STMVal_D5.22

Report on Information Flows from Test Beds

20180315

Co-financed by the Connecting Europe Facility of the European Union
TEN-T PROJECT NO: 2014-EU-TM-0206-S

The sole responsibility of this publication lies with the author. The European Union is not responsible for any use that may be made of the information contained therein.
# Table of Contents

1. Executive Summary .................................................................................................................. 4
2. Background ................................................................................................................................ 5
   2.1 Information Environment Architecture .................................................................................. 5
3. Data Origins from Test Beds ...................................................................................................... 7
   3.1.1 Data Collection in Ports .................................................................................................... 10
   3.1.2 Data Collection in Ships .................................................................................................... 14
   3.1.3 Data Collection in Shore Centres ....................................................................................... 19
4. Data gathering and processing ..................................................................................................... 20
   4.1 Data from Activity 1: Port CDM .............................................................................................. 20
      4.1.1 Data gathering .................................................................................................................. 21
      4.1.2 Data storage ..................................................................................................................... 22
      4.1.3 Data processing ............................................................................................................... 23
   4.2 Data from Activity 2: VIS logs .............................................................................................. 25
      4.2.1 Data gathering .................................................................................................................. 25
      4.2.2 Data storage ..................................................................................................................... 28
      4.2.3 Data processing ............................................................................................................... 28
   4.3 Expected outputs ..................................................................................................................... 29
   4.4 Software decisions ................................................................................................................. 30
5. Data Integration .......................................................................................................................... 30
6. Reference Material ...................................................................................................................... 33
1 Executive Summary

In the aim of facilitating STM validation through the quantification of the benefits associated with STM implementation, using the data collected from the test beds designed within the project, methods for evaluation as well as requirements for data collection must be defined.

Thus, the objective of this document is to describe the requirements for data collection from test beds and the processes followed to be compiled in a smart repository that would enable data integration in order to add value to the analyses.

Accordingly, the data sources are based in the units where the test beds take place, particularly: 300 ships, 13 ports and 5 shore centers. This way, the requirements for data collection can be defined by the processes of loading and filtering data, as well as the methodologies used for the development of a smart repository of data, which should be used for feeding the analysis engine.

The steps defined for this purpose that will be described in this report, are the following:

1. Data Origins from Test Beds: Agreement on data gathering processes with test beds managers.
2. Data Gathering: Design and implementation of the Extract, Transform and Load (ETL) processes.
3. Data Processing: Design of the smart repository according to data model structure.
4. Data Integration: Development of statistical analyses from the amount of data compiled in the smart repository.
2 Background

A&E was responsible for creating a common framework for information sharing between the different concepts of STM. This common framework was named “Information Environment” and consists on the ensemble of the different methodologies used for the organization of the information expected to be collected during the course of the Test Beds designed, with the aim of analysing and extracting conclusive statements that could facilitate the demonstration of the benefits of implementing Sea Traffic Management globally.

In this report, the focus is on the last layer, which originates in the information flows from the different data sources in the Test Bed. The objective is to define all the flows regarding data collection, processing and integration for analyses design.

![Figure 1. Functional Layer of Data Analysis supported by VESSL DataBase](image)

2.1 Information Environment Architecture

The following structure illustrates and explains in each layer, from the bottom to the top, the steps followed in the construction of the Information Environment for STM Analysis & Evaluation.

The first layer, STM Concept reviews the basis of the need of building up such a system and defines, with the inspiration of the aviation sector, the different STM concepts: Strategic Voyage Management, Dynamic Voyage Management, Flow Management, Port Collaborative Decision Making and Sea System Wide Information Management.

The different STM services have been refined and detailed during the first phase of the project. In this first stage, the responsible of each concept designed the specifications of the services. This is described in the second layer of the Information Environment Architecture: STM Services.
In the first phase of the project, A&E was indirectly involved in the mentioned process in order to gain understanding of the services and to determine which the most relevant objectives are as well as to define the hypotheses that they can fulfil in connection with the maritime sector. These results complete the third layer of the Information Environment Structure: STM Validation Hypotheses Model.

According to the hypotheses deducted and considering the data that can be extracted from the test beds, the simulations, and the transversal issues such as legal, business, operational, safety and training aspects, the fourth layer of the information environment is defined as Information Environment Sources.

The fifth layer of the Information Environment Structure is titled Collection, Processing and Analysis of Data, which contains the methodologies for collecting the data from the different sources in order to make analyses with the support of different tools and to show the results that can fulfil STM validation.
3 Data Origins from Test Beds

As specified in the Information Environment scheme, STM concept will be tested using several information sources from different natures and origins such as the STM Test Beds:

- PortCDM implementation guidelines have been initiated in eight European ports and it is foreseen to be extended to five more until the end of the project. Some of the data produced by these ports will be provided for the analysis.

