Standardized navigational data for situational awareness during simultaneous maritime operations

M. Hagaseth¹, S. P. Berge¹

¹SINTEF Ocean, Marinteknisk senter, Otto Nielsens vei 10, 7052 Trondheim, Norway
Marianne.Hagaseth@sintef.no

Abstract. The objective of this paper is to describe a protocol for exchange of navigational data between ships operating in close proximity. The protocol supports the exchange of data needed for relative positioning and includes geometry data of the objects involved in the operation in addition to inertial and positional data. The protocol takes into account the variable and limited communication capabilities that may be available for maritime operations. The paper describes one use case where two ships perform a simultaneous operation and another use case where a ship broadcasts its intended route. A S-100 Application Schema covering the required data is presented. Examples of the message exchange and bandwidth requirements are given. The resulting protocol will be useful for autonomous systems and ships that need standardized exchange of data to support situational awareness during their passage. The paper is written as part of the H2H project [1] which is developing an open system architecture to ensure interoperability between different vendors of autonomous systems when it comes to the data exchange related to position and geometry data.

1. Introduction

This paper describes the S-100 Application schema [2] for the H2H Framework, Figure 1. The H2H Framework is an open system architecture providing standardized information exchange between ships to help and assist both mariners and lately unmanned autonomous vessels to take the correct navigational decision when navigating in close proximity of other objects. The H2H Framework consists of two parts, the information model and the service description. The information model (domain model) covers the minimum set of data needed to compute the desired information for safe autonomous navigation including relative distances, velocities and the uncertainty zone. The service description includes services to establish the connection between two ships, to request data from a ship and to receive data that has been requested. The services are grouped as follows:

- Services to check the availability of the H2H system, its services and sensor data, and to establish connections to H2H Engines and Applications.
- Services to request data from a H2H Engine. This can be both sensor data but also calculated values as for instance uncertainty zones, relative distance and speed, in addition to object geometry.
- Services to receive data that has been requested.

In this paper, we will focus on the information model of the H2H Framework. The framework describes the information needed internally in the H2H Engine in addition to information that needs to be exchanged between two H2H engines on board different objects/ships (own and target) (see red arrows in Figure 1) and also the information exchanged between a H2H Application and the H2H Engine onboard the same object/ship (see red arrows in Figure 1). E2A means Engine to Application, while E2E stands for Engine to Engine. The operation of each H2H Engine, both on own object/ship and on target object/ship are transparent, meaning that the internal calculations done in the engines will not be part of the framework. Thus, the framework allows many H2H Applications and H2H Engines to work together in a total H2H System.
2. Simultaneous Operation (SimOps) Use Case

Figure 2 gives an outline of a simultaneous operation where the own ship is leading an operation involving another (following ship). This use case covers in principle any scenario where two or more ships perform a simultaneous operation in close proximity of each other, and where the goal is to operate as close as necessary while still maintaining safe distance.

The ship that is leading the operation starts the operation by broadcasting a message to possible participants. The own ship also defines a safety zone which shall not be entered by other vessels, unless approved by the leading ship. They also define an escape zone where the following ship should enter if the leading vessel need to abort the side-by-side-operation.

Figure 3 gives an overview of the processes, sub-processes and events that may be involved in a simultaneous operation. The main processes are Approach, Operation, and Finish. During the approach, the initial information is exchanged among the ships inside the Area of Interest (AOI), and end-to-end communications are set up. Also, alternative communication means including their priorities are agreed among the participants. A simple geometry model including the length and breadth of the ship is exchanged at the start of the approach. Later, before the operation starts, more detailed geometry models are exchanged. The level of detail of the models are decided during the exchange of the models. Also, all ships participating in the operation must exchange information about the available sensors to others, including the latency of the data and the position of the sensor relative to the ship's physical origin. Also, the requested update rates on the data are agreed among the ships.

During the operation, data from the sensors are exchanged among the participating ships, including position data, raw GNSS data, inertial data and range sensor data. Based on this data from its own sensors and also similar data received from the other participants, a ship can calculate both its own uncertainty zone (UZ) and the other ship's uncertainty zone (UZ). This uncertainty zone represents the uncertainty in the hull's position, specified with a certain probability, often 95%, Figure 4. This is thus a well-suited tool for estimation of the relative distance, speed and bearing between two ships which operates in close proximity, see the red arrows in Figure 4.

