Date: July 2019

OBJECT MANAGEMENT GROUP®

Command and Control Interface for Navigation

FTF – Beta 1

OMG Document Number: dtc/19-07-03

Normative reference: https://www.omg.org/spec/C2INAV/

Machine Consumable files

Normative:

https://www.omg.org/spec/C2INAV/20190601/C2INAV_Model_XMI.xml
https://www.omg.org/spec/C2INAV/20190601/IDL/Attitude.idl
https://www.omg.org/spec/C2INAV/20190601/IDL/Depth.idl
https://www.omg.org/spec/C2INAV/20190601/IDL/Ext.idl
https://www.omg.org/spec/C2INAV/20190601/IDL/Navigation_Domain.idl
https://www.omg.org/spec/C2INAV/20190601/IDL/Navigation_Services.idl
https://www.omg.org/spec/C2INAV/20190601/IDL/Position.idl
https://www.omg.org/spec/C2INAV/20190601/IDL/Reporting.idl
https://www.omg.org/spec/C2INAV/20190601/GRAPHQL/Domain_Model.Common_Types.Coordinates_and_Positions.graphql
s
https://www.omg.org/spec/C2INAV/20190601/GRAPHQL/Domain_Model.Common_Types.graphqls
https://www.omg.org/spec/C2INAV/20190601/GRAPHQL/Domain_Model.Navigation_Domain.Attitude.Ext.graphqls
https://www.omg.org/spec/C2INAV/20190601/GRAPHQL/Domain_Model.Navigation_Domain.graphqls
https://www.omg.org/spec/C2INAV/20190601/GRAPHQL/Domain_Model.Navigation_Domain.Position.Ext.graphqls
https://www.omg.org/spec/C2INAV/20190601/GRAPHQL/Domain_Model.Navigation_Domain.Position.graphqls
https://www.omg.org/spec/C2INAV/20190601/GRAPHQL/Domain_Model.Navigation_Domain.Reporting.Ext.graphqls
https://www.omg.org/spec/C2INAV/20190601/GRAPHQL/Domain_MOdel.Navigation_Domain.Reporting.graphqls
https://www.omg.org/spec/C2INAV/20190601/GRAPHQL/Service_Interfaces.Navigation_Services.graphqls

Informative:

https://www.omg.org/spec/C2INAV/20190601/C2NAV_Model.eap

This OMG document replaces the submission document (c4i/19-06-13, Alpha). It is an OMG Adopted Beta Specification and is currently in the finalization phase. Comments on the content of this document are welcome, and should be directed to issues@omg.org by February 10, 2020.

You may view the pending issues for this specification from the OMG revision issues web page https://issues.omg.org/issues/lists.

The FTF Recommendation and Report for this specification will be published in July 2020. If you are reading this after that date, please download the available specification from the OMG Specifications Catalog.

USE OF SPECIFICATION - TERMS, CONDITIONS & NOTICES

The material in this document details an Object Management Group specification in accordance with the terms, conditions and notices set forth below. This document does not represent a commitment to implement any portion of this specification in any company's products. The information contained in this document is subject to change without notice.

LICENSES

The companies listed above have granted to the Object Management Group, Inc. (OMG) a nonexclusive, royalty-free, paid up, worldwide license to copy and distribute this document and to modify this document and distribute copies of the modified version. Each of the copyright holders listed above has agreed that no person shall be deemed to have infringed the copyright in the included material of any such copyright holder by reason of having used the specification set forth herein or having conformed any computer software to the specification.

Subject to all of the terms and conditions below, the owners of the copyright in this specification hereby grant you a fully-paid up, non-exclusive, nontransferable, perpetual, worldwide license (without the right to sublicense), to use this specification to create and distribute software and special purpose specifications that are based upon this specification, and to use, copy, and distribute this specification as provided under the Copyright Act; provided that: (1) both the copyright notice identified above and this permission notice appear on any copies of this specification; (2) the use of the specifications is for informational purposes and will not be copied or posted on any network computer or broadcast in any media and will not be otherwise resold or transferred for commercial purposes; and (3) no modifications are made to this specification. This limited permission automatically terminates without notice if you breach any of these terms or conditions. Upon termination, you will destroy immediately any copies of the specifications in your possession or control.

PATENTS

The attention of adopters is directed to the possibility that compliance with or adoption of OMG specifications may require use of an invention covered by patent rights. OMG shall not be responsible for identifying patents for which a license may be required by any OMG specification, or for conducting legal inquiries into the legal validity or scope of those patents that are brought to its attention. OMG specifications are prospective and advisory only. Prospective users are responsible for protecting themselves against liability for infringement of patents.

GENERAL USE RESTRICTIONS

Any unauthorized use of this specification may violate copyright laws, trademark laws, and communications regulations and statutes. This document contains information which is protected by copyright. All Rights Reserved. No part of this work covered by copyright herein may be reproduced or used in any form or by any means--graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems--without permission of the copyright owner.

DISCLAIMER OF WARRANTY

WHILE THIS PUBLICATION IS BELIEVED TO BE ACCURATE, IT IS PROVIDED "AS IS" AND MAY CONTAIN ERRORS OR MISPRINTS. THE OBJECT MANAGEMENT GROUP AND THE COMPANIES

OMG's Issue Reporting Procedure

All OMG specifications are subject to continuous review and improvement. As part of this process we encourage readers to report any ambiguities, inconsistencies, or inaccuracies they may find by completing the Issue Reporting Form listed on the main web page http://www.omg.org, under Documents, Report a Bug/Issue.

Contents

Preface

OMG

Founded in 1989, the Object Management Group, Inc. (OMG) is an open membership, not-for-profit computer industry standards consortium that produces and maintains computer industry specifications for interoperable, portable, and reusable enterprise applications in distributed, heterogeneous environments. Membership includes Information Technology vendors, end users, government agencies, and academia.

OMG member companies write, adopt, and maintain its specifications following a mature, open process. OMG's specifications implement the Model Driven Architecture® (MDA®), maximizing ROI through a full-lifecycle approach to enterprise integration that covers multiple operating systems, programming languages, middleware and networking infrastructures, and software development environments. OMG's specifications include: UML® (Unified Modeling Language™); CORBA® (Common Object Request Broker Architecture); CWM™ (Common Warehouse Metamodel); and industry-specific standards for dozens of vertical markets.

More information on the OMG is available at https://www.omg.org/.

OMG Specifications

As noted, OMG specifications address middleware, modeling and vertical domain frameworks. All OMG Specifications are available from the OMG website at:
https://www.omg.org/spec

All of OMG's formal specifications may be downloaded without charge from our website. (Products implementing OMG specifications are available from individual suppliers.) Copies of specifications, available in PostScript and PDF format, may be obtained from the Specifications Catalog cited above or by contacting the Object Management Group, Inc. at:

OMG Headquarters
109 Highland Avenue
Needham, MA 02494
USA
Tel: +1-781-444-0404
Fax: +1-781-444-0320
Email: *pubs@omg.org*

Certain OMG specifications are also available as ISO standards. Please consult http://www.iso.org

Issues

The reader is encouraged to report any technical or editing issues/problems with this document by completing the Issue Reporting Form listed under OMG specifications on the main web page.

This page intentionally left blank.

1 Scope

This specification defines the interface between a Navigation System and Command and Control (C2) functions. It is concerned with the transfer of information regarding the location, movement, orientation and local environment of the platform of which the Navigation System is a part to C2 functions.

2 Conformance

This specification defines conformance points to promote both applicability and interoperability. Services within the specification relating to control of the information reported and specialist parts of the data model are optional. The mandatory services within the interface relate to the transfer of basic information with the default behavior of the Navigation System.

Table 2.1 - Conformance Points for C2INav

Conformance Point	Data Model Packages	Service Methods	Rationale
Basic Reporting	**Navigation Domain, Navigation Domain::Attitude, Navigation Domain::Position**	**write_rotational_attitude, write_position, write_velocity**	Supports a system where the most commonly used attributes are reported with the Navigation System's default behavior without any C2 control
Reporting Control	(Basic Reporting plus) **Navigation Domain::Reporting**	(Basic Reporting plus) **request_navigation_data**	Supports a system where the C2 system can control the Navigation System's reporting of the most commonly used attributes
Specialist Data	All Packages	All Services	Supports the whole specification in the Navigation System's reporting of specialist data subject to control by the C2 System.

3 Normative References

The following normative documents contain provisions which, through reference in this text, constitute provisions of this specification. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply.

- OARIS(formal/2016-03-02)
- DDS (formal/2015-04-10)
- IDL (formal/2018/01/05)
- EVOT (formal/2008-08-01)
- Network Time Protocol (www.ntp.org)
- Precision Time Protocol (IEEE 1588 – http://www.ieee1588.com)

4 Terms and Definitions

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

- AB (Architecture Board)
- API (Application Programming Interface)
- BC (Business Committee)
- BCQ (Business Committee Questionnaire)
- BoD (Board of Directors)
- CCM (CORBA Component Model)
- CMS (Combat Management System)
- CORBA (Common Object Request Broker Architecture)
- CWM (Common Warehouse Metamodel)
- DAIS (Data Acquisition from Industrial Systems)
- DDS (Data Distribution Service)
- EVOT (Enhanced View of Time)
- FTF (Finalization Task Force)
- GLONASS (Global Navigation Satellite System)
- GPS (Global Positioning System)
- IDL (Interface Definition Language)
- IFF (Interrogation, Friend or Foe)
- IIOP (Internet Inter-Orb Protocol)
- IPR (Intellectual Property Right)
- ISO (International Organization for Standardization)

- LOI (Letter of Intent)
- LORAN (Long Range Navigation)
- MDA (Model Driven Architecture)
- METOC (Meteorological and Oceanographic)
- MOF (Meta Object Facility)
- MQS (MQSeries)
- NNSI (Naval Navigation System Interface)
- NS (Naming Service)
- OARIS (Open Architecture Radar Interface Standard)
- ODF (Open Document Format)
- OMG (Object Management Group)
- PIM (Platform Independent Model)
- PSM (Platform Specific Model)
- P&P (Policies and Procedures of the OMG Technical Process)
- RFC (Request For Call)
- RFP (Request For Proposal)
- RTF (Revision Task Force)
- SLAM (Simultaneous Localization and Mapping)
- SOA (Service Oriented Architecture)
- SoaML (Service oriented architecture Modeling Language)
- SOLAS (Safety Of Life At Sea)
- TC (Technology Committee)
- TF (Task Force)
- UML (Unified Modeling Language)
- XMI (XML Metadata Interchange)
- XML (eXtensible Markup Language)

5 Symbols

No special symbols are introduced in this specification.

