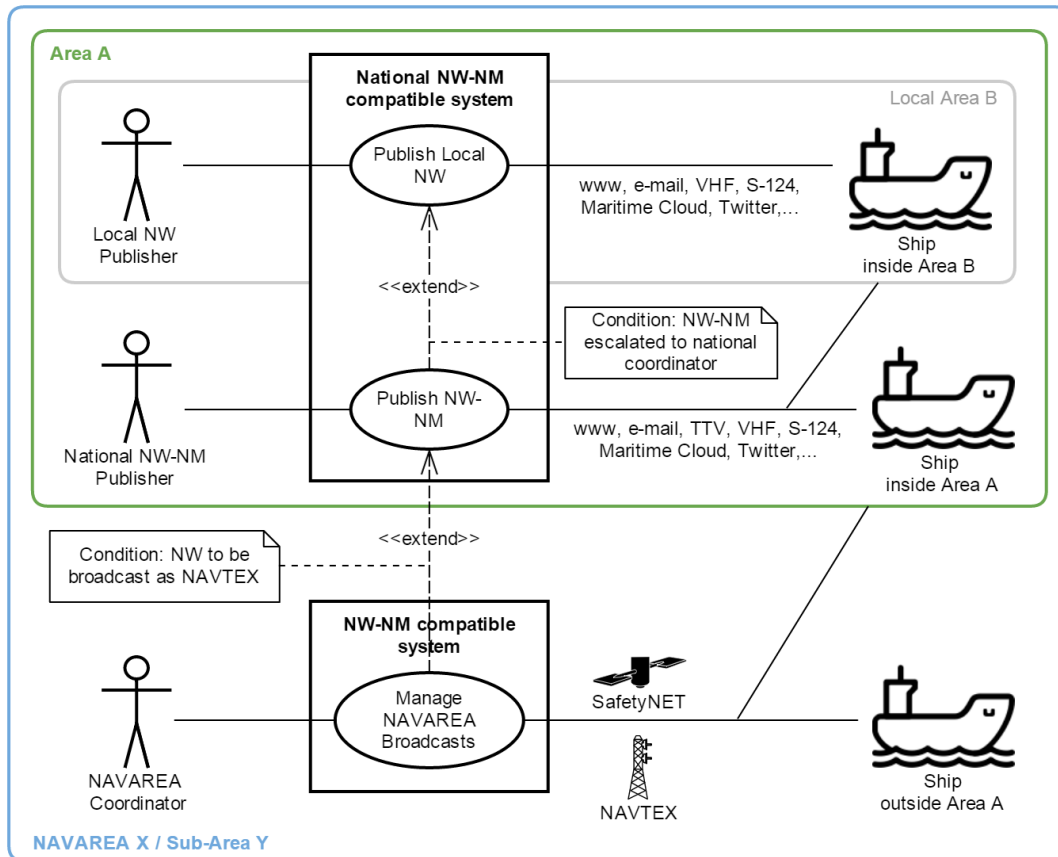
Today's NW broadcast regime, i.e. the operational context of NW promulgation at the component level, is depicted below:



Please note, NM T&P provision is currently via ordinary mail, or downloadable as PDF via a webpage of the NM provider.

System interfaces between NW publishers, NAVAREA (or Sub-Area) coordinator and broadcast service are not standardized, and may rely on manual processes involving e-mail, telephone, voice radio transmissions, fax, telex and manual re-entering of information from one system to another, or much more advanced solutions. Provision of NW or NM via web is not standardized. NAVTEX and SafetyNET cannot transport structured data formats for a joint NW-NM solution.

## 3.2 EfficienSea 2 Envisaged Operational Context



This Scenario depicts an envisaged future NW & NM T&P promulgation regime, as explored in EfficienSea 2.

Based on a standardized structured NW-NM format, compatible NW-NM systems will be able to exchange NW and NM T&P seamlessly. As depicted, each country may have a national NW-NM system, used by local authorities (e.g. harbor and port authorities) as well as the national authorities and agencies (e.g. national maritime safety agencies or hydrographic offices).

Local authorities will administer and publish local NW for their area of responsibility, whereas the national authorities will cater for NW and NM T&P on the national level. Local authorities should have the ability to escalate NW to the national coordinator.

The NW-NM received by ships will thus depend on the promulgation method of choice. If, say, a ship targets the website of a specific port authority; it may see the local NW published by this authority. If, however, the ship query for NW-NM via the Maritime Cloud, it will receive NW-NM from national and local authorities relevant to its current position and planned routes.

The NW-NM Service detailed in this specification only caters for a small part of this promulgation regime. It exposes a single service operation to fetch all currently published (*in force*) NW and NM message from the targeted authority. It may be used by any client, such as a ship, a website or an app.

## 3.3 Functional and Non-functional Requirements

The table below defines additional requirements for the NW-NM service.

Table 1: Requirements Definition

Requirement Id	urn:mrn:mcl:requirement:nw-nm:1
----------------	---------------------------------

Requirement Name	Combined NW-NM model
Requirement Text	The data model should encapsulate a combined NW-NM model.
Rationale	Navigational Warnings (NW) and temporary / preliminary Notice to Mariners (NM T&P) have many similarities and few differences. They largely serve the same purpose, with the main differences being down to the current speed, quality assurance and methods of promulgation

<b>Requirement Id</b>	<b>urn:mrn:mcl:requirement:nw-nm:2</b>
Requirement Name	Return all published NW-NM messages.
Requirement Text	The NW-NM service should make it possible to retrieve all published NW-NM messages from the given service provider.

### 3.4 Other Constraints

#### 3.4.1 Relevant Industrial Standards

The NW-NM model needs to cater for the IHO-IMO-WMO S-53 standard [S-53] on MSI (including NW) and the IHO S-4 standard [S-4] which covers NM T&P.

#### 3.4.2 Operational Nodes

Table 2: Operational Nodes providing the NW-NM service

Operational Node	Remarks
National NW-NM Publisher	Typically, the national maritime safety agencies or hydrographic offices will produce the NW and NM messages and publish via various channels, including the NW-NM Service. Additionally, local authorities (e.g. harbor authorities) may publish local NW's via the national system.
NAVAREA (and Sub-AREA) publishers.	NAVAREA (or Sub-Area) coordinators will receive eligible NW messages from the national agencies and publish these in their area of responsibility.

Table 3: Operational Nodes consuming the NW-NM service

Operational Node	Remarks
Ships, websites and apps	All sorts of clients can be envisaged to consume the published NW-NM messages. Examples may be an ECDIS on a ship, or a safety related sailing App.

#### 3.4.3 Operational Activities

Table 4: Operational Activities supported by the NW-NM service

Operational Activity	Remarks
Showing published NW-NM messages on an ECDIS.	An ECDIS may first look for NW-NM service instances for a relevant area in the Maritime Cloud Service Registry. If one or more NW-NM services instances have been resolved, it will call the operation to retrieve all published NW-NM messages, and display these on an ECDIS.

## 4 Service Overview

### 4.1 Service Interfaces

The NW-NM service consists of a single service, exposing a single operation to query the currently published NW-NM messages (Request/Reply Message Exchange Pattern).

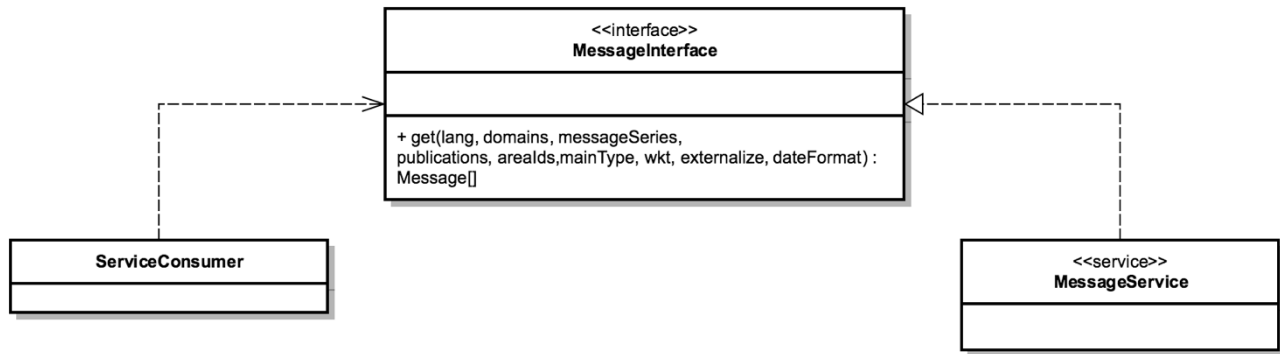


Figure 1: NW-NM Service Definition Diagram

ServiceInterface	Role (from service provider point of view)	ServiceOperation
MessageInterface	Provided	get(lang, domains, messageSeries, publications, arealds, mainType, wkt, externalize, dateFormat)

The returned data model is detailed in chapter 5 and the service operation in chapter 6.

## 5 Service Data Model

This section describes the logical data structures of the NW-NM service. The combined NW-NM model needs to cater for the IHO-IMO-WMO S-53 standard on MSI (including NW) and the IHO S-4 standard which covers NM T&P.

The overarching idea has been to generalize the constituent parts and fields of NW and NM T&P messages, and make the format both backwards compatible and future-proof by e.g. adding support for:

- Multi-language support. All messages must be localizable to any number of languages, including the base data they reference (e.g. areas). The pattern adopted to support this, is to let all classes with localizable attributes (such as *Message*) have an associated list of description entities (*MessageDesc*) which contains a language code and the localizable fields. The description entities are yellow in the UML diagram below.
- Rich text support. NMs in particular, can contain a rich layout containing features such as tables, links, embedded pictograms, etc. By supporting HTML descriptions this can be accommodated.
- New identifier format. The S-4 and S-53 standards loosely specifies a numbering scheme for NWs and NMs. However, the numbering scheme does not guarantee uniqueness in a combined NW-NM model, let alone a system that may contain messages from multiple authorities. Thus, the NW-NM data model introduces *message series* and adds a unique MRN (maritime resource name) to each message.
- Base data. Part of a combined NW-NM model is to define a relationship between messages and base data such as charts, categories and areas. Previous proposals have opted for rigid solutions with a fixed number of area and category levels, and with enumerated category values.
- S-124 compatibility. The IHO S-124 specification for Navigational Warnings has not been released yet, but an aim of the NW-NM model has been to be relatively future-compatible with the S-124 data format, to make it easy to exchange data between the two formats.