3. S-100 Application Schema for H2H

Figure 6 and Figure 7 show the S-100 Application Schema for the information that is needed to compute the position of each ship in a simultaneous operation with a certain probability. It also includes classes to describe the intended route as a list of waypoints and prediction zones that may be sent to others to be used during autonomous navigation. The rest of this section gives a brief description of the most important feature types and information types in the H2H S-100 Application Schema.
3.1. *ObjectPoint*

This is the class to represent a point related to the physical object/ship. The fromAft, fromCenter and fromKeel coordinates specifies the position relative to the ship's physical origin in three dimensions. The accuracy is represented by a number for each of the dimensions. *ObjectPoint* can represent the position of a real sensor or a virtual sensor. Also, a set of instances of *ObjectPoint* is used represent the object geometry (*ObjectGeometry*). The *ObjectPoint* class is used to define a vector between
points on two different objects, and further to do the calculations of relative distance, velocity and bearing between these two points. The ObjectPoint class is also used to describe an uncertainty zone or an operational zone.

Figure 4. Uncertainty Zone (in red) [Source: Kongsberg Seatex].

3.2. ObjectGeometry
This is the list of points that represents the detailed geometry of a ship's hull, or more generally, it describes the geometry of an object. The geometry has a certain accuracy. An ObjectGeometry consists of a list of ObjectPoints.

3.3. RequestedObjectGeometry
For each operation, a ship/object can request an object geometry from another ship/object with certain accuracy and size, dependent on the communication means available and dependent on the calculations that is to be done.

3.4. ShipDynamics
This is the position, speed, course and heading for a selected point on the ship, among other motion information related to a ship. Also, the intention of the voyage or leg can be given as a textual description. A more formal way of describing the intention, is to add a set of waypoints as an intended route.

3.5. Communication
This class contains information about the communication technologies that are available for this ship/object to be used in the calculations. Each technology is listed with their bandwidth, latency and the type of communication.

3.6. Ship
This class contains both the static and dynamic information about a ship, including the simple geometry model (length and breadth). It also contains a description of the position of the ship's physical origin, described as the distance from bow and aft, starboard and port side, and also in the z direction.

3.7. ObjectData
This class contains information that is common for both fixed and moveable objects, that is, for both ships, quays, bouys, locks etc.

3.8. Sensor
This class contains the information that is needed to describe a sensor onboard a ship, including the type, update rate, and whether it is available or not. It also contains the lever arm for the sensor, that is, the position of the sensor onboard the ship relative to the ship's physical origin.

- **Name**: The name of the sensor
- **IsAvailable**: This is a Boolean indicating whether the sensor is available for use by the calculations.
- **UpdateRate**: This is the update rate that the sensor will produce data at.
- **LeverArm**: This is the location of the sensor onboard the ship or on the fixed object, relative to the ship's/object's origin. This is shown as the "Sensor A modelled location" (a green circle) in Figure 5. The physical location onboard the ship or object is represented by FromAft, FromCenter, FromKeel and the installation error (AccuracyX, AccuracyY, AccuracyZ). This accuracy is the accuracy of the installation of the sensor onboard the ship or on the fixed object. This corresponds to the distance shown as "Installation error" in Figure 5.
- **SensorType**: This is the type of the sensor.

Each sensor produces a set of SensorData.

3.9. SensorData
The SensorData class and its specializations contain the actual measurements from the sensors. The SensorData contains the properties that are common for all different types of sensor data. This includes the key identifier of the sensor data and the Timestamp for the sensor data. The concepts linked to SensorData includes PositionData, GNSSRawData\(^1\), InertialSensorData, AngleSensorData and RangeSensorData. This means that all these concepts inherit the SensorDataId and Timestamp from the SensorData concept. Each instance of the SensorData concept represents data generated by a certain sensor. The Accuracy property indicates the accuracy of the data produced by the sensor, that is, the uncertainty shown as "Sensor Error" in Figure 5.

3.10. VirtualSensor
This class represents a set of calculated values in a position related to the ship's/object's physical origin. This means that this concept represents a virtual sensor, since no physical sensor is located at this position, but instead, a set of values for this position is calculated based on the sensor values otherwise available for the ship/object. The virtual sensor data is used to calculate relative distance, velocity and bearing between two points for two different ships/objects.

3.11. RelativeDistanceAndVelocity
The RelativeDistanceAndVelocity class contains information about relative distance and velocity from a given start point to a given end point. The start and end points normally refer to points on own and target geometry objects, respectively. It also contains the bearing from start to end reference point. This is shown in Figure 4 as the red arrows.