- STM has assigned four leading ECDIS suppliers to provide up to 300 STM-compatible on-board systems by the launch of a public procurement to further test the interoperability between various systems and providers. Those interactions will deliver quantitative and qualitative data for analysis.

- Shore services are being developed within some shore centres that will be able to interact with STM compliant ships navigating in their area of influence and that will produce qualitative and quantitative data for the analysis.

These three data sources from the test beds is what we call data entities and each entity generates different kind of files that will be loaded to the smart repository in order to have the complete picture of the situations given.

In this sense, the communications between the different entities included in the test beds will give rise to some files containing the information from the kind of communications that are taking place. On the other hand, AIS data provides the reality of the route performed in which the changes due to STM tools could be observed.

- Sharing of voyage plan is the backbone in STM and facilitates all other functions. Ships in the test bed will share voyage plans (VP) with shore side actors like Shore Centers (SC), ports and service providers. The ship/shipping company is the information owner of the VP and as such chooses which actors that should be granted access to the voyage plan. This is part of the access management functionality in SeaSWIM.

- The port call synchronization is the service connecting the ships Voyage Plan with the port of call/port of departure. The ETA in the voyage plan will be the starting point of the Port Collaborative Decision Making (PortCDM). When all port actors have agreed on a time for the port call, they will send back a RTA (recommended time of arrival) and ETA might be adjusted accordingly to gain the benefits of STM Port call synchronization.
The ships in the test bed will automatically always share their VPs with at least “STM validation centre”.

Apart from that, there are more services that are being developed by service providers:

- Route optimisation by SSPA
- ETA Calculator by SMHI
- Route optimisation by CIMNE
- Enhanced Monitoring by Navicon
- NPRS by SMA
- Baltic Navigational Warnings service by SMA
- Kongsberg in Tarifa shore center
- GBG, Horten, Kvitsøy Shore center
  - Transas
  - SAAB
  - Airbus
  - Kongsberg

**Table 1. List of STM Validation Shore Centres**

<table>
<thead>
<tr>
<th>Name of Shore Center</th>
<th>Country</th>
<th>Manufacturer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tallin VTS</td>
<td>Estonia</td>
<td>Transas</td>
</tr>
<tr>
<td>Kvitsøy VTS</td>
<td>Norway</td>
<td>Kongsberg</td>
</tr>
<tr>
<td>Horten VTS</td>
<td>Norway</td>
<td>Kongsberg</td>
</tr>
<tr>
<td>Gothenburg SC</td>
<td>Sweden</td>
<td>Testing all systems</td>
</tr>
<tr>
<td>Great Belt VTS</td>
<td>Denmark</td>
<td>Airbus</td>
</tr>
<tr>
<td>Tarifa VTS</td>
<td>Spain</td>
<td>Kongsberg</td>
</tr>
</tbody>
</table>
Figure 4. STM Servers and Service Instances
3.1.1 Data Collection in Ports

The overall scheme about information flows in a port using Port CDM architecture is depicted in Figure 5. The actors involved in a port call transmit the information related to their operations to the Port CDM platform. The `port call finder´ gathers and validates the data, generating a text line in which the information of the port call is collected and stored. The `message broker´ deals with the information sharing with the corresponding actors according to previous authorization, validation and authentication requirements.

![Figure 5: Information flows in Port CDM and specific zoom on Analytics for A&E activity. Source: Own elaboration](image)

Duration times:

A focus on duration times forms the basis for making calculations about service times and waiting times for different actors during the port call. Paying attention towards the duration time of the port visit could help satisfying a shipping company needs on the following factors: fast turnaround, just-in-time operations and minimal waiting times.