6 Additional Information

6.1 Acknowledgements

The following companies submitted this specification:

- BAE Systems

This page intentionally left blank.

7 Command and Control Interface for Navigation Systems

7.1 Introduction

The specification is captured as an Enterprise Architect (EA) UML version 2.1 model; this document being automatically generated as a report from the model.

The UML model is an extension of the OARIS model and follows the same hierarchical structure and naming conventions; this model also has dependencies on the Common Types package defined by the OARIS model (as shown in Figure 7.1) as it reuses classes defined there. The classes re-used from the OARIS specification are shown on the class diagrams for the Domain Model. The general purpose error handling service classes defined by OARIS are also re-used.

The C2INav model follows the OARIS model organization: there is a UML PIM data model (Navigation_Domain) under the Domain_Model package and a UML PIM for services and interfaces (Navigation_Services) under the Service_Interfaces package. The data model is organized by primary concerns for navigation information, for instance attitude or orientation versus absolute location in the environment, and also models the means of controlling reporting by a navigation system; a common abstraction for reporting is also modelled. The services and their interfaces model the means for a navigation system to report its data and a C2 system to control the characteristics of the navigation system's reporting.

Additionally OARIS service interfaces defined for the Provide Subsystem Identification and Manage Subsystem Parameters use cases can be used by C2 Systems to discover and configure a Navigation System. (E.g. choose to listen to a specific Navigation System having been informed of its latency and granularity characteristics).

C2INav follows OARIS modelling nomenclature whereby there are interfaces conceptually representing a CMS and a Subsystem for each use case. For C2INav, CMS should be read as representative of C2 Systems in general and Subsystem as being specifically a Navigation System.

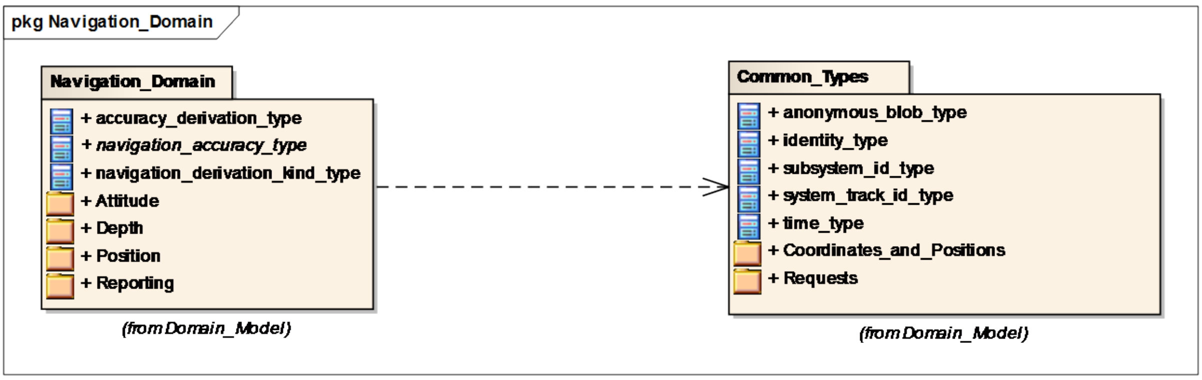

Figure 7.1 Navigation (Package diagram)

7.2 Navigation_Domain

Parent Package: Domain_Model

This package contains the Domain Models for the Navigation services. It is organised according to functionality: i.e. the modelling of attitude (the orientation and offset of the platform relative to its reported position), the platform's position (including velocity and acceleration), the depth of water the platform may be in and the reporting of this information.

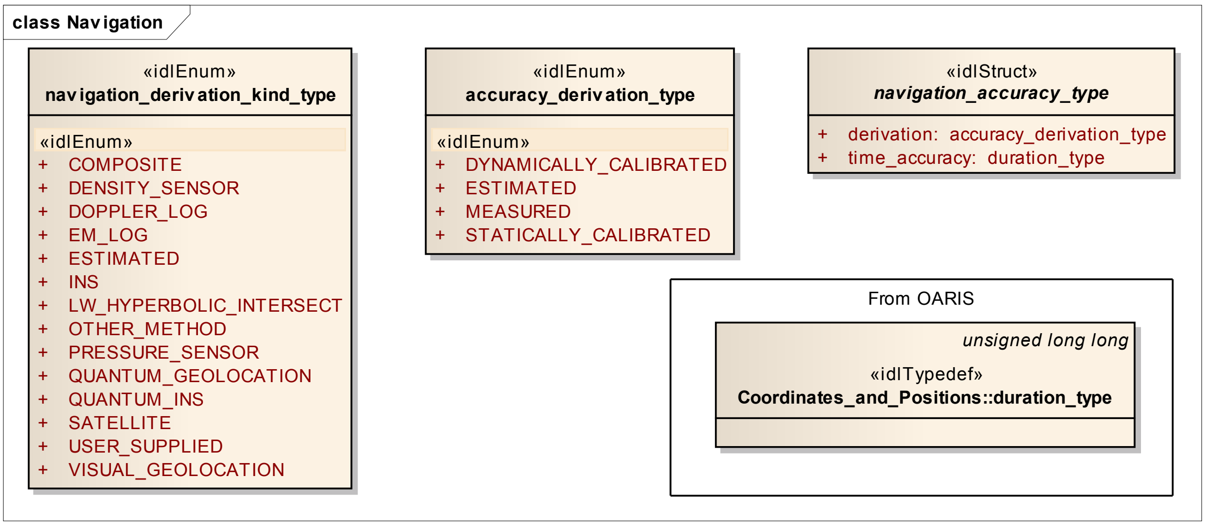

Figure 7.2 Navigation (Logical diagram)

7.2.1 accuracy_derivation_type

Type: IDLEnum

Package: Navigation_Domain

The set of methods describing the provenance of the accuracy values

Table 7.1- Attributes of IDLEnum accuracy_derivation_type

Attribute	Notes
«idlEnum» **DYNAMICALLY_CALIBRATED**	The accuracy values have been calibrated using real data to derive accuracy values for particular instruments given particular environmental conditions - i.e. accuracy values will in general vary over time for the same set of instruments.
«idlEnum» **ESTIMATED**	The values have been set using engineering judgement.
«idlEnum» **MEASURED**	The accuracy values have been measured using some dynamic process that is able to estimate the current performance of the instruments in use.
«idlEnum» **STATICALLY_CALIBRATED**	The accuracy values have been calibrated using real data to derive fixed accuracy values for particular instruments.

7.2.2 navigation_accuracy_type

Type: IDLStruct
Package: Navigation_Domain
A base type for classes that report the accuracy of navigational measurements

Table 7.2- Attributes of IDLStruct navigation_accuracy_type

Attribute	Notes
derivation accuracy_derivation_type	The provenance or method by which the accuracy values have been derived
time_accuracy duration_type	The accuracy (represented as one standard deviation) of the time value.

7.2.3 Navigation_derivation_kind_type

Type: IDLEnum
Package: Navigation_Domain
This is the set of instrument types and other means by which navigation information can be derived.

Table 7.3 - Attributes of IDLEnum navigation_derivation_kind_type

Attribute	Notes
«idlEnum» **COMPOSITE**	Information derived by fusing data from more than one of these types of derivation source.
«idlEnum» **DENSITY_SENSOR**	Information derived from an instrument that measures the (subsurface) sea or air density to estimate depth or altitude.
«idlEnum» **DOPPLER_LOG**	Information derived from an instrument that exploits the Doppler effect to measure speed relative to the immediate environment, particularly water.
«idlEnum» **EM_LOG**	Information derived from an instrument that exploits the electromagnetic dynamo effect (conductor moving through an electromagnetic field produces a proportional voltage) to measure speed relative to the immediate environment, particularly water.
«idlEnum» **ESTIMATED**	Information is estimated from previously measured values (e.g. dead-reckoning).
«idlEnum» **INS**	Information derived from instruments based on an Inertial Navigation System (e.g. Gyroscopes and Accelerometers)
«idlEnum» **LW_HYPERBOLIC_INTERSECT**	Navigation based on the intersection of hyperbolic curves derived from long wave radio signals from known ground stations with repeaters - e.g. LORAN B and C

Attribute	Notes
«idlEnum» **OTHER_METHOD**	Information has been derived using some other technology
«idlEnum» **PRESSURE_SENSOR**	Information derived from an instrument that measures the (subsurface) sea or air pressure to estimate depth or altitude.
«idlEnum» **QUANTUM_GEOLOCATION**	Information derived by sensing the Earth's gravitational and/or magnetic field and/or their gradients using single particle systems that exploit quantum effects.
«idlEnum» **QUANTUM_INS**	Information derived from instruments based on an Inertial Navigation System that measures acceleration using single particle systems that exploit quantum techniques.
«idlEnum» **SATELLITE**	Information derived from a satellite-based navigation system (e.g. GPS and GLONASS)
«idlEnum» **USER_SUPPLIED**	Information has been supplied by the user (e.g. manual entry from a non-integrated system).
«idlEnum» **VISUAL_GEOLOCATION**	Information derived by sensing the external environment and resolving position and orientation with reference to external data such as charts (e.g. SLAM techniques).

7.2.4 Attitude

Parent Package: Navigation_Domain

This package contains classes to model the attitude of the platform. That is the orientation and offset of the platform relative to its reported position.