The UML detailing the Message class, and its related classes, is given below:

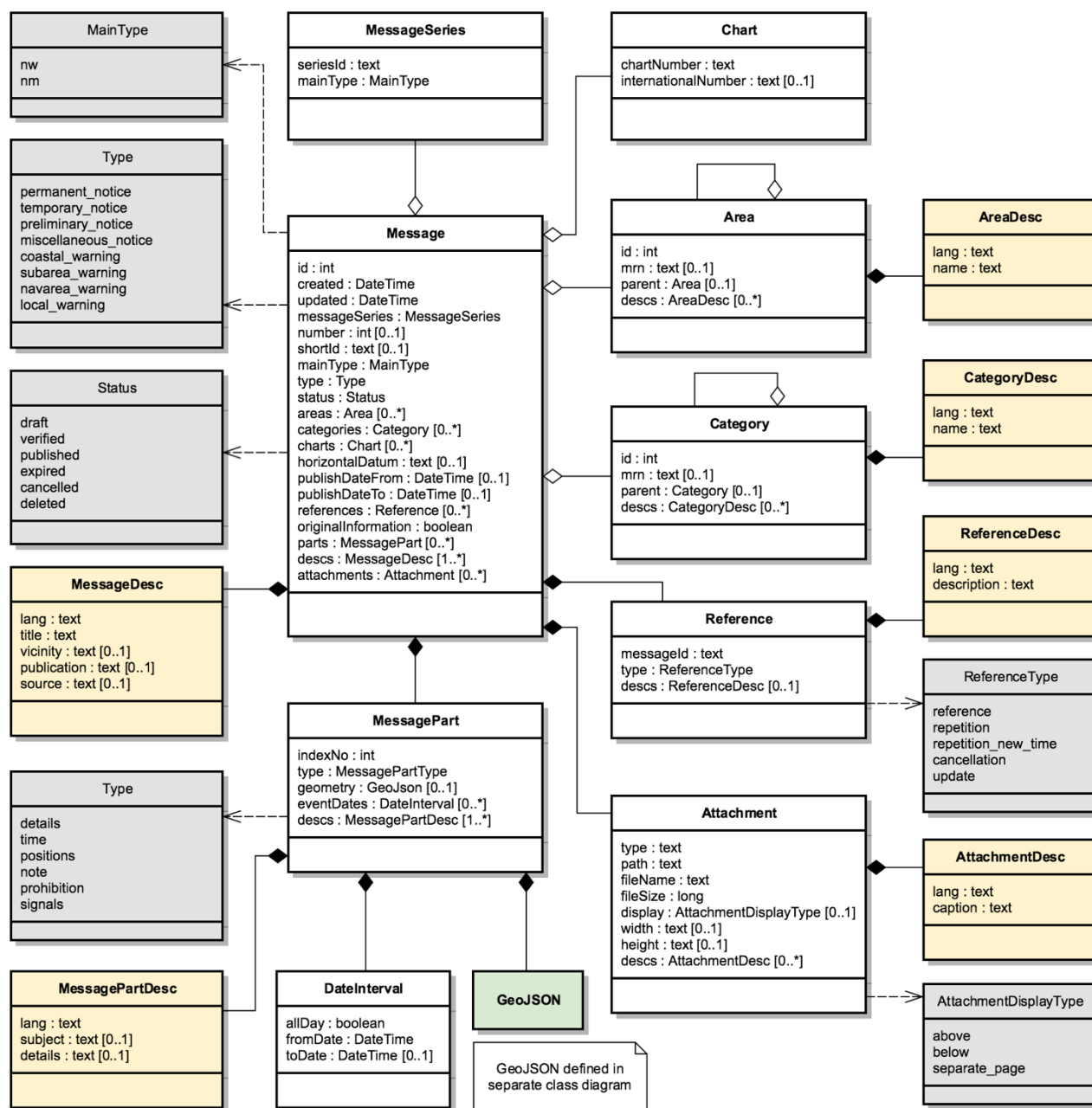


Figure 2: Message UML class diagram

The diagram uses the following colour codes:

- Light-gray background: Used for enumerations.
- Light-yellow background: Used for localized description entities - see *Design Pattern* section below.
- Light-green background: The *GeoJSON* model is expanded below:

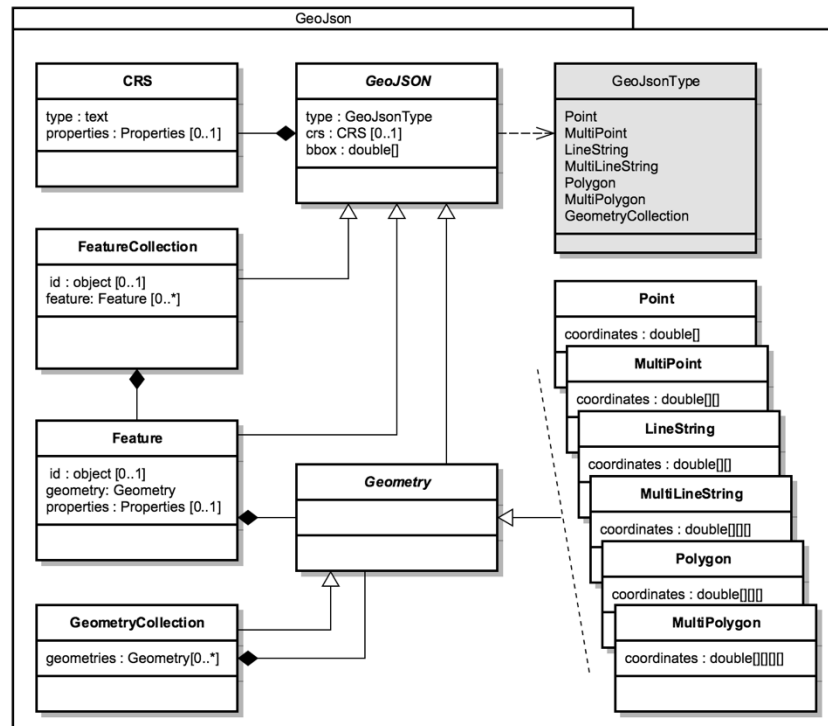


Figure 3: GeoJSON UML class diagram

The use of aggregation vs composition connectors above is mostly academic, since the UML is not a database model but merely an interchange format. However, the aggregation connector is used to signal that the associated entity represents base data in the producing system, and is not tied to the life cycle of the Message.

The remainder of the chapter will detail the individual classes.

## 5.1 MessageSeries

According to IHO, NW and NM messages must be numbered. For NW, it is e.g. mandated that:

*Navigational warnings in each series should be consecutively numbered throughout the calendar year, commencing with 1/YY at 0000 UTC on 1 January.*

The numbering scheme does not guarantee uniqueness in a combined NW-NM model, let alone a system that may contain messages from multiple countries and authorities. Thus, *message series* have been introduced in the NW-NM data model to group messages as appropriate. A country may e.g. have separate message series for NW and NM. However, they may also introduce separate message series to allow, say, local harbour authorities to maintain their own message series for local NWs, or, as is the case with Canada, divide the country into five regions, each with their own message series.

Attribute Name	Type	Description
seriesId	String	The ID of the message series in the implementing system. Should be globally unique
mainType	MainType	Either NW or NM.

## 5.2 Chart

A message can be assigned a list of *charts*. The charts are maintained administratively as base data in the producing system.

Attribute Name	Type	Description
chartNumber	String	Mandatory regional chart number (and identifier).
internationalNumber	String	Optional international chart number.

### 5.3 Area

Existing IHO standards for NW and NM both provide support for specifying multiple area levels (*general area* and *locality* for NW; *general region*, *sub-region* and *specific location* for NMs). However, in the NW-NM system, this has been generalized, and areas are administratively maintained in a hierarchical area tree (with each area having a localized name) of arbitrary depth. A message can be assigned a list of these areas, and by implication, the parent areas of the selected area. Additionally, a message can be assigned a localized textual *vicinity* description (part of the *MessageDesc* class – see 5.16), for detailed location information not defined in the area tree.

Attribute Name	Type	Description
id	Int	Internal system ID of the area
mrn	String	Optionally, an area may be assigned a globally unique MRN (maritime resource name). Adopting MRNs for areas would make interchange of message data between two NW-NM systems more robust.
parent	Area	Non-root areas will reference their parent areas. Example: Randers Havn -> Kattegat -> Danmark
descs	AreaDesc[]	The list of localizable attributes for an area. See 5.4.

### 5.4 AreaDesc

The AreaDesc class contains the list of localizable attributes for an area.

Attribute Name	Type	Description
lang	String	The ISO 639-1 language code.
name	String	The localized name of an area.

### 5.5 Category

Categories are administratively maintained in a hierarchical category tree (with each category having a localized name) of arbitrary depth. A message can be assigned a list of these categories, and by implication, the parent categories of a selected category.

At the top level, the categories will have entries such as *Aids to Navigation*, *Drifting Objects*, *Obstruction*, etc., which is the categorization used in the IHO standards. The sub-categories will represent the types of hazard relevant to the parent category. Examples of category lineages (top-down):

- Buoy -> Buoy Established
- Obstruction -> Wreck -> Marked Wreck

Attribute Name	Type	Description
id	Int	Internal system ID of the category
mrn	String	Optionally, a category may be assigned a globally unique MRN (maritime resource name). Adopting MRNs for categories would make interchange of message data between two NW-NM systems more robust.
parent	Category	Non-root category will reference their parent categories.



Attribute Name	Type	Description
descs	CategoryDesc[]	The list of localizable attributes for a category. See 5.6.

## 5.6 CategoryDesc

The CategoryDesc class contains the list of localizable attributes for a category.

Attribute Name	Type	Description
lang	String	The ISO 639-1 language code.
name	String	The localized name of a category.

## 5.7 Reference

The Reference class provides a typed, weak reference to a message.

Attribute Name	Type	Description
messageId	String	An identifier of the referenced message. If the messageId is recognized to be a <i>short-ID</i> of another message, it can e.g. be used to hyperlink to that message. However there are no requirements as to the format of the message ID.
type	ReferenceType	The type of the reference. One of the values "reference", "repetition", "repetition_new_time", "cancellation" or "update".
descs	ReferenceDesc[]	The list of localizable attributes for a Reference. See 5.8.

## 5.8 ReferenceDesc

The ReferenceDesc class contains the list of localizable attributes for a reference.

Attribute Name	Type	Description
lang	String	The ISO 639-1 language code.
description	String	The localized name of a reference.

## 5.9 Attachment

Messages can be associated with a list of attachments, such as images, PDF-files, etc. The physical attachment files will reside in a public repository on the producing system.

Attribute Name	Type	Description
type	String	The content type of the attachment file, such as "image/png".
path	String	The URL path to the attachment file in the producing system.
filename	String	The file name of the attachment.
fileSize	Long	The size of the attachment in bytes.
display	AttachmentDisplayType	If defined, this flag can be used to signal how the editor intended for the (image or video) attachment to be displayed when rendering the message for the end user. "above" and "below" signals that the attachment should be displayed above, respectively below, the message details. "separate_page" signals that the attachment

Attribute Name	Type	Description
		should be displayed on a separate page if rendered in paged media, such as a PDF file.
width	String	The width to use when displaying the (image or video) attachment. The width must include the type (i.e. <i>em</i> , <i>px</i> , <i>%</i> , <i>cm</i> , <i>mm</i> , <i>in</i> , <i>pt</i> or <i>pc</i> ). If the <i>height</i> attribute is left unspecified, the attachment should be scaled proportionally.
height	String	The height to use when displaying the (image or video) attachment. The height must include the type (i.e. <i>em</i> , <i>px</i> , <i>%</i> , <i>cm</i> , <i>mm</i> , <i>in</i> , <i>pt</i> or <i>pc</i> ). If the <i>width</i> attribute is left unspecified, the attachment should be scaled proportionally.
Descs	AttachmentDesc[]	The list of localizable attributes for an Attachment. See 5.10.

## 5.10 AttachmentDesc

The AttachmentDesc class contains the list of localizable attributes for an attachment.

Attribute Name	Type	Description
lang	String	The ISO 639-1 language code.
caption	String	A localized caption to display for the attachment.

## 5.11 DateInterval

A message will have an associated list of (possibly open-ended) date-time intervals, as defined by the DateInterval class. This defines the period of time for which the hazard described by the message applies.

Additionally, a message can be assigned a localized textual *time* description (part of the *MessageDesc* class – see 5.16).

Attribute Name	Type	Description
allDay	Boolean	If the <i>allDay</i> flag is set, the <i>fromDate/toDate</i> attributes should be treated as dates without a time-part by the producing system.
fromDate	DateTime	The start date-time of a date interval.
toDate	DateTime	An optional end date-time of a date interval.

## 5.12 MessagePart

A message defines an ordered list of *message parts*, which can be thought of as sub-stories.

Conceptually, each message part defines the time, positions, key subject and description of the hazard or event that the story pertains to.

Attribute Name	Type	Description
indexNo	Int	Specified the ascending index of the message part within the message.
type	MessagePartType	May be used by the client to tag the message tag details with a type. Valid types are "details", "time", "positions", "note", "prohibition" and "signals".
geometry	GeoJSON	The positions of the message part. The GeoJSON

Attribute Name	Type	Description
		type is treated in more details in 5.14.. In practice, Niord will always return a <i>FeatureCollection</i> GeoJSON entity..
eventDates	DateTime[]	The list of event dates for which the message part hazard pertains. The list should not be rendered for the end user by the client, but may be used for computations.
descs	MessagePartDesc[]	The list of localizable attributes for a MessagePart. See 5.13.

### 5.13 MessagePartDesc

The MessagePartDesc class contains the list of localizable attributes for a message part.

Attribute Name	Type	Description
lang	String	The ISO 639-1 language code.
subject	String	The key subject of the hazard or event that the message part pertains to.
details	String	A detailed description of the hazard or event that the message part pertains to. The type of the <i>details</i> field is mandated to be HTML, and thus allows for fairly advanced layout and typography, and may contain elements such as tables, links, images, etc.