3.12. Uncertainty Zone
The UncertaintyZone class contains information about an uncertainty zone that has been calculated by a H2H Engine. Since both sensor installation accuracy, sensor data accuracy and object geometry accuracy are used in the calculations of the uncertainty zone, the actual uncertainty zone geometry is more complex than just the object geometry adjusted with a specific sigma value. Therefore, a

\(^1\) Data in RTCM format
separate property UncertaintyZone is needed to contain the actual geometry of the uncertainty zone. One object can have several uncertainty zones, both because several calculations based on different sensors and sensor data will result in different uncertainty zones, but also because one calculation of an uncertainty zone can be viewed as several slices of the uncertainty zone geometry. Examples of these are that several 2D slices of the 3D uncertainty zone can be defined, and thus result in several uncertainty zones. Examples are a 2D slice at the water level or at the level where the geometry has the largest areal or based on some other definition.

Figure 5. Various Sensor Uncertainties [Source: H2H Project].

3.13. Operation
This class contains the properties to describe an operation, which can be SimOps, Docking, Lock or Sailing. An operation can involve several objects, while each object can participate in several operations.

3.14. OperationalZone
This class contains properties to describe operational zones and is closely linked to the applications. Notice that an operational zone can be defined independent of sensor values and object geometries, for instance in the case of escape zones. However, in other cases, it can be related to sensor values and geometries, for instance when covering regions related to trajectories. Area of interest and safety zone are two types of operational zones needed in the simultaneous operation case.
Figure 6. S-100 Application Schema for H2I information.
3.15. **Heartbeat**

This class contains information saying that the H2H Engine is alive.

3.16. **H2HEngineMetaData**

This class contains properties needed to keep track of the status of a H2H Engine. Each object can have more than one H2H Engine.

4. **Message Sequencing and VDES**

Figure 8 shows a sequence diagram for the SimOps use case. Figure 9 shows the S-100 message exchange model representing the response geometry message. The content of each of message type is defined based on the S-100 Application schema presented in this paper. An example message is shown in Table 1 for the ResponseGeometryMessage. This message contains a list of points that defines the geometry of an object in 2D or 3D including the accuracy of each of the points. The level of detail of the geometry model is decided by the number of points in the point list. When it comes to the VDES coding of the geometry, 10 points in the object geometry will give a message size of 1135 bits which...
fits into one VDES packet having 3 slots. Larger geometries can be sent in a message spanning more VDES packets\(^2\).

![SimOps Message Sequences](image)

**Figure 8.** SimOps Message Sequences.

5. Intended Route
An important functionality for autonomous ships that is related to the SimOps case is the possibility to broadcast the intended route to others. This is covered in the H2H Framework by amending the model with classes from the S-421 Route exchange product specification and selecting those parameters needed.

\(^2\) VDES is a newly specified digital communication technology utilizing the VHF band to provide free ship-ship communication with more bandwidth (ca 300 kbps for each channel (transmit and receive) in full duplex mode).
Figure 9. S-100 Exchange Message for sending geometry data.

Table 1. Message example: ResponseGeometryMessage.

| Parameter            | Number of bit | Description                                                                 |
|----------------------|---------------|----------------------------------------------------------------------------|
| MessageId            | 6             | ResponseGeometryMessage. The message type is used to send a response on an object geometry request. |
| SenderMMSI           | 30            | The sender ship MMSI                                                        |
| ReceiverMMSI         | 30            | The receiver ship MMSI                                                      |
| ObjectId             | 128           | The object id (UUID) for the sender MMSI                                    |
| ObjectGeometryUUID   | 128           | UUID of the sender's object geometry                                         |
| payloadSize          | 16            | Size of the geometry in kBytes                                              |
| geometryType         | 1             | 2D or 3D points?                                                            |
| NumberOfPoints       | 16            | Number of points in the PointList                                           |
| timestamp            | 6             | Timestamp of the geometry data                                              |
| fromAft              | 16            | pointList: Length in meters, including 2 decimals, from the ship's origin or from another reference point in the X direction. |
| fromCenter           | 16            | pointList: Length in meters, including 2 decimals, from the ship's origin or from another reference point in the Y direction. |
| fromKeel             | 16            | pointList: Length in meters, including 2 decimals, from the ship's origin or from another reference point in the Z direction. |
| AccuracyFromAft      | 12            | pointList: Accuracy in X direction, up to 40.96 meter                        |
| AccuracyFromCenter   | 12            | pointList: Accuracy in Y direction, up to 40.96 meter                        |
| AccuracyFromKeel     | 12            | pointList: Accuracy in Z direction, up to 40.96 meter                        |
| TOTAL                | 295 + 84x NumberOfPoints bits | The geometry size is dependent of the number of points:  
  • A simple "shoe box" representation of the ship geometry with 4 points gives 697 bits => 88 bytes  
  • A simple representation with 5 points gives 715 bits  
  • A geometry with 100 points gives 8695 bits  
  • With a VDES addressed message with 3 slots having a total of 1272 bit payload, 11 points can be transmitted in one VDES |
| Parameter                  | Number of bit | Description                                                                                                                                                                                                                                                                                                                                 |
|----------------------------|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| message coded as a ResponseGeometry message. |               |                                                                                                                                                                                                                                                                                                                                                             |