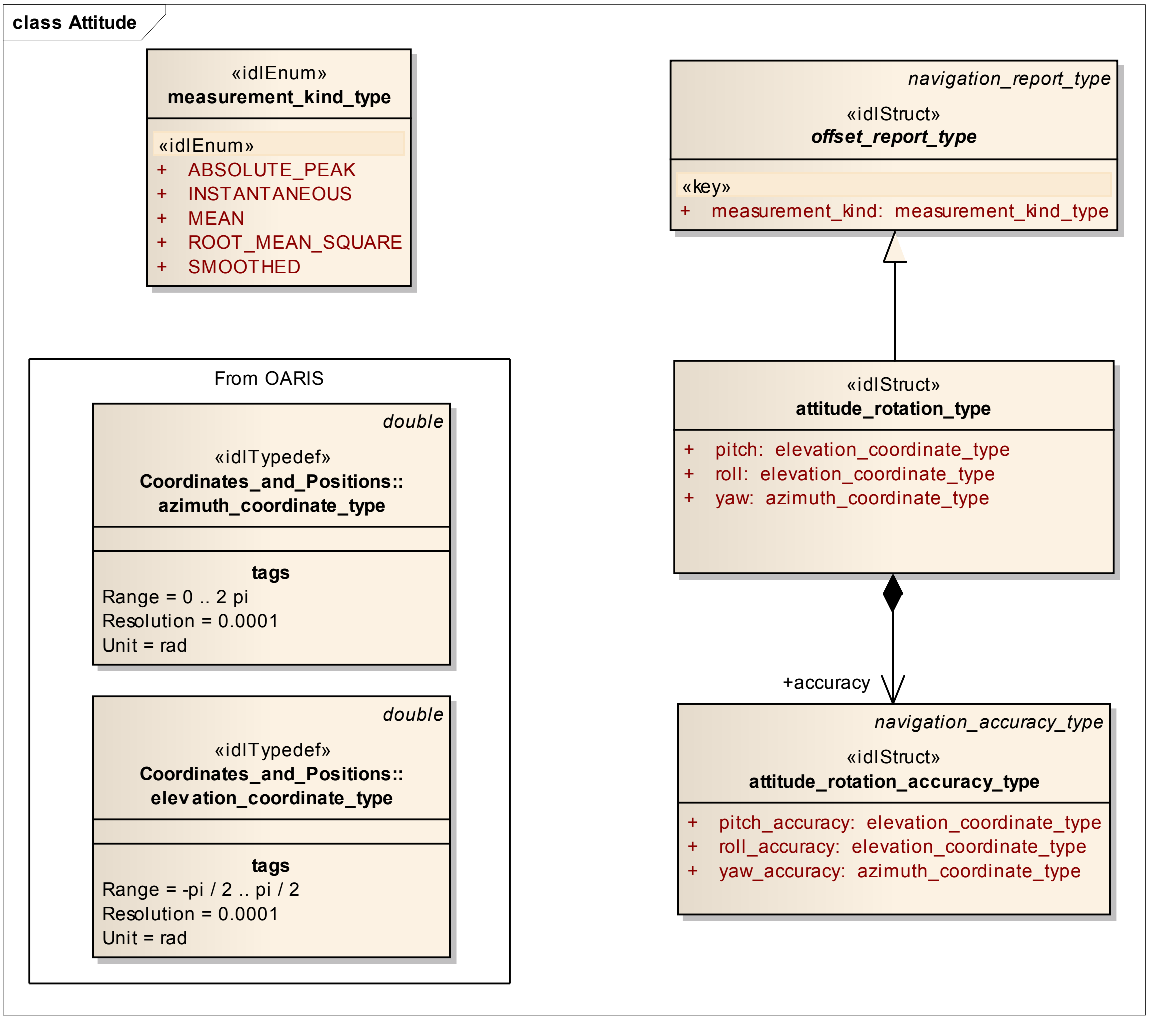

Figure 7.3 Attitude (Logical diagram)

7.2.4.1 attitude_rotation_accuracy_type

Type: IDLStruct navigation_accuracy_type

Package: Attitude

This class encapsulates the error estimates associated with the platform's rotational attitude values. Accuracies are reported as one standard deviation.

Table 7.4 - Attributes of IDLStruct attitude_rotation_accuracy_type

Attribute	Notes
pitch_accuracy elevation_coordinate_type	The accuracy of the pitch value to one standard deviation,
roll_accuracy elevation_coordinate_type	The accuracy of the roll value to one standard deviation,
yaw_accuracy azimuth_coordinate_type	The accuracy of the yaw value to one standard deviation,

7.2.4.2 attitude_rotation_type

Type: IDLStruct offset_report_type

Package: Attitude

This class encapsulates the instantaneous rotation of the platform from its nominal, at-rest orientation

Non-normative: typically due to the variable motion of the environment - sea, air, etc. - through which it is travelling.

Table 7.5 - Attributes of IDLStruct attitude_rotation_type

Attribute	Notes
pitch elevation_coordinate_type	The pitch of the platform, relative to its platform reference point in a vertical plane. It is the clockwise angle of rotation around the lateral axis (towards starboard/right) through the platform reference point. For sea systems, the angle from horizontal to the bow; for air systems, the angle from horizontal to the nose; for land systems, the angle from horizontal to the front.
roll elevation_coordinate_type	The roll of the platform, relative to its platform reference point in a vertical plane. It is the angle of rotation about the longitudinal axis through the platform reference point (front-to-back). The roll angle is defined as that of the at-rest horizontal through the platform's reference on the starboard side for sea and air systems and on the right (forward facing) for land systems.
yaw azimuth_coordinate_type	The yaw of the platform, relative to its platform reference point in a horizontal plane. It is the angle of rotation about the vertical axis through the platform reference point (top-to-bottom) relative to the platform's course. For sea systems, the angle to the bow; for air systems, the angle to the nose; for land systems, the angle to the front.

7.2.4.3 measurement_kind_type

Type: IDLEnum
Package: Attitude

The kind of measurement relating to the statistical process applied to the quantities in question over time.

Table 7.6- Attributes of IDLEnum measurement_kind_type

Attribute	Notes
«idlEnum» **ABSOLUTE_PEAK**	The maximum of the absolute value of the raw measurements over a complete cycle (a complete cycle is defined as the interval between local maxima).
«idlEnum» **INSTANTANEOUS**	The raw measurement at the stated time

Attribute	Notes
«idlEnum» **MEAN**	The arithmetic mean (average) of the raw measurements over a complete cycle (a complete cycle is defined as the interval between local maxima).
«idlEnum» **ROOT_MEAN_SQUARE**	The root mean square average of the raw measurements over a complete cycle (a complete cycle is defined as the interval between local maxima).
«idlEnum» **SMOOTHED**	The system's best estimate for the current value of the quantities based on recent raw measurements.

7.2.4.4 offset_report_type

Type: IDLStruct navigation_report_type
Package: Attitude
This is the base type for the reporting of all information that is an offset from the platform's mean reported motion and its at-rest orientation; hence this includes attitude information. These reports are keyed on the statistical kind of measurement, enabling different views of cyclical motion to be reported.

Table 7.7- Attributes of IDLStruct offset_report_type

Attribute	Notes
«key» **measurement_kind** measurement_kind_type	The kind of measurement being reported.

7.2.4.5 Attitude.Ext

Parent Package: Attitude

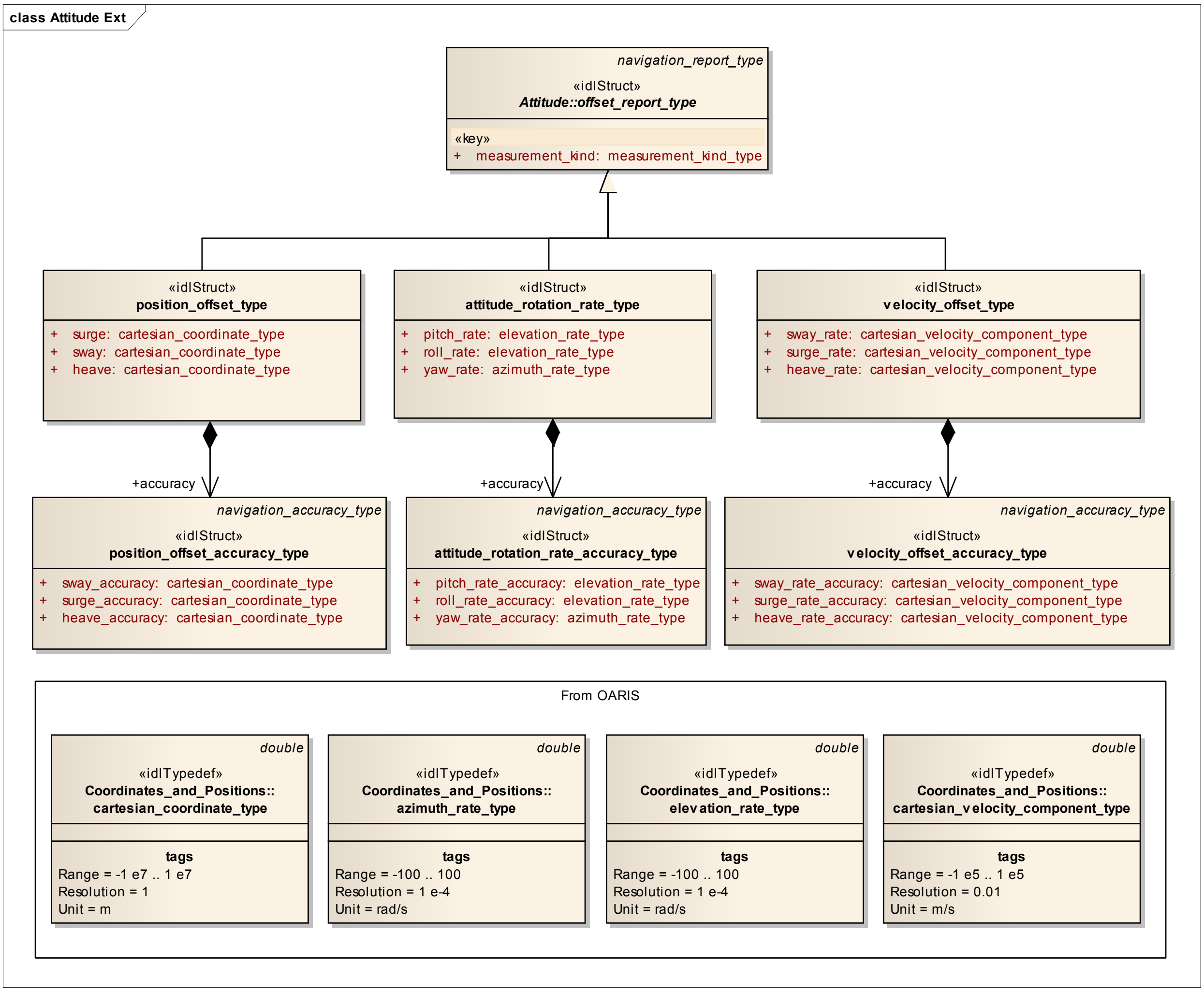

Figure 7.4 Attitude Ext (Logical diagram)

7.2.4.5.1 attitude_rotation_rate_accuracy_type

Type: IDLStruct navigation_accuracy_type
Package: Ext

This class encapsulates the error estimates associated with the platform's rotational attitude rate values. Accuracies are reported as one standard deviation.