### 5.14 GeoJSON

The GeoJSON package is an implementation of an external data model, as defined at <http://geojson.org/geojson-spec.html>. The classes will not be detailed in this chapter.

GeoJSON was picked as the representation of a message geometry, because it is widely adopted by client libraries, and, unlike e.g. WKT, the GeoJSON Feature class has associated properties that can be used to store various information, that may be used in the portrayal of the messages.

The NW-NM system thus defines the following GeoJSON Feature properties, that a client may use for improved portrayal (but is not mandated to do so):

Property Name	Description
name:<<lang>>	Contains a language specific name that can be displayed for the geometry of the entire Feature. Example: name:en = Area of reduced depth.
name:<<x>>:<<lang>>	Contains a language specific name that can be displayed for the x'th coordinate of the geometry of the Feature. Example: name:12:en = yellow spar buoy with topmark.
parentFeatureId radius radiusType	These properties are mostly used by the producing system to let a geometry be defined from another geometry. As an example, an <i>affected area</i> may be defined as a buffered geometry with a radius of 200 meters around, say, the position of a wreck (this being the parent geometry). Clients may choose to adjust the portrayal of Features containing these properties, to signal that the feature represents an affected areas, and not the hazard itself.

## 5.15 Message

The Message class represents either an NW or an NM message.

It has been a deliberate, if slightly controversial, choice to let NWs and NMs share the same Message class, rather than having a separate sub-class for each type. The main rationale for this is that NWs and NMs T&P are expected to converge in the future, once promulgation is handled completely via electronic means.

Attribute Name	Type	Description
id	Int	Internal system ID of the message
created	DateTime	The timestamp the message was created in the system.
updated	DateTime	The timestamp the message was last updated in the system.
messageSeries	MessageSeries	The message series of the message. See 5.1.
number	Int	The sequence number of a published message. See 5.1.
shortId	String	The short-ID of a published message. See 5.1.
mainType	MainType	The main type of the message, either NW or NM. In reality, this attribute is redundant, since the main type is also defined by the associated message series, and may be implied by the message type.
type	Type	The sub-type of the message. One of: <ul style="list-style-type: none"> <li>• permanent_notice</li> <li>• temporary_notice</li> <li>• preliminary_notice</li> <li>• miscellaneous_notice</li> <li>• coastal_warning</li> <li>• subarea_warning</li> <li>• navarea_warning</li> <li>• local_warning</li> </ul>
status	Status	<p>The status of the message. One of:</p> <ul style="list-style-type: none"> <li>• draft:</li> <li>• verified</li> <li>• published</li> <li>• expired</li> <li>• cancelled</li> <li>• deleted</li> </ul> <p>It is up to the producing system to manage the message life cycle, and enforce rules, such as:</p> <ul style="list-style-type: none"> <li>• Messages are created as drafts.</li> <li>• They can be assigned a secondary draft status, verified, to signal that they are ready for publication.</li> <li>• Only draft and verified messages can be deleted or published.</li> <li>• A published message can be cancelled manually or expired by the system if all associated date-intervals have passed.</li> </ul>
areas	Area[]	A list of the areas of a message. See 5.3.
categories	Category[]	A list of the categories of a message. See 5.5.
charts	Chart[]	A list of the charts of a message. See 5.2.

Attribute Name	Type	Description
horizontalDatum	String	The horizontal datum for the message. If unspecified, assume WGS-84.
publishDateFrom	DateTime	The timestamp for when the message was published – or should be published.
publishDateTo	DateTime	The timestamp for when the message was cancelled – or should be expired.
references	Reference[]	A list of message references. See 5.7.
originalInformation	Boolean	If the message was based on original information or not.
parts	MessagePart[]	The list of message parts of the message. See 5.12.
descs	MessageDesc[]	The list of localizable attributes for a Message. See 5.16.
attachments	Attachment[]	The list of message attachments of the message. See 5.9.

## 5.16 MessageDesc

The MessageDesc class contains the list of localizable attributes for a message.

Attribute Name	Type	Description
lang	String	The ISO 639-1 language code.
title	String	A complete title line to show for a message. Typically composed by concatenating the area lineage, vicinity and subject of a message. Example: “Denmark. The North Sea. Hanstholm SW. AIS buoyage established.”
vicinity	String	May be used for localized arbitrary area information not defined in the area tree. See 5.3.
publication	String	A textual listing of all the publications relevant to the message. The format of the publication field is mandated to be HTML, as it may contain links to the actual publications.
source	String	The source of the message hazard information.

## 6 Service Interface Specifications

This chapter describes the details of the NW-NM service interface.

### 6.1 Service Interface MessageService

The NW-NM service is comprised of a single service (MessageService) and operation, which follows the Request/Reply Message Exchange Pattern. Sequence diagram:

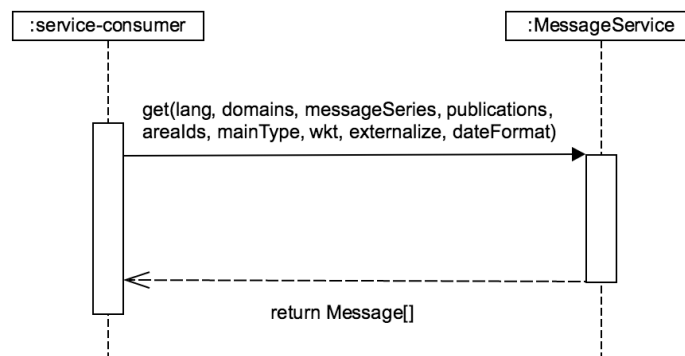


Figure 1: NW-NM Service Sequence Diagram

#### 6.1.1 Operation get()

The get() operation returns the list of all published NW and NM messages. The returned result can be controlled using the following parameters:

Parameter	Type	Description
lang	String	An optional ISO 639-1 language code. If specified, only this language variant is returned for localized entities such as Message, Area and Category. However, if, say, “en” is requested and an entity only has a “da” language description entity, then this is returned instead. A client may want to flag this to the end user.
domains	String[]	List of domain ID’s. Each Niord system organizes messages into domains, e.g. for Navigational Warnings, Notices to Mariners, Firing Exercises, Almanacs, etc.
messageSeries	String[]	List of message series ID’s. Each message is associated with a message series, as defined by the producing Niord system. As an example, there may be separate messages series for local navigational warnings and other navigational warnings.
publications	String[]	Some messages may have been published in <i>publications</i> , such as a weekly NtM. The <i>publication</i> parameter can be used to filter by specific publications.
arealds	String[]	Messages may be associated with one or more area, as defined by the producing Niord System. The <i>areald</i> parameter can be used to filter messages by areas using either the internal ID or the MRN of the areas.
mainType	String	Used to filter messages by their main type, i.e. "NM" or

Parameter	Type	Description
		"NW".
wkt	String	Optional Well-Known Text representation of a geometry. If specified, only messages intersecting the geometry will be returned.
externalize	Boolean	Whether to rewrite all embedded links and paths to be absolute URL's pointing back to their repository URL in the producing system. Default value is <i>true</i> .
dateFormat	String	The date format to use for JSON date-time encoding. Either <i>UNIX_EPOCH</i> (default) or <i>ISO_8601</i> .
Return	Type	Description
	Message[]	The list of matching published NW and NM.

## 7 References

Nr.	Reference
[1] Service Description Guidelines	E2_Deliverable D3.4 – Service Specification Template, version 1.0
[2] NW-NM Service Description	E2 – 3.1.3 NW-NM service description and requirements gathering, version 0.1
[3] Maritime Cloud	Documented at <a href="http://www.maritimecloud.net">www.maritimecloud.net</a>
[4] MSI-NM in the ACCSEAS project	ACCSEAS MSI-NM Annex, version 1.0
[5] MSI-NM S-100 proposal	ACCSEAS MSI-NM S-100 Product Specification input paper, version 1.0.
[S-4]	Regulations of the IHO for International Charts and Chart Specifications of the IHO. Edition 4.3.0, August 2012, International Hydrographic Bureau, Monaco.
[S-53]	Manual on Maritime Safety Information (MSI). Special Publication No. 53, July 2009 Edition. International Hydrographic Bureau, Monaco.
[S-100]	Universal Hydrographic Data Model. IHO Special Publication No. S-100, Edition 1.0.0, January 2010. International Hydrographic Bureau, Monaco.



## 8 Acronyms and Terminology

### 8.1 Acronyms

Term	Definition
<b>API</b>	Application Programming Interface
<b>DMA</b>	Danish Maritime Authority
<b>ECDIS</b>	Electronic Chart Display Information Systems
<b>ENC</b>	Electronic Navigational Chart
<b>EPD</b>	e-Navigation Prototype Display
<b>IHO</b>	International Hydrographic Organisation
<b>MC</b>	Maritime Cloud
<b>MEP</b>	Message Exchange Pattern
<b>MRN</b>	Maritime Resource Name
<b>MSI</b>	Maritime Safety Information
<b>NAF</b>	NATO Architectural Framework
<b>NM</b>	Notices to Mariners
<b>NW</b>	Navigational Warning
<b>REST</b>	Representational State Transfer
<b>SOAP</b>	Simple Object Access Protocol
<b>SSD</b>	Service Specification Document
<b>UML</b>	Unified Modelling Language
<b>URL</b>	Uniform Resource Locator
<b>VTs</b>	Vessel Traffic Service
<b>WSDL</b>	Web Service Definition Language
<b>XML</b>	Extendible Mark-up Language
<b>XSD</b>	XML Schema Definition

### 8.2 Terminology

Term	Definition
<b>External Data Model</b>	Describes the semantics of the “maritime world” (or a significant part thereof) by defining data structures and their relations. This could be at logical level (e.g., in UML) or at physical level (e.g., in XSD schema definitions), as for example standard data models, or S-100 based data produce specifications.
<b>Navigational Warnings</b>	Navigational Warnings (NW) are part of the Maritime Safety Information (MSI) system. Currently, NW’s are promulgated in text via SafetyNET, NAVTEX, and is in some countries accessible on the WWW or as voice broadcasts via coastal radio stations.
<b>Niord</b>	Anglified name of the Norse God Njord, associated with the sea and seafaring. Also the name of the EfficienSea 2 sub-project implementation of the NW-NM service and authoring system. See <a href="http://niord.org">http://niord.org</a>
<b>Notices to Mariners</b>	Notices to Mariners (NM) are promulgated weekly in order to keep nautical charts and publications, as far as possible, up to

	date. Temporary (T) and Preliminary (P) NMs advise mariners of important matters affecting navigational safety, including new hydrographic information (in advance of new editions or chart updates), changes to routing measures and aids to navigation, and other important categories of data. NM T&P's are today promulgated on paper weekly, fortnightly or monthly and are often accessible on the WWW in PDF format. Not all ENC's include T&P information currently.
<b>Operational Activity</b>	An activity performed by an operational node. Examples of operational activities in the maritime context are: Route Planning, Route Optimization, Logistics, Safety, Weather Forecast Provision, ...
<b>Operational Model</b>	A structure of operational nodes and associated operational activities and their inter-relations in a process model.
<b>Operational Node</b>	A logical entity that performs activities. Note: nodes are specified independently of any physical realisation. Examples of operational nodes in the maritime context are: Maritime Control Center, Maritime Authority, Ship, Port, Weather Information Provider, ...
<b>Service</b>	The contractual provision of something (a non-physical object), by one, for the use of one or more others. Services involve interactions between providers and consumers, which may be performed in a digital form (data exchanges) or through voice communication or written processes and procedures.
<b>Service Consumer</b>	A service consumer uses service instances provided by service providers. All users within the maritime domain can be service customers, e.g., ships and their crew, authorities, VTS stations, organizations (e.g., meteorological), commercial service providers, etc.
<b>Service Data Model</b>	Formal description of one dedicated service at logical level. The service data model is part of the service specification. Is typically defined in UML and/or XSD. If an external data model exists (e.g., a standard data model), then the service data model shall refer to it: each data item of the service data model shall be mapped to a data item defined in the external data model.
<b>Service Implementer</b>	Implementers of services from the service provider side and/or the service consumer side. Everybody can be a service implementer but mainly this will be commercial companies implementing solutions for shore and ship.
<b>Service Instance</b>	The implementation of a dedicated service in a dedicated technology. One service specification may result in several service instances, being implemented with different or same technologies.
<b>Service Instance Description</b>	Documents the details of a service instance (most likely documented by the service implementer). The service instance description includes (but is not limited to) a service instance model and describes the used technology, transport mechanism, quality of service, etc.
<b>Service Instance</b>	Describes the implementation of a dedicated service instance in