![Figure 10. Example of waypoints and prediction zone for intended route.](image1.png)

![Figure 11. S-100 Information model for Intended Route.](image2.png)

The route can be specified as a set of waypoints (black dots) defined as Latitude and Longitude and possibly with the addition of the accuracy of the Latitude and Longitude values, left part of Figure 10. In addition, a prediction zone (operational zone) that represents the uncertainty of the position of the ship's geometry can be specified for each waypoint. This is an optional information for the route intension. In addition, the uncertainty of the leg between each waypoint can be specified, right part of Figure 10. This includes a description of the legs between the waypoints including the size of the expected deviation from the intended course both to starboard and port side.
The S-100 information model for the intended route is shown in Figure 11. By proposing VDES messages for the exchange of intended routes and analysing this with respect to the bandwidth requirement, we find that a route consisting of 10 waypoints will take about 9.7 kbit. This can be reduced to about 3.5 kbit if the prediction zones for each waypoint are omitted, and only the waypoint and legs with uncertainties are transferred.

6. Conclusions
The proposal for a S-100 Application Schema covering position data, sensor data, geometry information, uncertainty zones and operational zones are defined. Based on this information, services to calculate the relative distance, speed and bearing between two points on two ships in close proximity can be defined. The message exchange needed to support calculation of uncertainty zones, relative distances, speeds and bearings can be specified using the same coding as for AIS data according to Recommendation ITU-R M.1371-2 [3]. When using VDES as the carrier, the message exchange will be free for the ships, and the bandwidth will be sufficient for simple geometry models and sensor data. When communication means with higher bandwidth are available, more detailed geometry models and more sensor data can be exchanged according to this Application Schema. Calculations of the total bandwidth requirement for the message exchange has been done, for different situations:

Two ships participate in the SimOps, one own ship, and one following ship, each ship has 5 sensors: For a geometry with 5 points for each ship, ca 84 kbit of data must be exchanged. With each ship geometry consisting of 100 points, 103 kbit must be exchanged, while 1000 points gives about 280 kbit of data for the initial setup. With a maximum capacity of 307.2 kb/s for VDES, this shows that the initialization should be feasible also in cases where the communication conditions are significantly worse that in the optimal case.

Six ships participate in the SimOps, one own ship, and five following ships, each ship has 5 sensors: In this case, an object geometry with 5 points gives a data exchange of 288 kbits, 100 points gives 580 kbits, while 1000 points gives 3344 kbits. This is still possible given the typical bandwidth of VDES communications.

In addition of proposing a new S-100 standard, we have made use of the S-421 route exchange standard to handle intended routes for autonomous ships. We have also recognize that a common S-100 feature catalogue for the operational domain is needed: Although a feature catalogue with concepts exists, no common set of definitions that can be reused, exists. For instance, the properties describing various ship particulars as for instance ship ids (MMSI, call sign) or ship size (length, breadth) are found in several product specifications. It is not clear which of these definitions should be reused when specifying a new application schema. This means that a common reference model consisting of a minimum common set of attributes, features, information types and associations should be defined and included to the S-100 feature catalogue.

Acknowledgement
This work has been done as part of the H2H project which has received funding from the European GNSS Agency under the European Union's Horizon 2020 research and innovation programme grant agreement No 775998.

References
[1] https://www.sintef.no/projectweb/hull-to-hull/
[2] http://s100.iho.int/S100/home/s100-introduction
[3] ITU, Rec. ITU-R M.1371-2, Technical characteristics for a universal shipborne automatic identification system using time division multiple access in the VHF maritime mobile band

ANNEX 1