Table 7.8 - Attributes of IDLStruct attitude_rotation_rate_accuracy_type

Attribute	Notes
pitch_rate_accuracy elevation_rate_type	The accuracy of the pitch rate value to one standard deviation,
roll_rate_accuracy elevation_rate_type	The accuracy of the roll rate value to one standard deviation,
yaw_rate_accuracy azimuth_rate_type	The accuracy of the yaw rate value to one standard deviation,

7.2.4.5.2 attitude_rotation_rate_type

Type: IDLStruct offset_report_type
Package: Ext

This class encapsulates the instantaneous rate of rotation of the platform (relative to the Earth).

Table 7.9 - Attributes of IDLStruct attitude_rotation_rate_type

Attribute	Notes
pitch_rate elevation_rate_type	The pitch rate of the platform, relative to its platform reference point in a vertical plane. It is the rate of change of angle of rotation around the lateral axis (towards starboard/right) through the platform reference point. For sea systems, the angle to the bow; for air systems, the angle to the nose; for land systems, the angle to the front.
roll_rate elevation_rate_type	The roll rate of the platform, relative to its platform reference point in a vertical plane. It is the rate of change of the angle of rotation about the longitudinal axis through the platform reference point (front-to-back). The roll angle is defined as that of the at-rest horizontal through the platform's reference on the starboard side for sea and air systems and on the right (forward facing) for land systems.
yaw_rate azimuth_rate_type	The yaw rate of the platform, relative to its platform reference point in a horizontal plane. It is the rate of change of the angle of rotation about the vertical axis through the platform reference point (top-to-bottom) relative to the platform’s course. For sea systems, the angle to the bow; for air systems, the angle to the nose; for land systems, the angle to the front.

7.2.4.5.3 position_offset_accuracy_type

Type: IDLStruct navigation_accuracy_type
Package: Ext

This class encapsulates the error estimates associated with the platform's attitude offset values.
Accuracies are reported as one standard deviation.

Table 7.10 - Attributes of IDLStruct position_offset_accuracy_type

Attribute	Notes
sway_accuracy cartesian_coordinate_type	The accuracy of the lateral value to one standard deviation,
surge_accuracy cartesian_coordinate_type	The accuracy of the longitudinal value to one standard deviation,
heave_accuracy cartesian_coordinate_type	The accuracy of the vertical value to one standard deviation,

7.2.4.5.4 position_offset_type

Type: IDLStruct offset_report_type
Package: Ext

This class encapsulates the instantaneous offset of the platform from its mean reported motion. Non-normative: typically due to the variable motion of the environment - sea, air, etc. - through which it is travelling.

Table 7.11 - Attributes of IDLStruct position_offset_type

Attribute	Notes
sway cartesian_coordinate_type	The instantaneous offset from mean motion on the lateral horizontal axis through the platform reference point. For sea and air systems port-starboard - starboard positive For land systems side-to-side - right positive when facing forwards.
surge cartesian_coordinate_type	The instantaneous offset from mean motion on the longitudinal horizontal axis through the platform reference point. For sea systems stern-bow - bow positive For air systems tail-nose - nose positive For land systems back-to-front - front positive.
heave cartesian_coordinate_type	The instantaneous offset from mean motion on the vertical axis through the platform reference point. For sea systems keel-mast - mast positive For air and land systems bottom-to-top - top positive.

7.2.4.5.5 velocity_offset_accuracy_type

Type: IDLStruct navigation_accuracy_type
Package: Ext

This class encapsulates the error estimates associated with the platform's attitude offset rate values. Accuracies are reported as one standard deviation.

Table 7.12 - Attributes of IDLStruct velocity_offset_accuracy_type

Attribute	Notes
sway_rate_accuracy cartesian_velocity_component_type	The accuracy of the lateral rate value to one standard deviation,
surge_rate_accuracy cartesian_velocity_component_type	The accuracy of the longitudinal rate value to one standard deviation,
heave_rate_accuracy cartesian_velocity_component_type	The accuracy of the vertical rate value to one standard deviation,

7.2.4.5.6 velocity_offset_type

Type: IDLStruct offset_report_type
Package: Ext

This class encapsulates the instantaneous rate of change of the offset of the platform from its mean reported motion.

Table 7.13 - Attributes of IDLStruct velocity_offset_type

Attribute	Notes
sway_rate cartesian_velocity_component_type	The instantaneous rate of change of the offset from mean motion on the lateral horizontal axis through the platform reference point. For sea and air systems port-starboard - starboard positive For land systems side-to-side - right positive when facing forwards.
surge_rate cartesian_velocity_component_type	The instantaneous rate of change of the offset from mean motion on the longitudinal horizontal axis through the platform reference point. For sea systems stern-bow - bow positive For air systems tail-nose - nose positive For land systems back-to-front - front positive.
heave_rate cartesian_velocity_component_type	The instantaneous rate of change of the offset from mean motion on the vertical axis through the platform reference point. For sea systems keel-mast - mast positive For air and land systems bottom-to-top - top positive.

7.2.5 Depth

Parent Package: Navigation_Domain

This package contains classes to model of the depth of water the platform may be in.

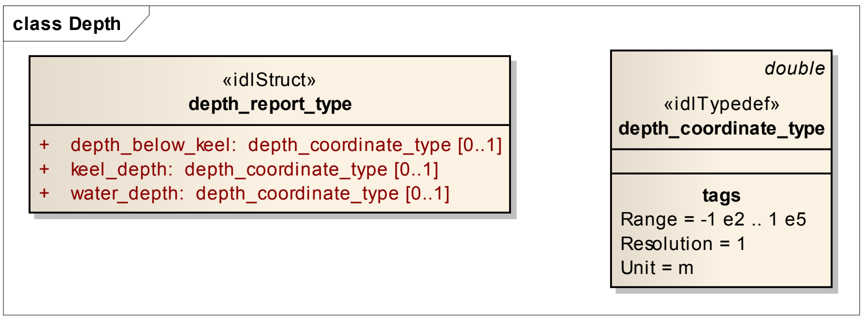

Figure 7.5 Depth (Logical diagram)

7.2.5.1 depth_coordinate_type

Type: IDLTypeDef double

Package: Depth

Measured positive down in meters. c.f. altitude_coordinate_type in OARIS Common_Types package.

Range = -1 e2 .. 1 e5

Resolution = 1

Unit = m

7.2.5.2 depth_report_type

Type: IDLStruct

Package: Depth

Used by waterborne craft to report depth information.

Table 7.14 - Attributes of IDLStruct depth_report_type

Attribute	Notes
depth_below_keel depth_coordinate_type [0..1]	The depth of the bed below the keel.
keel_depth depth_coordinate_type [0..1]	The depth of the keel below the surface of the water.
water_depth depth_coordinate_type [0..1]	The depth of the bed below the water's surface.

7.2.6 Position

Parent Package: Navigation_Domain

This package contains classes to model the platform's position including height/depth/altitude, velocity and acceleration.

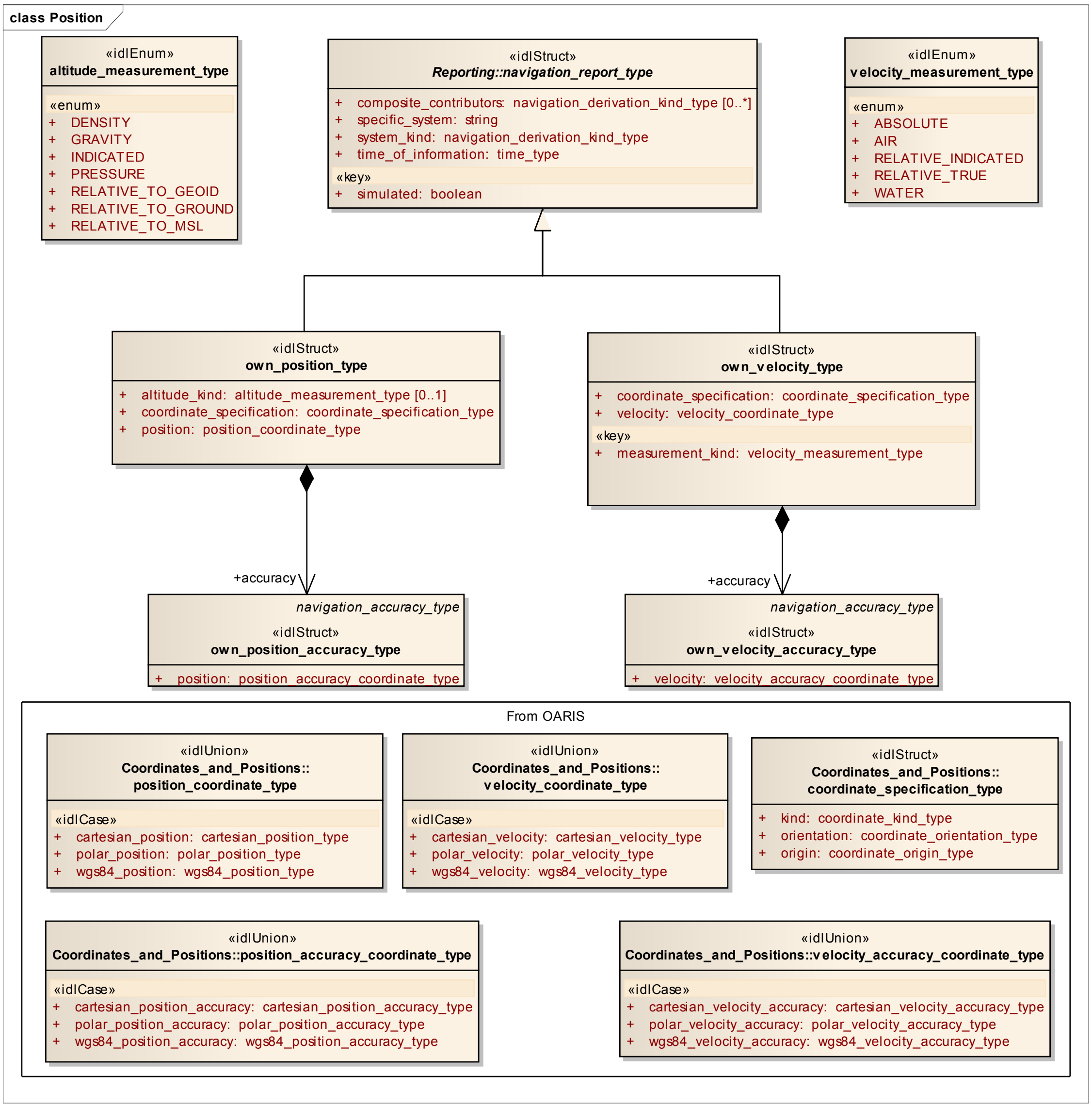

Figure 7.6 Position (Logical diagram)

7.2.6.1 altitude_measurement_type

Type: IDLEnum

Package: Position

This class indicates the semantics of the height attribute. It describes how the reported height (or depth) has been derived.