<b>Model</b>	<p>a dedicated technology. This includes a detailed description of the data payload to be exchanged by this service instance. The actual format of the service instance model depends on the chosen technology. Examples may be WSDL and XSD files (e.g., for SOAP services) or swagger (Open API) specifications (e.g., for REST services). If an external data model exists (e.g., a standard data model), then the service instance model shall refer to it: each data item of the service instance model shall be mapped to a data item defined in the external data model.</p> <p>In order to prove correct implementation of the service specification, there shall exist a mapping between the service instance model and the service data model. This means, each data item used in the service instance model shall be mapped to a corresponding data item of the service data model. (In case of existing mappings to a common external (standard) data model from both the service data model and the service instance model, such a mapping is implicitly given.)</p>
<b>Service Interface</b>	The mechanism by which a service communicates.
<b>Service Provider</b>	A service provider provides instances of services according to a service specification and service instance description. All users within the maritime domain can be service providers, e.g., authorities, VTS stations, organizations (e.g., meteorological), commercial service providers, etc.
<b>Service Specification</b>	Describes one dedicated service at logical level. The Service Specification is technology-agnostic. The Service Specification includes (but is not limited to) a description of the Service Interfaces and Service Operations with their data payload. The data payload description may be formally defined by a Service Data Model.
<b>Service Specification Producer</b>	Producers of service specifications in accordance with the service description guidelines.
<b>Service Technology Catalogue</b>	List and specifications of allowed technologies for service implementations. Currently, SOAP and REST are envisaged to be allowed service technologies. The service technology catalogue shall describe in detail the allowed service profiles, e.g., by listing communication standards, security standards, stacks, bindings, etc.

## Appendix A Service Specification XML

```
<?xml version="1.0" encoding="UTF-8"?>
<serviceSpecification
  xmlns="http://efficiensea2.org/maritime-cloud/service-registry/v1/ServiceSpecificationSchema.xsd"
  xmlns:ServiceSpecificationSchema="http://efficiensea2.org/maritime-cloud/service-
registry/v1/ServiceSpecificationSchema.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://efficiensea2.org/maritime-cloud/service-
registry/v1/ServiceSpecificationSchema.xsd ServiceSpecificationSchema.xsd "
  xmlns:xs="http://www.w3.org/2001/XMLSchema" >

  <name>NW-NM TP Maritime Cloud Service</name>
  <status>released</status>
  <id>urn:mrn:mcl:service:specification:dma:nw-nm</id>
  <version>0.4</version>
  <description>The NW-NM service specification defines a combined NW-NM TP model along with the actual
service API used for accessing NW-NM data, as registered in the Maritime Cloud service
catalogue.</description>

  <keywords>NW, NM, Navigational Warnings, Notices to Mariners, MSI, Maritime Cloud Service</keywords>
  <isSpatialExclusive>false</isSpatialExclusive>

  <authorInfos>
    <authorInfo>
      <id>urn:mrn:mcl:user:dma:mfs</id>
      <name>Mads Friis Sørensen</name>
      <description>Responsible for the NW-NM service</description>
      <contactInfo>mfs@dma.dk</contactInfo>
    </authorInfo>
  </authorInfos>

  <requirements>
    <requirement>
      <id>urn:mrn:mcl:requirement:nw-nm:1</id>
      <name>Combined NW-NM model</name>
      <text>The data model should encapsulate a combined NW-NM model.</text>
    </requirement>
    <requirement>
      <id>urn:mrn:mcl:requirement:nw-nm:2</id>
      <name>Return all published NW-NM messages.</name>
      <text>The NW-NM service should make it possible to retrieve all published NW-NM messages from
the given service provider.</text>
    </requirement>
  </requirements>

  <serviceDataModel>
    <definitionAsXSD>

      <xs:schema version="1.0" xmlns:xs="http://www.w3.org/2001/XMLSchema">

        <xs:element name="area" type="areaVo"/>

        <xs:element name="attachment" type="attachmentVo"/>

        <xs:element name="category" type="categoryVo"/>

        <xs:element name="chart" type="chartVo"/>

        <xs:element name="feature" type="featureVo"/>

        <xs:element name="featureCollection" type="featureCollectionVo"/>

        <xs:element name="geometryCollection" type="geometryCollectionVo"/>

        <xs:element name="lineString" type="lineStringVo"/>

      </xs:schema>

    </definitionAsXSD>
  </serviceDataModel>
</serviceSpecification>
```

```

<xs:element name="message" type="messageVo"/>

<xs:element name="messageSeries" type="messageSeriesVo"/>

<xs:element name="multiLineString" type="multiLineStringVo"/>

<xs:element name="multiPoint" type="multiPointVo"/>

<xs:element name="multiPolygon" type="multiPolygonVo"/>

<xs:element name="point" type="pointVo"/>

<xs:element name="polygon" type="polygonVo"/>

<xs:complexType name="doubleArray" final="#all">
  <xs:sequence>
    <xs:element name="item" type="xs:double" minOccurs="0" maxOccurs="unbounded"
nillable="true"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="doubleArrayArray" final="#all">
  <xs:sequence>
    <xs:element name="item" type="doubleArray" minOccurs="0" maxOccurs="unbounded"
nillable="true"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="doubleArrayArrayArray" final="#all">
  <xs:sequence>
    <xs:element name="item" type="doubleArrayArray" minOccurs="0" maxOccurs="unbounded"
nillable="true"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="messageVo">
  <xs:sequence>
    <xs:element ref="messageSeries" minOccurs="0"/>
    <xs:element name="number" type="xs:int" minOccurs="0"/>
    <xs:element name="shortId" type="xs:string" minOccurs="0"/>
    <xs:element name="mainType" type="mainType" minOccurs="0"/>
    <xs:element name="type" type="type" minOccurs="0"/>
    <xs:element name="status" type="status" minOccurs="0"/>
    <xs:element name="areas" type="areaVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="categories" type="categoryVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="charts" type="chartVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="horizontalDatum" type="xs:string" minOccurs="0"/>
    <xs:element name="publishDateFrom" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="publishDateTo" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="followUpDate" type="xs:dateTime" minOccurs="0"/>
    <xs:element name="references" type="referenceVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="originalInformation" type="xs:boolean" minOccurs="0"/>
    <xs:element name="parts" type="messagePartVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="descs" type="messageDescVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="attachments" type="attachmentVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="created" type="xs:dateTime"/>
  <xs:attribute name="id" type="xs:string"/>
  <xs:attribute name="updated" type="xs:dateTime"/>
</xs:complexType>

<xs:complexType name="messageSeriesVo">
  <xs:sequence>
    <xs:element name="seriesId" type="xs:string" minOccurs="0"/>
    <xs:element name="mainType" type="mainType" minOccurs="0"/>

```

```

    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="areaVo">
    <xs:sequence>
      <xs:element name="mrn" type="xs:string" minOccurs="0"/>
      <xs:element name="active" type="xs:boolean"/>
      <xs:element name="parent" type="areaVo" minOccurs="0"/>
      <xs:element name="descs" type="areaDescVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int"/>
  </xs:complexType>

  <xs:complexType name="areaDescVo">
    <xs:sequence>
      <xs:element name="name" type="xs:string" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="lang" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="categoryVo">
    <xs:sequence>
      <xs:element name="mrn" type="xs:string" minOccurs="0"/>
      <xs:element name="active" type="xs:boolean"/>
      <xs:element name="parent" type="categoryVo" minOccurs="0"/>
      <xs:element name="descs" type="categoryDescVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:int"/>
  </xs:complexType>

  <xs:complexType name="categoryDescVo">
    <xs:sequence>
      <xs:element name="name" type="xs:string" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="lang" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="chartVo">
    <xs:sequence>
      <xs:element name="chartNumber" type="xs:string" minOccurs="0"/>
      <xs:element name="internationalNumber" type="xs:int" minOccurs="0"/>
      <xs:element name="active" type="xs:boolean"/>
      <xs:element name="scale" type="xs:int" minOccurs="0"/>
      <xs:element name="name" type="xs:string" minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="referenceVo">
    <xs:sequence>
      <xs:element name="messageId" type="xs:string" minOccurs="0"/>
      <xs:element name="type" type="referenceType" minOccurs="0"/>
      <xs:element name="descs" type="referenceDescVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>

  <xs:complexType name="referenceDescVo">
    <xs:sequence>
      <xs:element name="description" type="xs:string" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="lang" type="xs:string"/>
  </xs:complexType>

  <xs:complexType name="messagePartVo">
    <xs:sequence>
      <xs:element name="indexNo" type="xs:int"/>
      <xs:element name="type" type="messagePartType" minOccurs="0"/>
      <xs:element name="eventDates" type="dateIntervalVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
      <xs:element name="geometry" type="featureCollectionVo" minOccurs="0"/>

```

```

        <xs:element name="descs" type="messagePartDescVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="hideSubject" type="xs:boolean" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="dateIntervalVo">
    <xs:sequence>
        <xs:element name="allDay" type="xs:boolean" minOccurs="0"/>
        <xs:element name="fromDate" type="xs:dateTime" minOccurs="0"/>
        <xs:element name="toDate" type="xs:dateTime" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="featureCollectionVo">
    <xs:sequence>
        <xs:element name="bbox" type="xs:double" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="crs" type="crs" minOccurs="0"/>
        <xs:element ref="feature" minOccurs="0" maxOccurs="unbounded"/>
        <xs:element name="id" type="xs:anyType" minOccurs="0"/>
        <xs:element name="type" type="xs:string" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="crs">
    <xs:sequence>
        <xs:element name="properties">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="entry" minOccurs="0" maxOccurs="unbounded">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:element name="key" minOccurs="0" type="xs:anyType"/>
                                <xs:element name="value" minOccurs="0" type="xs:anyType"/>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="type" type="xs:string" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="featureVo">
    <xs:sequence>
        <xs:element name="bbox" type="xs:double" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="crs" type="crs" minOccurs="0"/>
        <xs:choice>
            <xs:element ref="point"/>
            <xs:element ref="multiPoint"/>
            <xs:element ref="lineString"/>
            <xs:element ref="multiLineString"/>
            <xs:element ref="polygon"/>
            <xs:element ref="multiPolygon"/>
            <xs:element ref="geometryCollection"/>
        </xs:choice>
        <xs:element name="id" type="xs:anyType" minOccurs="0"/>
        <xs:element name="properties">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="entry" minOccurs="0" maxOccurs="unbounded">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:element name="key" minOccurs="0" type="xs:string"/>
                                <xs:element name="value" minOccurs="0" type="xs:anyType"/>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
    </xs:sequence>

```



```

        </xs:complexType>
        </xs:element>
        <xs:element name="type" type="xs:string" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="geometry" abstract="true">
    <xs:sequence>
        <xs:element name="bbox" type="xs:double" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
        <xs:element name="crs" type="crs" minOccurs="0"/>
        <xs:element name="type" type="xs:string" minOccurs="0"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="pointVo">
    <xs:complexContent>
        <xs:extension base="geometry">
            <xs:sequence>
                <xs:element name="coordinates" type="xs:double" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:complexType name="multiPointVo">
    <xs:complexContent>
        <xs:extension base="geometry">
            <xs:sequence>
                <xs:element name="coordinates" type="doubleArray" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:complexType name="lineStringVo">
    <xs:complexContent>
        <xs:extension base="geometry">
            <xs:sequence>
                <xs:element name="coordinates" type="doubleArray" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:complexType name="multiLineStringVo">
    <xs:complexContent>
        <xs:extension base="geometry">
            <xs:sequence>
                <xs:element name="coordinates" type="doubleArrayArray" nillable="true"
minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:complexType name="polygonVo">
    <xs:complexContent>
        <xs:extension base="geometry">
            <xs:sequence>
                <xs:element name="coordinates" type="doubleArrayArray" nillable="true"
minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:complexType name="multiPolygonVo">
    <xs:complexContent>