Table 7.15 - Attributes of IDLEnum altitude_measurement_type

Attribute	Notes
«enum» **DENSITY**	It is reporting a measurement based on local density (of air or water).
«enum» **GRAVITY**	It is reporting a measurement based on local gravity.
«enum» **INDICATED**	As reported by the platform's own altimeter (or depth sensor). Typically, but not necessarily this is pressure related. The precise option rather than this one should be reported if known.
«enum» **PRESSURE**	It is derived from a pressure sensor.
«enum» **RELATIVE_TO_GEOID**	It is reporting a measurement made relative to the GEOID (WGS84) - e.g. using a satellite navigation system.
«enum» **RELATIVE_TO_GROUND**	The distance to the ground below (or above for underground systems) is being measured and reported.
«enum» **RELATIVE_TO_MSL**	It is reporting a measurement made relative to mean sea level.

7.2.6.2 own_position_accuracy_type

Type: IDLStruct navigation_accuracy_type

Package: Position

The accuracy of the platform's own position report.

Table 7.16 - Attributes of IDLStruct own_position_accuracy_type

Attribute	Notes
position position_accuracy_coordinate_type	The accuracy of the reported position in the chosen coordinate system for reporting. This should be the same choice as for the position itself.

7.2.6.3 own_position_type

Type: IDLStruct navigation_report_type

Package: Position

The platform's own position report.

Table 7.17 - Attributes of IDLStruct own_position_type

Attribute	Notes
altitude_kind altitude_measurement_type [0..1]	Describes the semantics of the position's altitude attribute. Optional: omit only if altitude is not reported.

Attribute	Notes
coordinate_specification coordinate_specification_type	The specification of the coordinate system used for reporting own position and its accuracy. In most use cases Cartesian or WGS84, Earth Referenced choices are expected; in some use cases other choices for reporting relative to some known datum could be used.
position position_coordinate_type	The position of the reporting platform in the chosen coordinate system for reporting.

7.2.6.4 own_velocity_accuracy_type

Type: IDLStruct navigation_accuracy_type
Package: Position
The accuracy of the platform's own velocity report.

Table 7.18 - Attributes of IDLStruct own_velocity_accuracy_type

Attribute	Notes
velocity velocity_accuracy_coordinate_type	The accuracy of the reporting platform's velocity with reference to the coordinate system used for reporting.

7.2.6.5 own_velocity_type

Type: IDLStruct navigation_report_type
Package: Position
The platform's own velocity report.

Table 7.19 - Attributes of IDLStruct own_velocity_type

Attribute	Notes
coordinate_specification coordinate_specification_type	The specification of the coordinate system used for reporting own velocity and its accuracy. In most use cases Cartesian or WGS84, Earth Referenced choices are expected; in some use cases other choices for reporting relative to some known datum could be used.
«key» **measurement_kind** velocity_measurement_type	The definition of the velocity being measured.
velocity velocity_coordinate_type	The velocity of the reporting platform with reference to the chosen coordinate system for reporting.

7.2.6.6 velocity_measurement_type

Type: IDLEnum
Package: Position
This class defines what it is that is having its velocity measured.

Table 7.20 - Attributes of IDLEnum velocity_measurement_type

Attribute	Notes
«enum» **ABSOLUTE**	The measurement is of absolute velocity (i.e. relative to the Earth).
«enum» **AIR**	The measurement is of the air in the local environment itself (i.e. the wind speed). This is only to be reported as a true value; i.e. it is not be based on a pressure-based 'indicated' sensor reading for relative air speed that isn't corrected for density.
«enum» **RELATIVE_INDICATED**	The measurement is of velocity relative to the environment (i.e. water or air) using an indirect approximation such as air pressure.
«enum» **RELATIVE_TRUE**	The measurement is of velocity relative to the environment (i.e. water or air) using a method that is not subject to systematic approximation error as is the case with 'Indicated Air Speed' as measured by a pressure sensor.
«enum» **WATER**	The measurement is of the water current in the local environment itself (i.e. the movement of the water).

7.2.6.7 Position Ext

Parent Package: Position

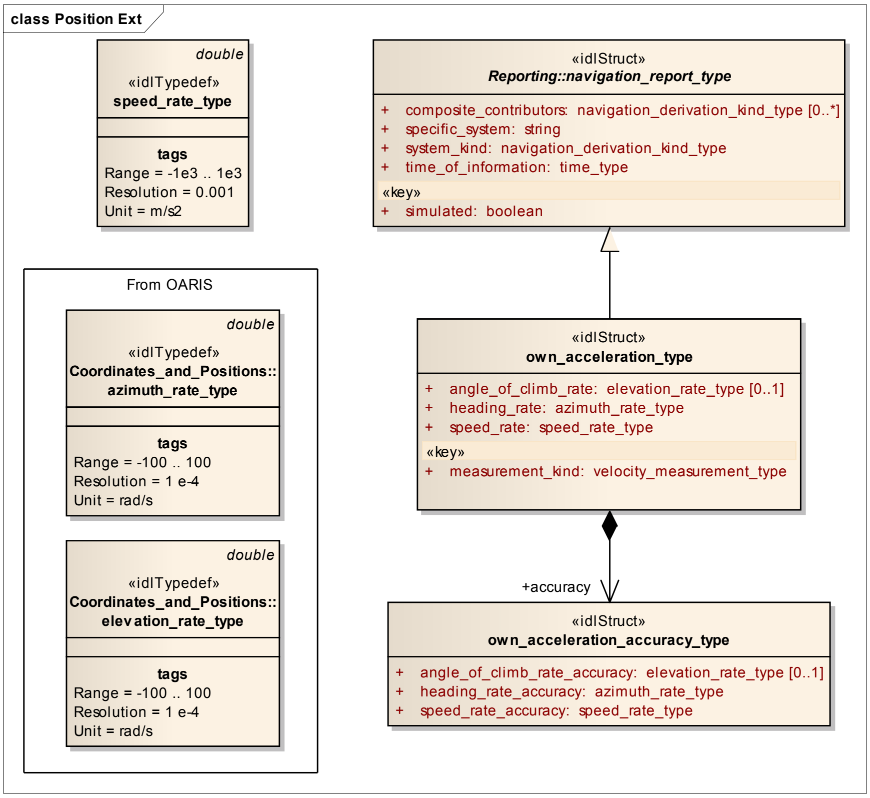

Figure 7.7 Position Ext (Logical diagram)

7.2.6.7.1 own_acceleration_accuracy_type

Type: IDLStruct
Package: Ext

The accuracy of the platform's own acceleration report.

Table 7.21 - Attributes of IDLStruct own_acceleration_accuracy_type

Attribute	Notes
angle_of_climb_rate_accuracy elevation_rate_type [0..1]	The accuracy of the angle of climb rate - 1 standard deviation
heading_rate_accuracy azimuth_rate_type	The accuracy of the heading rate - 1 standard deviation
speed_rate_accuracy speed_rate_type	The accuracy of the speed rate - 1 standard deviation

7.2.6.7.2 own_acceleration_type

Type: IDLStruct navigation_report_type
Package: Ext

The platform's reporting of its change in velocity

Table 7.22 - Attributes of IDLStruct own_acceleration_type

Attribute	Notes
angle_of_climb_rate elevation_rate_type [0..1]	The rate at which the angle of climb is changing
heading_rate azimuth_rate_type	The rate at which the heading is changing
«key» **measurement_kind** velocity_measurement_type	The definition of the acceleration (change in velocity) being measured.
speed_rate speed_rate_type	The rate at which the speed is changing

7.2.6.7.3 speed_rate_type

Type: IDLTypeDef double
Package: Ext

The rate of change of speed in meters per second-squared

Range = -1e3 .. 1e3

Resolution = 0.001
Unit = m/s2

7.2.7 Reporting

Parent Package: Navigation_Domain

This package contains classes that provide a common abstraction for the reporting of navigation information.

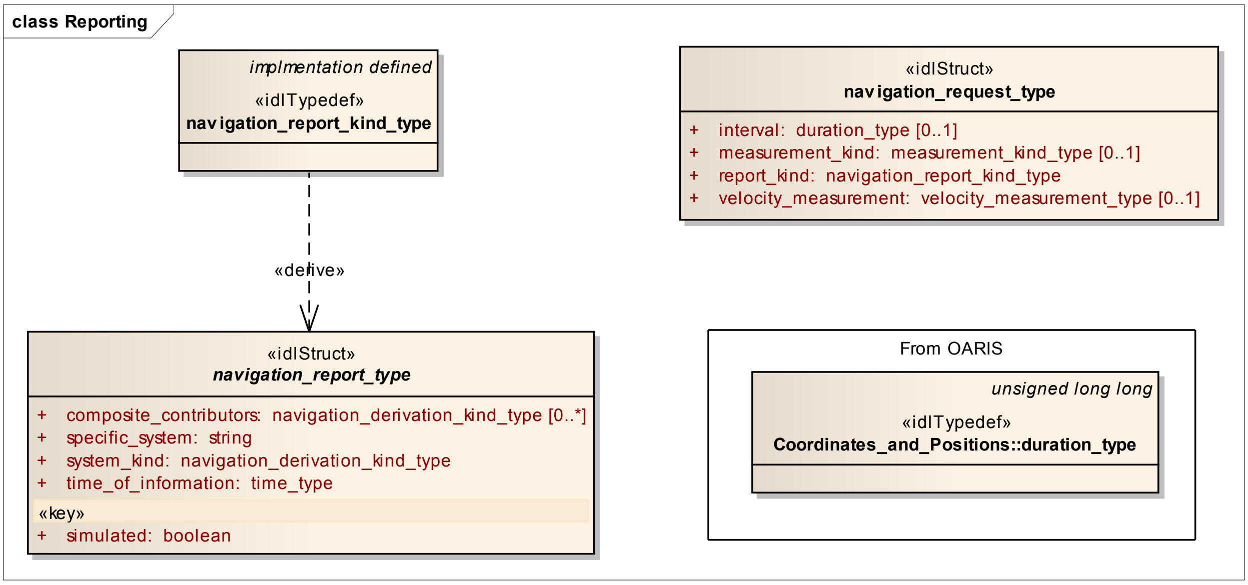

Figure 7.8 Reporting (Logical diagram)

7.2.7.1 navigation_report_kind_type

Type: IDLTypeDef implmentation defined
Package: Reporting

This class is used in service selection to specify the type of navigation report being requested. Its implementation is determined by the PSM mapping.

7.2.7.2 navigation_report_type

Type: IDLStruct
Package: Reporting

A base type for classes that report navigational measurements

Table 7.23 - Attributes of IDLStruct navigation_report_type

Attribute	Notes
composite_contributors navigation_derivation_kind_type [0..*]	An optional set of contributing sensor kinds that have been used to derive the report. This set is defined when the system_kind is COMPOSITE.
«key» **simulated** boolean	Whether the information has been simulated - e.g. for operator training.
specific_system string	The specific system employed - e.g. GPS, LORAN-B
system_kind navigation_derivation_kind_type	The generic type of navigation system used.
time_of_information time_type	The time for which the report values are valid.

7.2.7.3 navigation_request_type

Type: IDLStruct
Package: Reporting

This class is used to construct requests for data to a navigation system and also to cancel such requests.

When used to cancel requests the interval should be omitted by the C2 System and ignored by the Navigation System.

Table 7.24 - Attributes of IDLStruct navigation_request_type

Attribute	Notes
interval duration_type [0..1]	The nominal interval between reports being requested. Omit to request at the default rate for the navigation system.
measurement_kind measurement_kind_type [0..1]	The kind of statistical reporting of the measurements being requested. Omit to request all measurement kinds supported by the navigation system.
report_kind navigation_report_kind_type	The type of data to be reported in response to the request.
velocity_measurement velocity_measurement_type [0..1]	The type of velocity measurement requested. This is only valid if the report kind indicates velocity. Omit to request all velocity measurements supported by the navigation system.

7.2.7.4 Reporting.Ext

Parent Package: Reporting

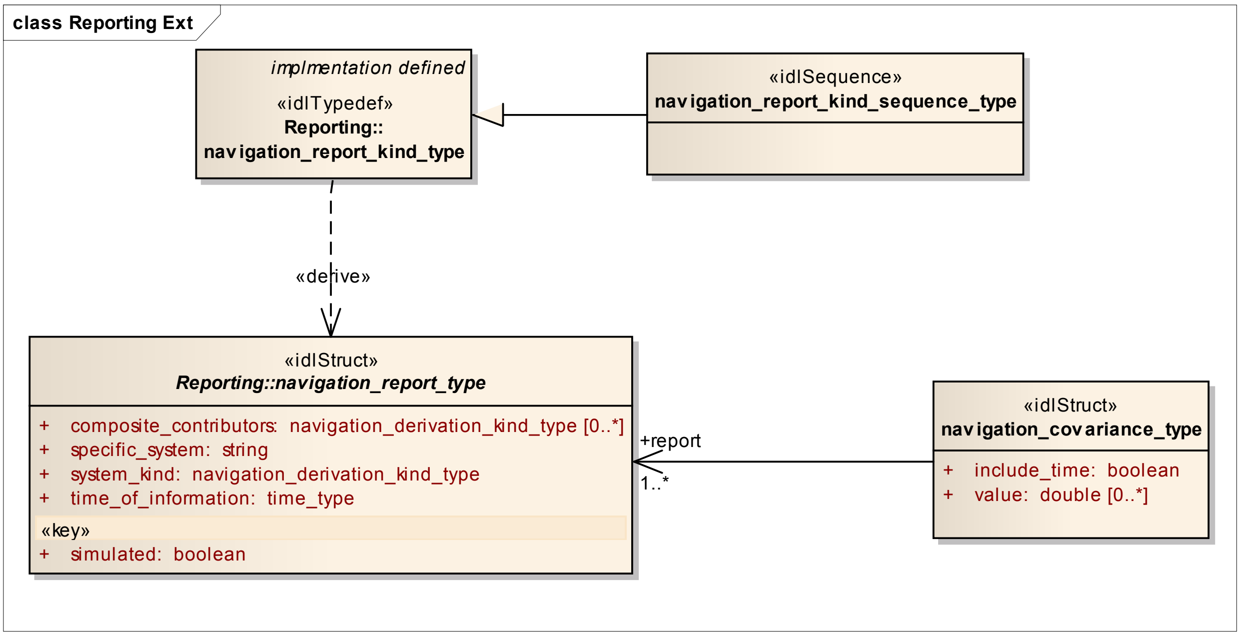

Figure 7.9 Reporting Ext (Logical diagram)

7.2.7.4.1 navigation_covariance_type

Type: IDLStruct

Package: Ext

This class encapsulates the covariance between the measurements in one or more navigation_report_instances. The rows and columns of the triangular covariance matrix relate to the (3) quantities from the first instance, optionally the (3) quantities from each of the further instances and finally, optionally, time.

This, for instance allows the covariance between rotational attitude and position to be represented or the covariance between rotational attitude, position, all their rates of change and time.

Table 7.25 - Attributes of IDLStruct navigation_covariance_type

Attribute	Notes
include_time boolean	Whether time is included in the covariance - it is always represented in the last row and column.
value double [0..*]	The content of the triangular covariance matrix omitting symmetric (duplicate) values. Valid lengths are sum(n=1..*; 3n) and sum(n=1..*; 3n+1). The sequence starts 6, 10, 21, 28, 45, ...

7.2.7.4.2 navigation_report_kind_sequence_type

Type: IDLSequence navigation_report_kind_type
Package: Ext

A sequence of navigation report kinds. It is expected that this class will map to implementation specific mechanisms in PSMs.

7.3 Navigation Services

Parent Package: Service_Interfaces

The Navigation Services define the methods for a Navigation System to report its data to a C2 System and for a C2 System to control the rate and content of the data reported by a Navigation System. The usage of these services is presented in a series of sequence diagrams aligned with this specification's conformance points from section **Error! Reference source not found.**; basic flows describe normal operation and alternative flows describe error handling.

Navigation_CMS and Navigation_Sub are the interfaces to be implemented by C2 (including CMS) and Navigation System components respectively.

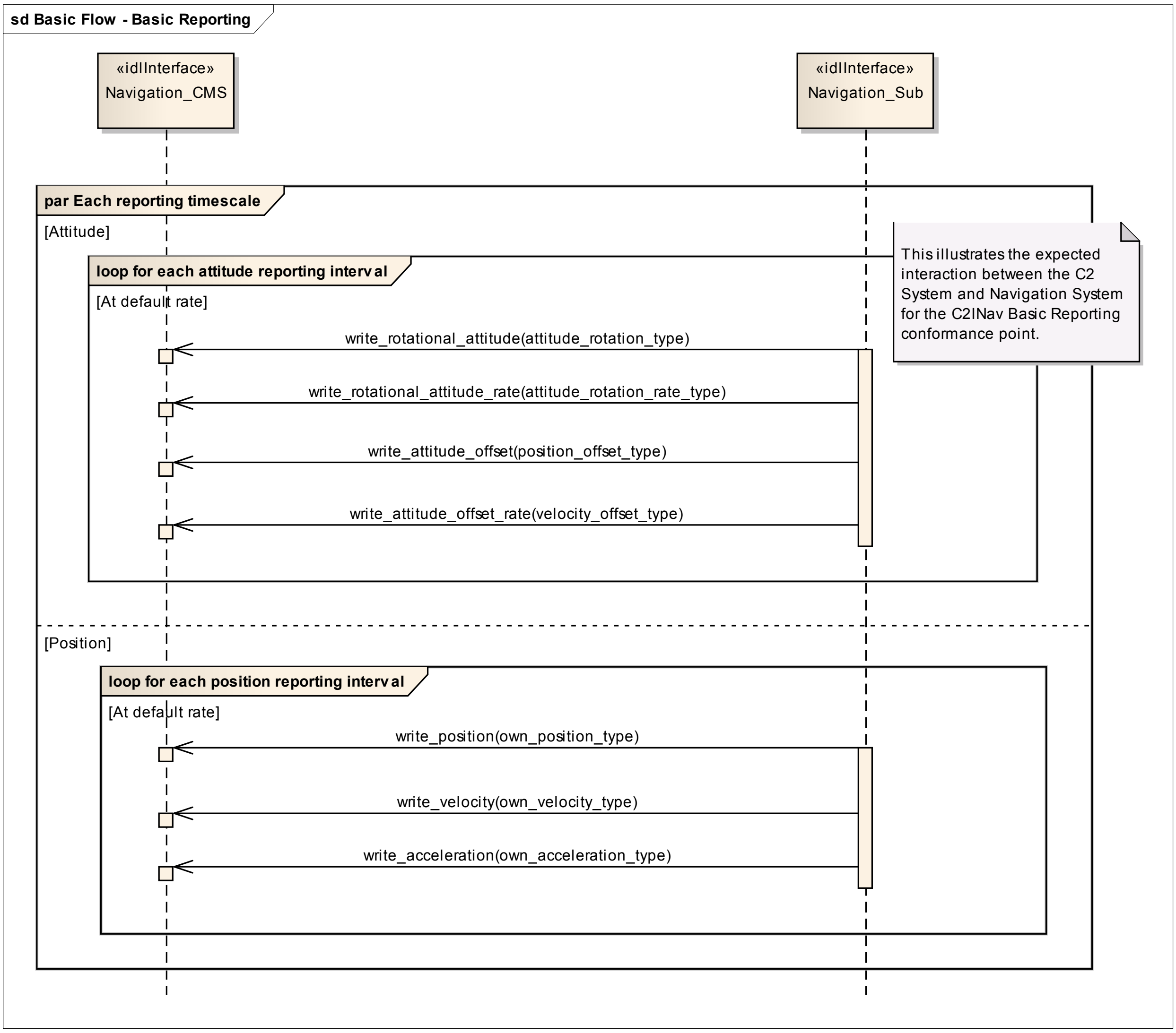

Figure 7.10 Basic Flow - Basic Reporting (Sequence diagram)