```



```

        <xs:extension base="geometry">
            <xs:sequence>
                <xs:element name="coordinates" type="doubleArrayArrayArray" nillable="true"
minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:complexType name="geometryCollectionVo">
    <xs:complexContent>
        <xs:extension base="geometry">
            <xs:sequence>
                <xs:choice minOccurs="0" maxOccurs="unbounded">
                    <xs:element ref="point"/>
                    <xs:element ref="multiPoint"/>
                    <xs:element ref="lineString"/>
                    <xs:element ref="multiLineString"/>
                    <xs:element ref="polygon"/>
                    <xs:element ref="multiPolygon"/>
                    <xs:element ref="geometryCollection"/>
                </xs:choice>
            </xs:sequence>
        </xs:extension>
    </xs:complexContent>
</xs:complexType>

<xs:complexType name="messagePartDescVo">
    <xs:sequence>
        <xs:element name="subject" type="xs:string" minOccurs="0"/>
        <xs:element name="details" type="xs:string" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="lang" type="xs:string"/>
</xs:complexType>

<xs:complexType name="messageDescVo">
    <xs:sequence>
        <xs:element name="title" type="xs:string" minOccurs="0"/>
        <xs:element name="vicinity" type="xs:string" minOccurs="0"/>
        <xs:element name="publication" type="xs:string" minOccurs="0"/>
        <xs:element name="internalPublication" type="xs:string" minOccurs="0"/>
        <xs:element name="source" type="xs:string" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="lang" type="xs:string"/>
</xs:complexType>

<xs:complexType name="attachmentVo">
    <xs:sequence>
        <xs:element name="type" type="xs:string" minOccurs="0"/>
        <xs:element name="fileName" type="xs:string" minOccurs="0"/>
        <xs:element name="path" type="xs:string" minOccurs="0"/>
        <xs:element name="fileSize" type="xs:long" minOccurs="0"/>
        <xs:element name="display" type="attachmentDisplayType" minOccurs="0"/>
        <xs:element name="width" type="xs:string" minOccurs="0"/>
        <xs:element name="height" type="xs:string" minOccurs="0"/>
        <xs:element name="descs" type="attachmentDescVo" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="attachmentDescVo">
    <xs:sequence>
        <xs:element name="caption" type="xs:string" minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="lang" type="xs:string"/>
</xs:complexType>

<xs:simpleType name="mainType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="NW"/>
        <xs:enumeration value="NM"/>
    </xs:restriction>

```

```

</xs:simpleType>

<xs:simpleType name="type">
  <xs:restriction base="xs:string">
    <xs:enumeration value="PERMANENT_NOTICE"/>
    <xs:enumeration value="TEMPORARY_NOTICE"/>
    <xs:enumeration value="PRELIMINARY_NOTICE"/>
    <xs:enumeration value="MISCELLANEOUS_NOTICE"/>
    <xs:enumeration value="COASTAL_WARNING"/>
    <xs:enumeration value="SUBAREA_WARNING"/>
    <xs:enumeration value="NAVAREA_WARNING"/>
    <xs:enumeration value="LOCAL_WARNING"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="status">
  <xs:restriction base="xs:string">
    <xs:enumeration value="DRAFT"/>
    <xs:enumeration value="VERIFIED"/>
    <xs:enumeration value="PUBLISHED"/>
    <xs:enumeration value="EXPIRED"/>
    <xs:enumeration value="CANCELLED"/>
    <xs:enumeration value="DELETED"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="referenceType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="REFERENCE"/>
    <xs:enumeration value="REPETITION"/>
    <xs:enumeration value="REPETITION_NEW_TIME"/>
    <xs:enumeration value="CANCELLATION"/>
    <xs:enumeration value="UPDATE"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="messagePartType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="DETAILS"/>
    <xs:enumeration value="TIME"/>
    <xs:enumeration value="POSITIONS"/>
    <xs:enumeration value="NOTE"/>
    <xs:enumeration value="PROHIBITION"/>
    <xs:enumeration value="SIGNALS"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="attachmentDisplayType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="ABOVE"/>
    <xs:enumeration value="BELOW"/>
    <xs:enumeration value="SEPARATE_PAGE"/>
  </xs:restriction>
</xs:simpleType>
</xs:schema>

</definitionAsXSD>
</serviceDataModel>

<serviceInterfaces>
  <serviceInterface>
    <name>MessageService</name>
    <description>Works according to the request response pattern.</description>
    <dataExchangePattern>REQUEST_RESPONSE</dataExchangePattern>

    <operations>
      <operation>
        <name>get</name>
        <description>Retrieves published NW-NM messages.</description>
        <returnValueType>
          <typeReference>Message[]</typeReference>

```

```
</returnValueType>
<parameterTypes>
  <parameterType>
    <typeReference>lang</typeReference>
  </parameterType>
  <parameterType>
    <typeReference>domains</typeReference>
  </parameterType>
  <parameterType>
    <typeReference>messageSeries</typeReference>
  </parameterType>
  <parameterType>
    <typeReference>publications</typeReference>
  </parameterType>
  <parameterType>
    <typeReference>areaIds</typeReference>
  </parameterType>
  <parameterType>
    <typeReference>mainType</typeReference>
  </parameterType>
  <parameterType>
    <typeReference>wkt</typeReference>
  </parameterType>
  <parameterType>
    <typeReference>externalize</typeReference>
  </parameterType>
  <parameterType>
    <typeReference>dateFormat</typeReference>
  </parameterType>
</parameterTypes>
</operation>
</operations>

</serviceInterface>
</serviceInterfaces>
</serviceSpecification>
```

# ANNEX 2



# **NW-NM REST Service Technical Design v0.4**

---

## Contents

1	Introduction .....	3
1.1	Purpose of the document .....	3
1.2	Intended readership .....	3
2	Service Design Identification .....	4
3	Technology Introduction .....	5
3.1	REST .....	5
4	Service Design Overview .....	6
4.1	Service Interfaces .....	6
5	Physical Data Model .....	7
5.1	Data Model .....	7
6	Service Interface Design .....	8
6.1	Service Endpoint /public/v1/messages .....	8
6.1.1	Operation HTTP GET .....	8
7	References .....	10
8	Acronyms and Terminology .....	11
8.1	Acronyms .....	11
8.2	Terminology .....	11
Appendix A	Service Design Description XML .....	12
Appendix B	Swagger API sans GeoJSON .....	13

# 1 Introduction

## 1.1 Purpose of the document

This document covers a REST-based technical design of the MW-NM service specification [1], according to the guidelines given in the Service Description Guidelines [2].

## 1.2 Intended readership

This service technical design is intended to be read by service architects, system engineers and developers in charge of designing and developing an instance of the REST-bases NW-NM service.

Furthermore, this service specification is intended to be read by enterprise architects, service architects, information architects, system engineers and developers in pursuing architecting, design and development activities of other related services.

## 2 Service Design Identification

<b>Name</b>	NW-NM T&P Maritime Cloud REST Service
<b>ID</b>	urn:mrnx:mcl:service:dma:nw-nm:rest
<b>Version</b>	0.4
<b>Technology</b>	REST
<b>Service Specification ID</b>	urn:mrnx:mcl:service:dma:nw-nm
<b>Service Specification Version</b>	0.4
<b>Description</b>	The NW-NM Maritime Cloud REST service specification defines a combined NW-NM T&P model in JSON, along with the actual REST service API used for accessing NW-NM data, as registered in the Maritime Cloud service catalogue.
<b>Keywords</b>	NW, NM, Navigational Warnings, Notices to Mariners, MSI, Maritime Cloud Service. REST.
<b>Architect(s)</b>	e-Navigation Team Danish Maritime Authority Carl Jacobsens Vej 31 DK-2500 København K Telephone: +45 40 72 61 08 Email: <a href="mailto:mfs@dma.dk">mfs@dma.dk</a>
<b>Status</b>	Released.



## 3 Technology Introduction

### 3.1 REST

From the *Tide Level Information Technical Design*:

*REST (REpresentational State Transfer) is one way of providing interoperability between system on the internet. It allows requesting systems to access and manipulate textual representations of web resources using a uniform and predefined set of stateless operations: more than efficiently WSDL and SOAP.*

*In a web service which using REST, requests made to a resource's URI will elicit a response that maybe in XML, HTML, JSON or some other defined format. The response may confirm that some alteration has been made to the stored resource, and it may provide hypertext links to other related resources or collections of resources.*

*Using HTTP, as is most common, the kind of operations available include those predefined by the HTTP verbs GET, POST, PUT, DELETE and so on. By making use of a stateless protocol and standard operations, REST aim for fast performance, reliability, and the ability to grow, by re-using components that can be managed and updated without affecting the system as a whole, even while it is running.*

For more details, please refer to [https://en.wikipedia.org/wiki/Representational\\_state\\_transfer](https://en.wikipedia.org/wiki/Representational_state_transfer)

## 4 Service Design Overview

This chapter will outline the REST implementation of the services described in the Service Specification [1].

### 4.1 Service Interfaces

The NW-NM REST service consists of a single public REST endpoint to query the currently published NW-NM messages (Request/Reply Message Exchange Pattern).

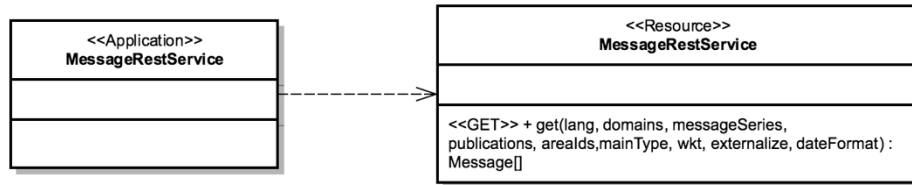


Figure 1: NW-NM Service Definition Diagram

ServiceInterface	Role (from service provider point of view)	ServiceOperation
MessageRestService	Provided	get(lang, domains, messageSeries, publications, arealds, mainType, wkt, externalize, dateFormat)

The returned data model is detailed in chapter 5 and the service operation in chapter 6.

## 5 Physical Data Model

This chapter details the concrete JSON data model implementation of the data model described in the Service Specification [1].

### 5.1 Data Model

The JSON data model is detailed in Appendix B as a Swagger Specification (a.k.a. OpenAPI specification) [3].

There is a very direct 1:1 mapping between the UML/XML model detailed in the Service Specification and the JSON data model adopted in this technical design. In fact, both are generated from the same Java class library (niord-model).

Hence, this technical design does not provide an explicit mapping table between the Service Specification model and the Technical Design JSON model.

## 6 Service Interface Design

This chapter describes the details of the NW-NM REST service interface, which is the REST based implementation of the MessageService specified in the Service Specification [1].

### 6.1 Service Endpoint /public/v1/messages

The NW-NM service is comprised of a single REST endpoint, /public/v1/messages, which follows the Request/Reply Message Exchange Pattern. Sequence diagram:

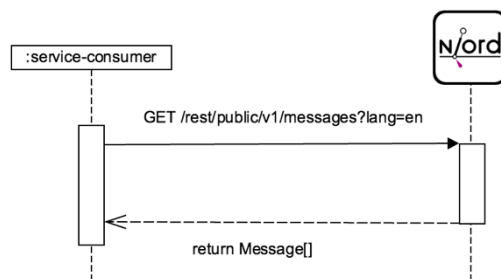


Figure 1: NW-NM Service Sequence Diagram

The formal service interface is detailed in Appendix B as a Swagger Specification (a.k.a. OpenAPI specification) [3].

#### 6.1.1 Operation HTTP GET

The HTTP GET operation returns the list of all published NW and NM messages. The returned result can be controlled using the following parameters:

Parameter	Type	Description
lang	String	An optional ISO 639-1 language code. If specified, only this language variant is returned for localized entities such as Message, Area and Category. However, if, say, “en” is requested and an entity only has a “da” language description entity, then this is returned instead. A client may want to flag this to the end user.
domains	String[]	List of domain ID’s. Each Niord system organizes messages into domains, e.g. for Navigational Warnings, Notices to Mariners, Firing Exercises, Almanacs, etc.
messageSeries	String[]	List of message series ID’s. Each message is associated with a message series, as defined by the producing Niord system. As an example, there may be separate messages series for local navigational warnings and other navigational warnings.
publications	String[]	Some messages may have been published in <i>publications</i> , such as a weekly NtM.