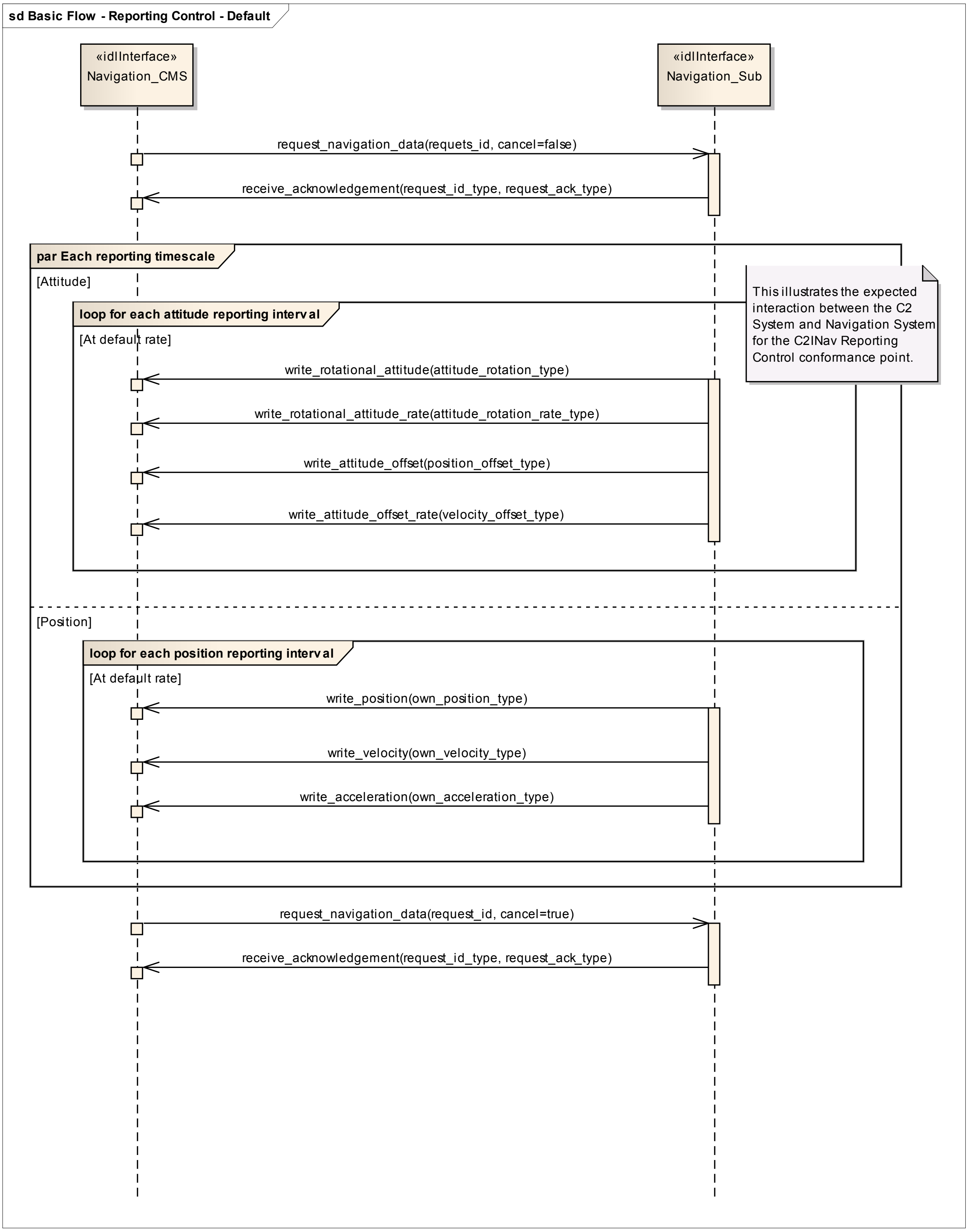

Figure 7.11 Basic Flow - Reporting Control - Default (Sequence diagram)

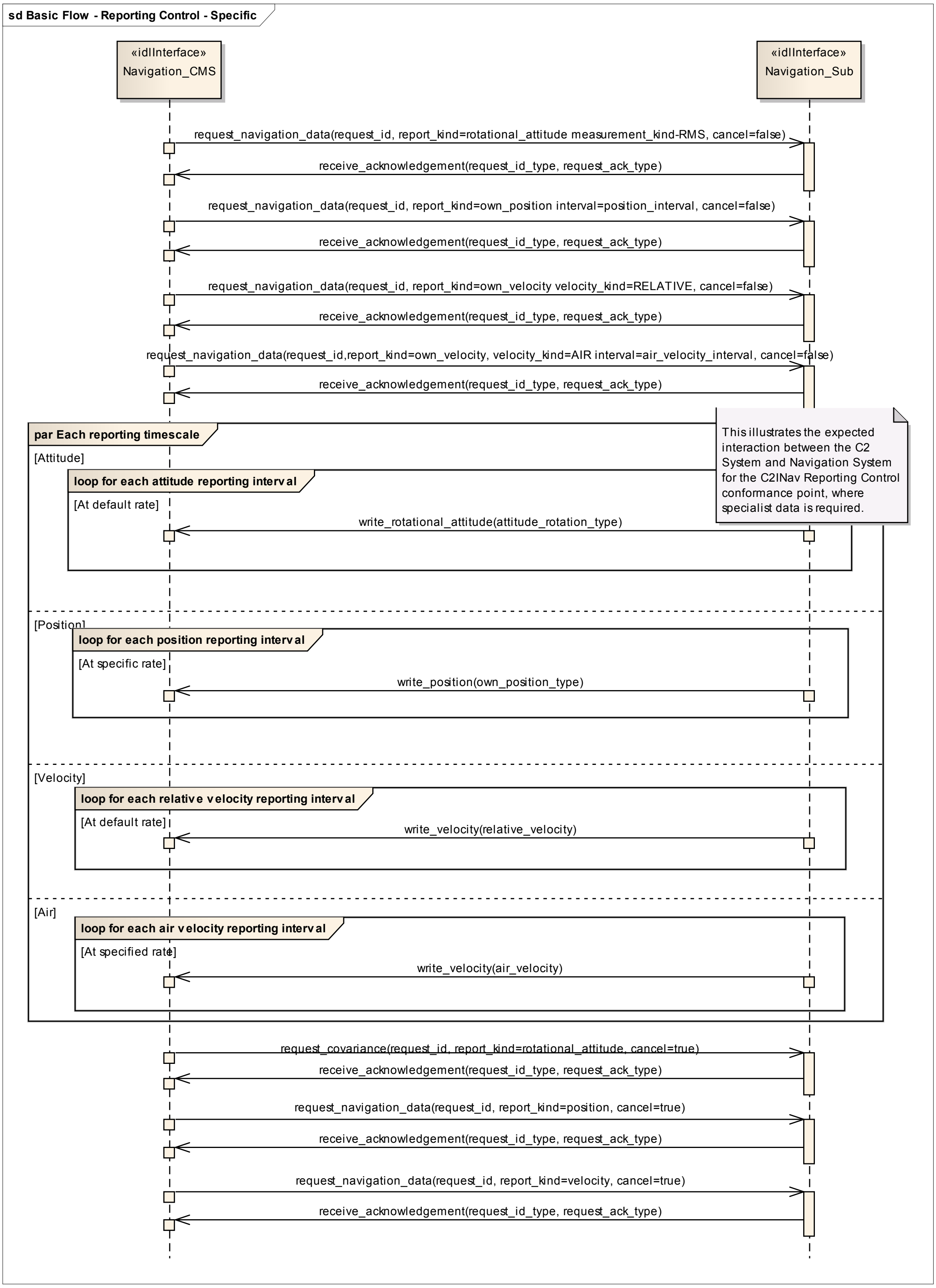

Figure 7.12 Basic Flow - Reporting Control - Specific (Sequence diagram)

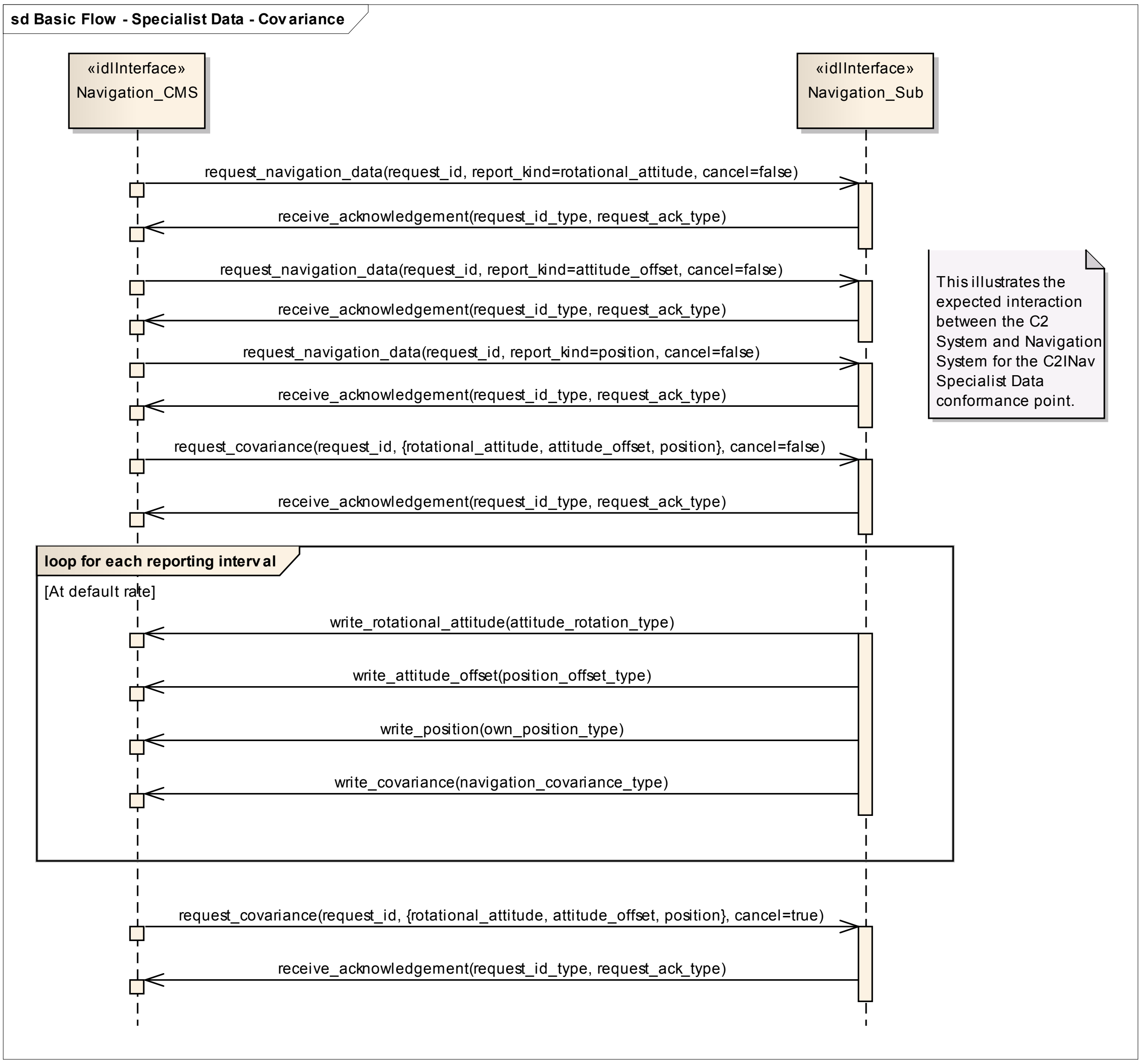

Figure 7.13 Basic Flow - Specialist Data - Covariance (Sequence diagram)