Parameter	Type	Description
		The <i>publication</i> parameter can be used to filter by specific publications.
arealds	String[]	Messages may be associated with one or more area, as defined by the producing Niord System. The <i>areald</i> parameter can be used to filter messages by areas using either the internal ID or the MRN of the areas.
mainType	String	Used to filter messages by their main type, i.e. "NM" or "NW".
wkt	String	Optional Well-Known Text representation of a geometry. If specified, only messages intersecting the geometry will be returned.
externalize	Boolean	Whether to rewrite all embedded links and paths to be absolute URL's pointing back to their repository URL in the producing system. Default value is <i>true</i> .
dateFormat	String	The date format to use for JSON date-time encoding. Either <i>UNIX_EPOCH</i> (default) or <i>ISO_8601</i> .
Header	values	Description
Accept	application/json, application/xml	Request that the returned data is in the JSON or XML format. Only the JSON format is described in this document.
Return	Type	Description
	Message[]	The list of published NW and NM messages in the requested format.

## 7 References

Nr.	Reference
[1] NW-NM Service Specification	E2 - NW-NM Service Specification version 0.4
[2] Service Description Guidelines	E2_Deliverable D3.4 – Service Specification Template, version 1.0
[3] Swagger (OpenAPI) specification.	Available at <a href="https://github.com/OAI/OpenAPI-Specification">https://github.com/OAI/OpenAPI-Specification</a>

## 8 Acronyms and Terminology

### 8.1 Acronyms

Term	Definition
<b>API</b>	Application Programming Interface
<b>DMA</b>	Danish Maritime Authority
<b>MC</b>	Maritime Cloud
<b>MEP</b>	Message Exchange Pattern
<b>MRN</b>	Maritime Resource Name
<b>NM</b>	Notices to Mariners
<b>NW</b>	Navigational Warning
<b>REST</b>	Representational State Transfer
<b>XML</b>	Extendible Mark-up Language
<b>XSD</b>	XML Schema Definition

### 8.2 Terminology

Term	Definition
<b>Navigational Warnings</b>	Navigational Warnings (NW) are part of the Maritime Safety Information (MSI) system. Currently, NW's are promulgated in text via SafetyNET, NAVTEX, and is in some countries accessible on the WWW or as voice broadcasts via coastal radio stations.
<b>Niord</b>	Anglified name of the Norse God Njord, associated with the sea and seafaring. Also the name of the EfficienSea 2 sub-project implementation of the NW-NM service and authoring system. See <a href="http://niord.org">http://niord.org</a>
<b>niord-model</b>	A java implementation of the JSON/XML model used by the NW-NM REST service. Available at <a href="https://github.com/NiordOrg/niord/tree/master/niord-model">https://github.com/NiordOrg/niord/tree/master/niord-model</a>
<b>Notices to Mariners</b>	Notices to Mariners (NM) are promulgated weekly in order to keep nautical charts and publications, as far as possible, up to date. Temporary (T) and Preliminary (P) NMs advise mariners of important matters affecting navigational safety, including new hydrographic information (in advance of new editions or chart updates), changes to routing measures and aids to navigation, and other important categories of data. NM T&P's are today promulgated on paper weekly, fortnightly or monthly and are often accessible on the WWW in PDF format. Not all ENC's include T&P information currently.
<b>Swagger API</b>	Swagger provides a framework (and utilities) for specifying a REST API.

## Appendix A Service Design Description XML

This appendix contains the formal XML definition of the service Technical Design. The actual Swagger API has been included separately in Appendix B.

```
<?xml version="1.0" encoding="UTF-8"?>
<serviceDesign
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xsi:schemaLocation="http://efficiensea2.org/maritime-cloud/service-
registry/v1/ServiceDesignSchema.xsd ServiceDesignSchema.xsd "
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ServiceDesignSchema="http://efficiensea2.org/maritime-cloud/service-
registry/v1/ServiceDesignSchema.xsd"
  xmlns:ServiceSpecificationSchema="http://efficiensea2.org/maritime-cloud/service-
registry/v1/ServiceSpecificationSchema.xsd"
  xmlns="http://efficiensea2.org/maritime-cloud/service-registry/v1/ServiceDesignSchema.xsd">

  <id>urn:mrn:mcl:service:design:dma:nw-nm-rest</id>
  <version>0.4</version>
  <name>NW-NM TP Maritime Cloud REST Service</name>
  <status>released</status>
  <description>A REST-based implementation of the NW-NM Maritime Cloud service
specification.</description>
  <offersTransport>
    <offersTransport>
      <name>REST</name>
      <description>This service implementation is available as REST over HTTPS</description>
      <protocol>HTTPS</protocol>
    </offersTransport>
  </offersTransport>
  <designsServiceSpecifications>
    <designsServiceSpecifications>
      <id>urn:mrn:mcl:service:specification:dma:nw-nm</id>
      <version>0.4</version>
    </designsServiceSpecifications>
  </designsServiceSpecifications>
  <designedBy>
    <ServiceSpecificationSchema:id>urn:mrn:mcl:user:dma:mfs</ServiceSpecificationSchema:id>
    <ServiceSpecificationSchema:name>Mads Friis Sørensen</ServiceSpecificationSchema:name>
    <ServiceSpecificationSchema:description>Responsible for the NW-NM
service</ServiceSpecificationSchema:description>
    <ServiceSpecificationSchema:contactInfo>mfs@dma.dk</ServiceSpecificationSchema:contactInfo>
    <ServiceSpecificationSchema:isCommercial>>false</ServiceSpecificationSchema:isCommercial>
  </designedBy>
  <servicePhysicalDataModel>
    <name>NW-NM REST Service Swagger API</name>
    <description>Swagger API of the NW-NM REST Service</description>
    <modelType>JSON</modelType>
    <model>
      http://niord.e-navigation.net/rest/swagger.json
    </model>
  </servicePhysicalDataModel>
</serviceDesign>
```



## Appendix B Swagger API sans GeoJSON

This appendix contains the formal Swagger definition of the NW-NM REST service.

The complete Swagger definition can be found at <http://niord.e-navigation.net/rest/swagger.json>. The Swagger definition defines a few operations and classes not part of the NW-NM service specification. For clarity reasons, these have been removed from the Swagger file below:

```
{
  "swagger": "2.0",
  "info": {
    "version": "1.0.0"
  },
  "basePath": "/rest",
  "tags": [
    {
      "name": "messages"
    }
  ],
  "paths": {
    "/public/v1/messages": {
      "get": {
        "tags": [
          "messages"
        ],
        "summary": "Returns the published NW and NM messages",
        "description": "",
        "operationId": "searchMessages",
        "produces": [
          "application/json;charset=UTF-8"
        ],
        "parameters": [
          {
            "name": "lang",
            "in": "query",
            "description": "Two-letter ISO 639-1 language code",
            "required": false,
            "type": "string",
            "x-example": "en"
          },
          {
            "name": "domain",
            "in": "query",
            "description": "The IDs of the domains to select messages from",
            "required": false,
            "type": "array",
            "items": {
              "type": "string"
            },
            "collectionFormat": "multi",
            "x-example": "niord-client-nw"
          },
          {
            "name": "messageSeries",
            "in": "query",
            "description": "Specific message series to select messages from",
```

```

    "required": false,
    "type": "array",
    "items": {
      "type": "string"
    },
    "collectionFormat": "multi",
    "x-example": "dma-nw"
  },
  {
    "name": "publication",
    "in": "query",
    "description": "The IDs of the publications to select message from",
    "required": false,
    "type": "array",
    "items": {
      "type": "string"
    },
    "collectionFormat": "multi"
  },
  {
    "name": "areaId",
    "in": "query",
    "description": "The IDs of the areas to select messages from",
    "required": false,
    "type": "array",
    "items": {
      "type": "string"
    },
    "collectionFormat": "multi",
    "x-example": "urn:mrn:iho:country:dk"
  },
  {
    "name": "mainType",
    "in": "query",
    "description": "Either NW (navigational warnings) or NM (notices to mariners)",
    "required": false,
    "type": "array",
    "items": {
      "type": "string",
      "enum": [
        "NW",
        "NM"
      ]
    },
    "collectionFormat": "multi",
    "x-example": "NW"
  },
  {
    "name": "wkt",
    "in": "query",
    "description": "Well-Known Text for geographical extent",
    "required": false,
    "type": "string",
    "x-example": "POLYGON((7 54, 7 57, 13 56, 13 57, 7 54))"
  },
  {
    "name": "externalize",
    "in": "query",
    "description": "Whether to rewrite all embedded links and paths to be absolute
URL's",
    "required": false,
    "type": "boolean",
    "default": true,
    "x-example": true
  }

```

```

    },
    {
      "name": "dateFormat",
      "in": "query",
      "description": "The date format to use for JSON date-time encoding. Either
'UNIX_EPOCH' or 'ISO_8601'",
      "required": false,
      "type": "string",
      "default": "UNIX_EPOCH",
      "x-example": "UNIX_EPOCH",
      "enum": [
        "UNIX_EPOCH",
        "ISO_8601"
      ]
    }
  ],
  "responses": {
    "200": {
      "description": "successful operation",
      "schema": {
        "type": "array",
        "items": {
          "$ref": "#/definitions/Message"
        }
      }
    }
  }
},
{
  "definitions": {
    "MultiPoint": {
      "allOf": [
        {
          "$ref": "#/definitions/GeoJson"
        },
        {
          "type": "object",
          "properties": {
            "coordinates": {
              "type": "array",
              "items": {
                "type": "array",
                "items": {
                  "type": "number",
                  "format": "double"
                }
              }
            }
          }
        }
      ],
      "description": "GeoJson MultiPoint type",
      "xml": {
        "name": "multiPoint"
      }
    }
  }
},
{
  "MultiLineString": {
    "allOf": [
      {
        "$ref": "#/definitions/GeoJson"
      },
      {
        "type": "object",

```

```

        "properties": {
          "coordinates": {
            "type": "array",
            "items": {
              "type": "array",
              "items": {
                "type": "array",
                "items": {
                  "type": "number",
                  "format": "double"
                }
              }
            }
          }
        },
        "description": "GeoJson MultiLineString type",
        "xml": {
          "name": "multiLineString"
        }
      }
    ],
    "MessageSeries": {
      "type": "object",
      "properties": {
        "seriesId": {
          "type": "string"
        },
        "mainType": {
          "type": "string",
          "enum": [
            "NW",
            "NM"
          ]
        }
      },
      "description": "A message series",
      "xml": {
        "name": "messageSeries"
      }
    },
    "AttachmentDesc": {
      "type": "object",
      "properties": {
        "lang": {
          "type": "string",
          "xml": {
            "attribute": true
          }
        },
        "caption": {
          "type": "string"
        }
      },
      "description": "Translatable fields of the Attachment model"
    },
    "Category": {
      "type": "object",
      "properties": {
        "id": {
          "type": "integer",
          "format": "int32",
          "xml": {
            "attribute": true
          }
        }
      }
    }
  }

```

```

    },
    "mrn": {
      "type": "string"
    },
    "active": {
      "type": "boolean",
      "default": false
    },
    "parent": {
      "$ref": "#/definitions/Category"
    },
    "descs": {
      "type": "array",
      "items": {
        "$ref": "#/definitions/CategoryDesc"
      }
    }
  },
  "description": "Hierarchical category model",
  "xml": {
    "name": "category"
  }
},
"Message": {
  "type": "object",
  "properties": {
    "id": {
      "type": "string",
      "xml": {
        "attribute": true
      }
    },
    "created": {
      "type": "string",
      "format": "date-time",
      "xml": {
        "attribute": true
      }
    },
    "updated": {
      "type": "string",
      "format": "date-time",
      "xml": {
        "attribute": true
      }
    },
    "messageSeries": {
      "$ref": "#/definitions/MessageSeries"
    },
    "number": {
      "type": "integer",
      "format": "int32"
    },
    "shortId": {
      "type": "string"
    },
    "mainType": {
      "type": "string",
      "enum": [
        "NW",
        "NM"
      ]
    }
  }
},

```

```

"type": {
  "type": "string",
  "enum": [
    "PERMANENT_NOTICE",
    "TEMPORARY_NOTICE",
    "PRELIMINARY_NOTICE",
    "MISCELLANEOUS_NOTICE",
    "COASTAL_WARNING",
    "SUBAREA_WARNING",
    "NAVAREA_WARNING",
    "LOCAL_WARNING"
  ]
},
"status": {
  "type": "string",
  "enum": [
    "DRAFT",
    "VERIFIED",
    "PUBLISHED",
    "EXPIRED",
    "CANCELLED",
    "DELETED"
  ]
},
"areas": {
  "type": "array",
  "items": {
    "$ref": "#/definitions/Area"
  }
},
"categories": {
  "type": "array",
  "items": {
    "$ref": "#/definitions/Category"
  }
},
"charts": {
  "type": "array",
  "items": {
    "$ref": "#/definitions/Chart"
  }
},
"horizontalDatum": {
  "type": "string"
},
"publishDateFrom": {
  "type": "string",
  "format": "date-time"
},
"publishDateTo": {
  "type": "string",
  "format": "date-time"
},
"followUpDate": {
  "type": "string",
  "format": "date-time"
},
"references": {
  "type": "array",
  "items": {
    "$ref": "#/definitions/Reference"
  }
},
"originalInformation": {

```

```

        "type": "boolean",
        "default": false
    },
    "parts": {
        "type": "array",
        "items": {
            "$ref": "#/definitions/MessagePart"
        }
    },
    "descs": {
        "type": "array",
        "items": {
            "$ref": "#/definitions/MessageDesc"
        }
    },
    "attachments": {
        "type": "array",
        "items": {
            "$ref": "#/definitions/Attachment"
        }
    }
},
"description": "Main NW and NM message class",
"xml": {
    "name": "message"
}
},
"Crs": {
    "type": "object",
    "properties": {
        "type": {
            "type": "string"
        },
        "properties": {
            "type": "object",
            "additionalProperties": {
                "type": "string"
            }
        }
    }
},
"description": "GeoJson Coordinate Reference System"
},
"Attachment": {
    "type": "object",
    "properties": {
        "type": {
            "type": "string"
        },
        "path": {
            "type": "string"
        },
        "fileName": {
            "type": "string"
        },
        "fileSize": {
            "type": "integer",
            "format": "int64"
        },
        "display": {
            "type": "string",
            "enum": [
                "ABOVE",
                "BELOW",
                "SEPARATE_PAGE"
            ]
        }
    }
}

```

```

    ],
    "width": {
      "type": "string"
    },
    "height": {
      "type": "string"
    },
    "descs": {
      "type": "array",
      "items": {
        "$ref": "#/definitions/AttachmentDesc"
      }
    }
  },
  "description": "Message attachment model",
  "xml": {
    "name": "attachment"
  }
},
"FeatureCollectionVo": {
  "allOf": [
    {
      "$ref": "#/definitions/FeatureCollection"
    },
    {
      "type": "object",
      "properties": {},
      "description": "GeoJson FeatureCollection type",
      "xml": {
        "name": "featureCollection"
      }
    }
  ]
},
"GeoJson": {
  "type": "object",
  "discriminator": "type",
  "properties": {
    "type": {
      "type": "string"
    },
    "crs": {
      "$ref": "#/definitions/Crs"
    },
    "bbox": {
      "type": "array",
      "items": {
        "type": "number",
        "format": "double"
      }
    }
  },
  "description": "Superclass for GeoJson types"
},
"DateInterval": {
  "type": "object",
  "properties": {
    "allDay": {
      "type": "boolean",
      "default": false
    },
    "fromDate": {
      "type": "string",

```



```

    "format": "date-time"
  },
  "toDate": {
    "type": "string",
    "format": "date-time"
  }
},
"description": "Date interval"
},
"Feature": {
  "type": "object",
  "discriminator": "type",
  "properties": {
    "type": {
      "type": "string"
    },
    "crs": {
      "$ref": "#/definitions/Crs"
    },
    "bbox": {
      "type": "array",
      "items": {
        "type": "number",
        "format": "double"
      }
    },
    "id": {
      "type": "object"
    },
    "geometry": {
      "$ref": "#/definitions/GeoJson"
    },
    "properties": {
      "type": "object",
      "additionalProperties": {
        "type": "object"
      }
    }
  }
},
"description": "GeoJson Feature type",
"xml": {
  "name": "feature"
}
},
"FeatureVo": {
  "allOf": [
    {
      "$ref": "#/definitions/Feature"
    },
    {
      "type": "object",
      "properties": {},
      "description": "GeoJson Feature type",
      "xml": {
        "name": "feature"
      }
    }
  ]
},
"MessageDesc": {
  "type": "object",
  "properties": {
    "lang": {
      "type": "string",

```

```

        "xml": {
          "attribute": true
        },
        "title": {
          "type": "string"
        },
        "vicinity": {
          "type": "string"
        },
        "publication": {
          "type": "string"
        },
        "internalPublication": {
          "type": "string"
        },
        "source": {
          "type": "string"
        }
      },
      "description": "Translatable fields of the Message model"
    },
    "GeometryCollection": {
      "allOf": [
        {
          "$ref": "#/definitions/GeoJson"
        },
        {
          "type": "object",
          "properties": {
            "geometries": {
              "type": "array",
              "items": {
                "$ref": "#/definitions/GeoJson"
              }
            }
          }
        }
      ],
      "description": "GeoJson GeometryCollection type",
      "xml": {
        "name": "geometryCollection"
      }
    }
  ],
  "ReferenceDesc": {
    "type": "object",
    "properties": {
      "lang": {
        "type": "string",
        "xml": {
          "attribute": true
        }
      }
    },
    "description": {
      "type": "string"
    }
  },
  "description": "Translatable fields of the Reference model"
},
"Polygon": {
  "allOf": [
    {
      "$ref": "#/definitions/GeoJson"
    },

```

```

{
  "type": "object",
  "properties": {
    "coordinates": {
      "type": "array",
      "items": {
        "type": "array",
        "items": {
          "type": "array",
          "items": {
            "type": "number",
            "format": "double"
          }
        }
      }
    }
  }
},
{
  "description": "GeoJson Polygon type",
  "xml": {
    "name": "polygon"
  }
}
],
"AreaDesc": {
  "type": "object",
  "properties": {
    "lang": {
      "type": "string",
      "xml": {
        "attribute": true
      }
    }
  },
  "name": {
    "type": "string"
  }
},
"description": "Translatable fields of the Area model"
},
"Chart": {
  "type": "object",
  "properties": {
    "chartNumber": {
      "type": "string"
    },
    "internationalNumber": {
      "type": "integer",
      "format": "int32"
    },
    "active": {
      "type": "boolean",
      "default": false
    },
    "scale": {
      "type": "integer",
      "format": "int32"
    },
    "name": {
      "type": "string"
    }
  },
  "description": "Sea chart model",
  "xml": {
    "name": "chart"
  }
}

```

```

    },
    "CategoryDesc": {
      "type": "object",
      "properties": {
        "lang": {
          "type": "string",
          "xml": {
            "attribute": true
          }
        },
        "name": {
          "type": "string"
        }
      }
    },
    "description": "Translatable fields of the Category model"
  },
  "Reference": {
    "type": "object",
    "properties": {
      "messageId": {
        "type": "string"
      },
      "type": {
        "type": "string",
        "enum": [
          "REFERENCE",
          "REPETITION",
          "REPETITION_NEW_TIME",
          "CANCELLATION",
          "UPDATE"
        ]
      },
      "descs": {
        "type": "array",
        "items": {
          "$ref": "#/definitions/ReferenceDesc"
        }
      }
    },
    "description": "A message reference"
  },
  "MessagePart": {
    "type": "object",
    "properties": {
      "indexNo": {
        "type": "integer",
        "format": "int32"
      },
      "type": {
        "type": "string",
        "enum": [
          "DETAILS",
          "TIME",
          "POSITIONS",
          "NOTE",
          "PROHIBITION",
          "SIGNALS"
        ]
      },
      "eventDates": {
        "type": "array",
        "items": {
          "$ref": "#/definitions/DateInterval"
        }
      }
    }
  }
}

```

```

    },
    "geometry": {
      "$ref": "#/definitions/FeatureCollection"
    },
    "descs": {
      "type": "array",
      "items": {
        "$ref": "#/definitions/MessagePartDesc"
      }
    },
    "hideSubject": {
      "type": "boolean",
      "default": false
    }
  },
  "description": "Main NW and NM message part class"
},
"MessagePartDesc": {
  "type": "object",
  "properties": {
    "lang": {
      "type": "string",
      "xml": {
        "attribute": true
      }
    },
    "subject": {
      "type": "string"
    },
    "details": {
      "type": "string"
    }
  },
  "description": "Translatable fields of the MessagePart model"
},
"Point": {
  "allOf": [
    {
      "$ref": "#/definitions/GeoJson"
    },
    {
      "type": "object",
      "properties": {
        "coordinates": {
          "type": "array",
          "items": {
            "type": "number",
            "format": "double"
          }
        }
      }
    }
  ],
  "description": "GeoJson Point type",
  "xml": {
    "name": "point"
  }
}
]
},
"Area": {
  "type": "object",
  "properties": {
    "id": {
      "type": "integer",

```

```

        "format": "int32",
        "xml": {
          "attribute": true
        }
      },
      "mrn": {
        "type": "string"
      },
      "active": {
        "type": "boolean",
        "default": false
      },
      "parent": {
        "$ref": "#/definitions/Area"
      },
      "descs": {
        "type": "array",
        "items": {
          "$ref": "#/definitions/AreaDesc"
        }
      }
    },
    "description": "Hierarchical area model",
    "xml": {
      "name": "area"
    }
  },
  "FeatureCollection": {
    "type": "object",
    "discriminator": "type",
    "properties": {
      "type": {
        "type": "string"
      },
      "crs": {
        "$ref": "#/definitions/Crs"
      },
      "bbox": {
        "type": "array",
        "items": {
          "type": "number",
          "format": "double"
        }
      },
      "id": {
        "type": "object"
      },
      "features": {
        "type": "array",
        "items": {
          "$ref": "#/definitions/Feature"
        }
      }
    },
    "description": "GeoJson FeatureCollection type",
    "xml": {
      "name": "featureCollection"
    }
  },
  "LineString": {
    "allOf": [
      {
        "$ref": "#/definitions/GeoJson"
      }
    ]
  },

```

```

{
  "type": "object",
  "properties": {
    "coordinates": {
      "type": "array",
      "items": {
        "type": "array",
        "items": {
          "type": "number",
          "format": "double"
        }
      }
    }
  }
},
"description": "GeoJson LineString type",
"xml": {
  "name": "lineString"
}
]
},
"MultiPolygon": {
  "allOf": [
    {
      "$ref": "#/definitions/GeoJson"
    },
    {
      "type": "object",
      "properties": {
        "coordinates": {
          "type": "array",
          "items": {
            "type": "array",
            "items": {
              "type": "array",
              "items": {
                "type": "number",
                "format": "double"
              }
            }
          }
        }
      }
    }
  ],
  "description": "GeoJson MultiPolygon type",
  "xml": {
    "name": "multiPolygon"
  }
}
]
}
}
}

```

# ANNEX 3





# **NW-NM DMA Service Instance v0.4**

---

## Contents

1	Introduction .....	3
1.1	Purpose of the document .....	3
1.2	Intended readership .....	3
2	Service Instance Identification .....	4
3	Service Implementation and Instantiation Details .....	5
3.1	NW-NM REST Service URL .....	5
3.2	Service Endpoint /public/v1/messages .....	5
4	Coverage Area .....	6
5	Service Level .....	7
6	Commercial Information .....	8
7	References .....	9
8	Acronyms and Terminology .....	10
8.1	Acronyms .....	10
8.2	Terminology .....	10
	Appendix A Service Instance XML .....	11

# 1 Introduction

## 1.1 Purpose of the document

This document covers a DMA instance of the REST-based technical design [3] of the MW-NM service specification [1], according to the guidelines given in the Service Description Guidelines [2].