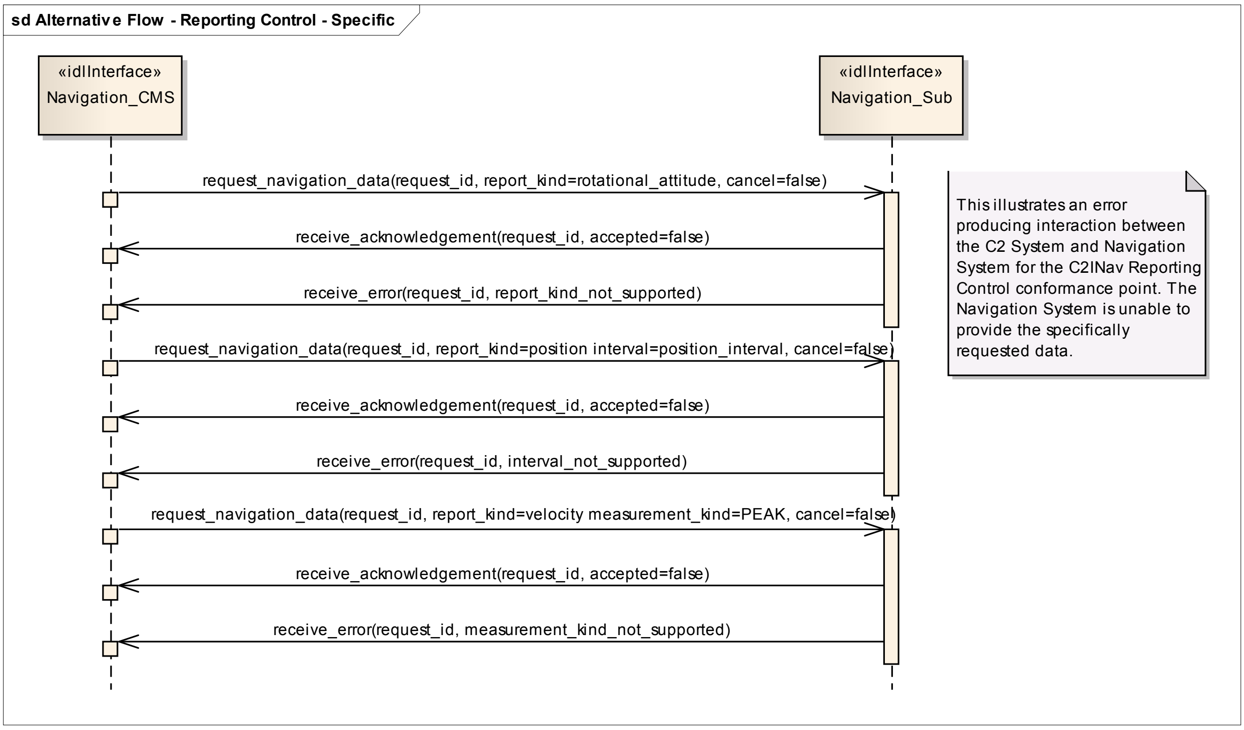

Figure 7.14 Alternative Flow - Reporting Control - Specific (Sequence diagram)

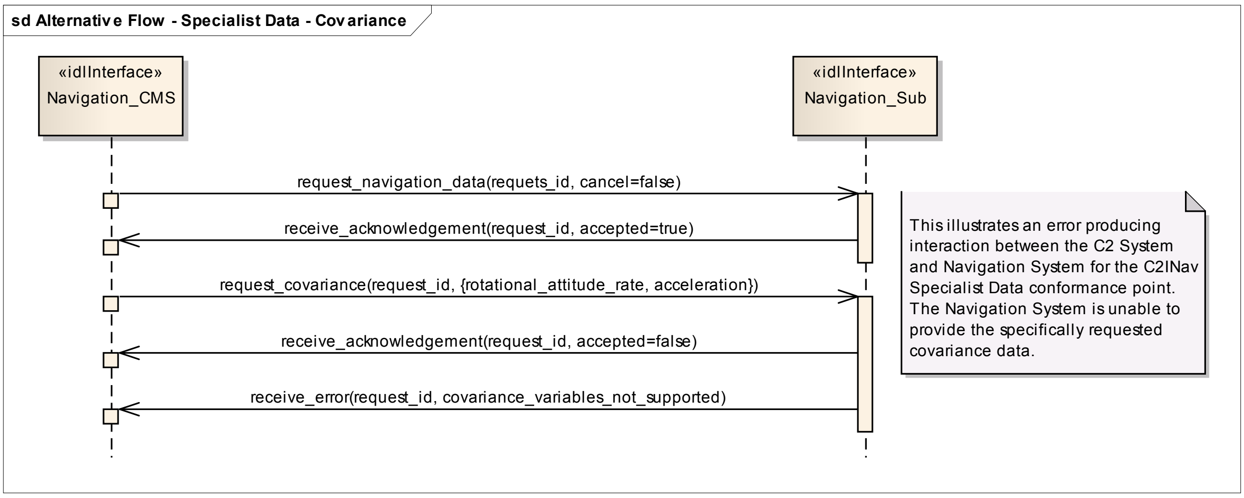

Figure 7.15 Alternative Flow - Specialist Data - Covariance (Sequence diagram)

Figure 7.16 Service Interface Navigation (Logical diagram)

7.3.1 Navigation_CMS

Type: IDLInterface common_use_case_interface
Package: Navigation_Services
The interface implemented by C2 components to consume C2INav services

Table 7.26 - Methods of IDLInterface Navigation_CMS

Method	Notes	Parameters
write_acceleration()		own_acceleration_type **acceleration**
write_attitude_offset()		position_offset_type **offset**
write_attitude_offset_rate()		velocity_offset_type **rate**
write_covariance()	The reported covariance between the selected quantities.	navigation_covariance_type **covariance**
write_depth()		depth_report_type **depth**
write_position()		own_position_type **position**
write_rotational_attitude()		attitude_rotation_type **rotation**
write_rotational_attitude_rate()		attitude_rotation_rate_type **rate**
write_velocity()		own_velocity_type **velocity**

7.3.2 Navigation_Sub

Type: IDLInterface
Package: Navigation_Services
The interface implemented by a navigation system to provide C2INav services.

Table 7.27 - Methods of IDLInterface Navigation_Sub

Method	Notes	Parameters
request_covariance()	Requests the covariance between a chosen set of quantities to be reported. The covariance is to be reported at the fastest reporting rate of the chosen quantities. Can also cancel requests to the navigation system to send the particular covariance information.	request_id_type **request_id** navigation_report_kind_sequence_type **report_kinds** boolean **cancel**
request_navigation_data()	Request the navigation system to send all its available navigational information at the navigation system's default rate for the data. Where the reported data can have different statistical representations (measurement kinds) all forms supported by the navigation subsystem are reported at the navigation system's default rate for that measurement and report kind. Can also cancel requests to the navigation system to send any of its available navigational information.	request_id_type **request_id** boolean **cancel**
request_navigation_data()	Request the navigation system to send a particular kind of navigation report with specified measurement types at a configurable interval. If the data is already being reported then this request updates the interval at which it is reported (or sets it to the navigation system's default rate); it does not act cumulatively on existing data reporting. Can also cancel requests to the navigation system to send any of its available navigational information.	request_id_type **request_id** navigation_request_type **request** boolean **cancel**

8 Domain Model Platform-Specific Models

8.1 DDS PSM

The DDS Data Model PSM defines a set of IDL files for the Data Model packages defined by the PIM. Topic types (i.e., IDL structs with keys) are defined for those IDL struct stereotyped classes that classify a single parameter on an interface method. This avoids redundant indirection. Comments are added to the IDL files to reflect the mapping rules below.

IDL types referred to by this PSM but defined by OARIS are to be found in the DDS PSM files for the OARIS specification.

The detailed rules for the MDA code generation from the Data Model PIM to the DDS PSM IDL are as follows:

- The PIM attributes are mapped to IDL attributes.
- Optional attributes are mapped to a union type with a single member present when the exists case attribute is true.
- Collections in the PIM are mapped to IDL sequences.
- Specialization / Generalization PIM relationships are mapped to in-lined base type attributes.

8.2 GraphQL PSM

The GraphQL (see https://graphql.org/) Data Model PSM defines a set of schema files for the Data Model packages defined by the PIM.

The detailed rules for the MDA code generation from the Data Model PIM to the DDS PSM IDL are as follows:

- The PIM attributes are mapped to GraphQL type attributes.
- Mandatory attributes are mapped to a mandatory GraphQL attributes with an exclamation mark – '!'.
- Optional attributes are mapped to an optional GraphQL attributes without an exclamation mark.
- Collections in the PIM are mapped to GraphQL sequences.
- Specialization / Generalization PIM relationships are mapped to in-lined base type attributes.

This page intentionally left blank.

9 Service Model Platform Specific Models

9.1 DDS PSM

The DDS Services PSM defines IDL files for each package defined in the Services PIM. For each method on each interface class an IDL struct for a DDS topic named for the method is generated; each parameter is mapped to an attribute of the IDL struct. This is unless there is only one attribute (of IDL struct stereotype) in which case the topic type is defined in the Domain Model (i.e. it corresponds to the parameter's class). Note that the PIM only defines in parameters, there are no return parameters defined and all methods have at least one parameter.

The DDS PSM maps the request_all_navigation_data and request_navigation_data method to the DDS discovery, publish and subscribe functionality. Consequently, these methods are not explicitly defined by the DDS IDL.

9.2 GraphQL PSM

The GraphQL Services PSM defines a set of schema files files for each package defined in the Services PIM. For each method on each interface class a GraphQL GraphQL Query type is generated; each parameter is mapped to an attribute of the Query type.

The GraphQL PSM maps the request_all_navigation_data and request_navigation_data methods to the GraphQL subscription types.