## 1.2 Intended readership

This service instance document is intended to be read by service architects, system engineers and developers in charge of designing and developing client services consuming NW-NM messages from the Danish Maritime Waters.

## 2 Service Instance Identification

<b>Name</b>	DMA NW-NM T&P Maritime Cloud REST Service
<b>ID</b>	urn:mrn:mcl:service:instance:dma:nw-nm
<b>Version</b>	0.4
<b>Technology</b>	REST
<b>Service Specification ID</b>	urn:mrn:mcl:service:specification:dma:nw-nm
<b>Service Specification Version</b>	0.4
<b>Service Design ID</b>	urn:mrn:mcl:service:dma:nw-nm:rest
<b>Service Design Version</b>	0.4
<b>Description</b>	A DMA instance of the NW-NM REST Service.
<b>Keywords</b>	NW, NM, Navigational Warnings, Notices to Mariners, MSI, Maritime Cloud Service. REST. Danish Maritime Authority, DMA.
<b>Supplier</b>	e-Navigation Team Danish Maritime Authority Carl Jacobsens Vej 31 DK-2500 København K Telephone: +45 40 72 61 08 Email: <a href="mailto:mfs@dma.dk">mfs@dma.dk</a>
<b>Status</b>	Released.

### 3 Service Implementation and Instantiation Details

This chapter will define the absolute URL of DMA's NW-NM REST service.

#### 3.1 NW-NM REST Service URL

The DMA NW-NM REST service is located at:

- <http://niord-dma.e-navigation.net/rest>

#### 3.2 Service Endpoint /public/v1/messages

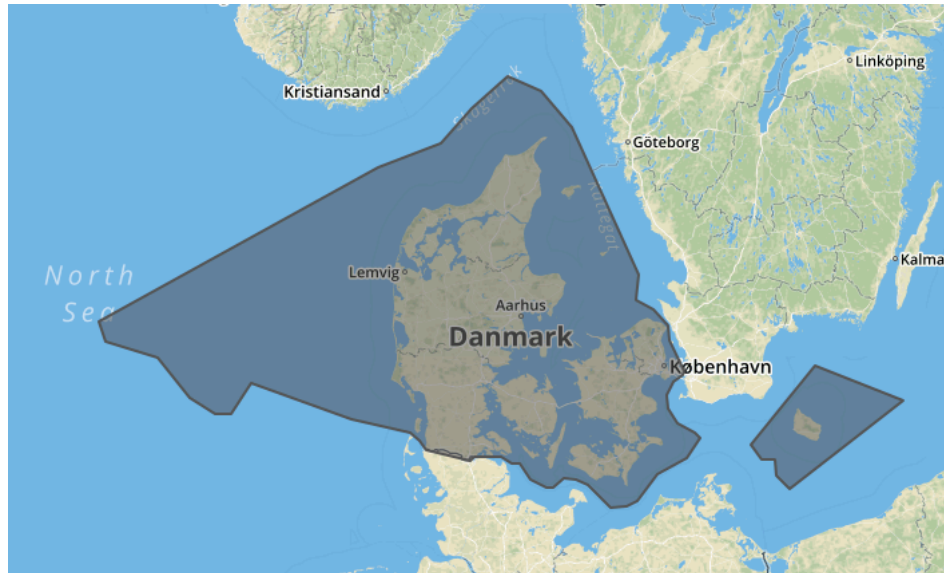
The NW-NM REST Service Specification only specifies a single service interface and operation (i.e. REST endpoint via HTTP GET): /public/v1/messages

The full URL of the service endpoint thus becomes:

- <http://niord-dma.e-navigation.net/rest/public/v1/messages>

## 4 Coverage Area

The coverage area of DMA's NW-NM service is the Danish Maritime Waters, which is approximately given by:



The actual WKT definition is part of the service instance XML in Appendix A.

## 5 Service Level

Availability is not guaranteed.

## 6 Commercial Information

DMA's NW-NM REST service is publicly available and can be used for free.



## 7 References

Nr.	Reference
[1] NW-NM Service Specification	E2 - NW-NM Service Specification version 0.1
[2] Service Description Guidelines	E2_Deliverable D3.4 – Service Specification Template, version 1.0
[3] NW-NM REST Service Technical Design.	E2 - NW-NM REST Service Technical Design version 0.1

## 8 Acronyms and Terminology

### 8.1 Acronyms

Term	Definition
<b>API</b>	Application Programming Interface
<b>DMA</b>	Danish Maritime Authority
<b>MC</b>	Maritime Cloud
<b>MEP</b>	Message Exchange Pattern
<b>MRN</b>	Maritime Resource Name
<b>NM</b>	Notices to Mariners
<b>NW</b>	Navigational Warning
<b>REST</b>	Representational State Transfer
<b>WKT</b>	Well-Known Text
<b>XML</b>	Extendible Mark-up Language
<b>XSD</b>	XML Schema Definition

### 8.2 Terminology

Term	Definition
<b>Docker</b>	Docker containers wrap up a piece of software in a complete filesystem that contains everything it needs to run: code, runtime, system tools, system libraries.
<b>Navigational Warnings</b>	Navigational Warnings (NW) are part of the Maritime Safety Information (MSI) system. Currently, NW's are promulgated in text via SafetyNET, NAVTEX, and is in some countries accessible on the WWW or as voice broadcasts via coastal radio stations.
<b>Keycloak</b>	JBoss Keycloak is an integrated single-sign-on and identity management system for browser apps and RESTful web services.
<b>Niord</b>	Anglified name of the Norse God Njord, associated with the sea and seafaring. Also the name of the EfficienSea 2 sub-project implementation of the NW-NM service and authoring system. See <a href="http://niord.org">http://niord.org</a>
<b>Notices to Mariners</b>	Notices to Mariners (NM) are promulgated weekly in order to keep nautical charts and publications, as far as possible, up to date. Temporary (T) and Preliminary (P) NMs advise mariners of important matters affecting navigational safety, including new hydrographic information (in advance of new editions or chart updates), changes to routing measures and aids to navigation, and other important categories of data. NM T&P's are today promulgated on paper weekly, fortnightly or monthly and are often accessible on the WWW in PDF format. Not all ENC's include T&P information currently.

## Appendix A Service Instance XML

This appendix contains the formal XML definition of the service instance.

```
<?xml version="1.0" encoding="UTF-8"?>
<serviceInstance
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xsi:schemaLocation="http://efficiensea2.org/maritime-cloud/service-
registry/v1/ServiceInstanceSchema.xsd ServiceInstanceSchema.xsd "
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ServiceInstanceSchema="http://efficiensea2.org/maritime-cloud/service-
registry/v1/ServiceInstanceSchema.xsd"
  xmlns:ServiceSpecificationSchema="http://efficiensea2.org/maritime-cloud/service-
registry/v1/ServiceSpecificationSchema.xsd"
  xmlns="http://efficiensea2.org/maritime-cloud/service-registry/v1/ServiceInstanceSchema.xsd">

  <id>urn:mrn:mcl:service:instance:dma:nw-nm</id>
  <version>0.4</version>
  <name>Danish NW and NM T&P Service</name>
  <status>released</status>
  <description> A DMA instance of the NW-NM REST Service. </description>
  <keywords> NW, NM, Navigational Warnings, Notices to Mariners, MSI, Maritime Cloud Service. REST.
Danish Maritime Authority, DMA </keywords>
  <URL>https://niord-dma.e-navigation.net/rest</URL>
  <requiresAuthorization>false</requiresAuthorization>
  <implementsServiceDesign>
    <id>urn:mrn:mcl:service:design:dma:nw-nm-rest</id>
    <version>0.4</version>
  </implementsServiceDesign>
  <offersServiceLevel>
    <availability>0</availability>
    <name>DMA NW-NM Service Level</name>
    <description> No pay => no availability is guaranteed. </description>
  </offersServiceLevel>
  <coversAreas>
    <coversArea>
      <name>Danish Maritime Waters</name>
      <description> Loosely defined region in the western part of the North Atlantic Ocean. </description>
      <geometryAsWKT>MULTIPOLYGON (((9.624023437500002 54.838663612975125, 9.448242187500002
54.84498993218759,
      9.382324218750002 54.807017138462555, 9.206542968750002 54.832336301970344, 8.6572265625
54.90819859298938,
      8.536376953125 54.990221720048936, 8.382568359375002 55.065786886591724, 7.415771484375
55.19768334019969,
      5.778808593749998 55.528630522571916, 5.44921875 55.24781504467555, 5.185546875
55.24155203565252,
      4.757080078125 55.391592107033404, 4.229736328125 55.76421316483771, 3.3837890624999996
55.91227293006361,
      3.2739257812499996 56.09042714399155, 7.8662109375 57.48040333923342, 8.887939453125
57.692405535264584,
      9.404296875 57.99063188288076, 9.99755859375 58.269065573473284, 10.535888671875
58.14751859907358,
      11.041259765625002 57.83305491291088, 12.15087890625 56.5231395643722, 12.10693359375
56.29825315291387,
      12.384338378906248 56.20975914792473, 12.634277343749996 56.058235955596075, 12.664489746093746
56.015272531542365,
      12.656249999999998 55.91996893509676, 12.711181640624998 55.82134464477079, 12.892456054687496
55.64659898563684,
      12.878723144531248 55.60783270038269, 12.716674804687498 55.541064956111, 12.7056884765625
55.48819145580225,
      12.617797851562498 55.41654360858007, 12.6397705078125 55.285372382493534, 12.7935791015625
55.15376626853558,
      13.062744140624998 55.06893234377864, 13.1561279296875 55.01542594056298, 12.930908203124998
54.82917227452137,
      12.7276611328125 54.76267040025496, 12.453002929687498 54.680183097099984, 12.117919921875
54.41573362292809,
      11.942138671874996 54.36455818952146, 11.678466796874998 54.35815677227373, 11.321411132812498
54.56569261911193,
      11.118164062499996 54.62933821655574, 10.925903320312498 54.63569730606386, 10.739135742187498
```

```

54.54339315407256,
  10.623779296874998 54.54339315407256, 10.360107421874998 54.62933821655574, 10.184326171874998
54.77534585936445,
  10.057983398437496 54.77534585936445, 9.876708984374998 54.8386636129751, 9.624023437500002
54.838663612975125)),
  ((14.0020751953125 54.95869417101662, 15.0457763671875 55.6930679264579, 16.5069580078125
55.363502833950776,
  14.633789062500002 54.53383250794428, 14.414062499999998 54.65794628989232, 14.3975830078125
54.81334841741929,
  14.161376953124998 54.81334841741929, 14.0020751953125 54.95869417101662))) </geometryAsWKT>
</coversArea>
</coversAreas>
<producedBy>
  <ServiceSpecificationSchema:id>urn:mrn:mcl:user:dma:mfs</ServiceSpecificationSchema:id>
  <ServiceSpecificationSchema:name>Mads Friis Sørensen</ServiceSpecificationSchema:name>
  <ServiceSpecificationSchema:description> Responsible for producing the DMA NW-NM REST service
</ServiceSpecificationSchema:description>
  <ServiceSpecificationSchema:contactInfo>mfs@dma.dk</ServiceSpecificationSchema:contactInfo>
  <ServiceSpecificationSchema:isCommercial>>false</ServiceSpecificationSchema:isCommercial>
</producedBy>
<providedBy>
  <ServiceSpecificationSchema:id>urn:mrn:mcl:user:dma:mfs</ServiceSpecificationSchema:id>
  <ServiceSpecificationSchema:name>Mads Friis Sørensen</ServiceSpecificationSchema:name>
  <ServiceSpecificationSchema:description>Responsible for providing the DMA NW-NM REST
service</ServiceSpecificationSchema:description>
  <ServiceSpecificationSchema:contactInfo>mfs@dma.dk</ServiceSpecificationSchema:contactInfo>
  <ServiceSpecificationSchema:isCommercial>>false</ServiceSpecificationSchema:isCommercial>
</providedBy>
</serviceInstance>

```