The main purpose of the analytical service “duration time” provided by Port CDM is to enable the calculation of the time spent on the port visit divided into the different Port CDM process time types. Thus, the duration time (Total Turnaround Time) is the total time spent on the port visit comprising the following port call process depicted in the following figure:
- Movement Time
- Nautical Service Time
- Stationary Service Time
- Anchoring Time
- Berth Visit Duration Time
- Additional Service Time
- Waiting Time

**Figure 6:** Port call process time to calculate stages durations. Source: Own elaboration
Based on the PCMF, the identification of the services mentioned in the port call process duration are the following:

**Table 2: Identification of possible services in a port call. Source: STM project**

<table>
<thead>
<tr>
<th>Services Identification</th>
<th>Pilotage</th>
<th>Escort Towage</th>
<th>Ice Breaking</th>
<th>Towage</th>
<th>Berth shifting</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Stationary Services:</strong></td>
<td>Bunkering Operation</td>
<td>Gangway</td>
<td>Security</td>
<td>Arrival mooring operation</td>
<td>Departure moorers</td>
</tr>
<tr>
<td></td>
<td>Cargo Operation</td>
<td>Garbage Operation</td>
<td>Lubeoil Operation</td>
<td>Pontoons &amp; fenders</td>
<td>Provision Operation</td>
</tr>
<tr>
<td></td>
<td>Arrival moorers</td>
<td>Slop Operation</td>
<td>Tours</td>
<td>Water Operation</td>
<td>Pre-cargo Survey</td>
</tr>
<tr>
<td></td>
<td>Skilled worker</td>
<td>Sludge Operation</td>
<td>Forklift</td>
<td>Embarking</td>
<td>Post-cargo Survey</td>
</tr>
</tbody>
</table>

A focus on the duration time is aimed both to inform actors about the amount of time spent on different operations for port calls in the different stages of the turnaround process as a basis to optimise port calls following three dimensions:

- A particular port call (estimates and actuals) divided into the different processes times.
- For different types of port calls sorted by vessel type divided into the different processes times.
- For a desired aggregation, such as for a selected time span, the latest X port calls, (presented as sums, trends and/or correlations) divided into the different processes times.

In order to determine duration times, several different time types are being used as stated in Table 3. According to these calculations described by Port CDM, the validation is enabled via Port CDM services and the effects are studied based the data extracted from them and the aggregation of it.
<table>
<thead>
<tr>
<th>Port CDM process times</th>
<th>Methods of calculation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Movement time</td>
<td>The time between the instances of Arrival_Vessel_TrafficArea and Arrival_Vessel_Berth plus (if applicable)</td>
</tr>
<tr>
<td></td>
<td>The time between the instances of Arrival_Vessel_TrafficArea and Arrival_Vessel_AnchorageArea plus (if applicable)</td>
</tr>
<tr>
<td></td>
<td>The time between the instances of Departure_Vessel_AnchorageArea and Arrival_Vessel_Berth plus (if applicable)</td>
</tr>
<tr>
<td></td>
<td>The time between the instances of Departure_Vessel_AnchorageArea and Arrival_Vessel_AnchorageArea plus (if applicable)</td>
</tr>
<tr>
<td></td>
<td>The time between the instances of Departure_Vessel_Berth and Arrival_Vessel_Berth plus (if applicable)</td>
</tr>
<tr>
<td></td>
<td>The time between the instances of Departure_Vessel_Berth and Arrival_Vessel_AnchorageArea plus (if applicable)</td>
</tr>
<tr>
<td></td>
<td>The time between the instances of Departure_Vessel_Berth and Departure_Vessel_TrafficArea plus (if applicable)</td>
</tr>
<tr>
<td></td>
<td>The time between the instances of Departure_Vessel_AnchorageArea and Departure_Vessel_TrafficArea plus (if applicable)</td>
</tr>
<tr>
<td>Duration time</td>
<td>The time between the instances of Arrival_Vessel_AnchorageArea and Departure_Vessel_AnchorageArea</td>
</tr>
<tr>
<td>(estimated, calculated, actual)</td>
<td>The time between the instances of Arrival_Vessel_Berth and Departure_Vessel_Berth</td>
</tr>
<tr>
<td>Nautical service time</td>
<td>The first time of (NauticalServiceObjectX) commenced and the last time of (NauticalServiceObjectY) completed (where X and Y could be the same or different NauticalServiceObject)</td>
</tr>
<tr>
<td>Stationary service time</td>
<td>The first time of (StationaryServiceObjectX) commenced and the last time of (StationaryServiceObjectY) completed (where X and Y could be the same or different StationaryServiceObject)</td>
</tr>
<tr>
<td>Additional service time</td>
<td>The first time of [AdditionalServiceObjectX] commenced and the last time of [AdditionalServiceObjectY] completed (where X and Y could be the same or different AdditionalServiceObject)</td>
</tr>
<tr>
<td>Waiting time</td>
<td>The time by taking the instances of berth visit(s) and anchorage(s) of the time between arrival_vessel_anchorage area/arrival_vessel_berth and the first time of [StationaryServiceObject] commenced</td>
</tr>
<tr>
<td></td>
<td>the time by taking the instances of berth visit(s) and anchorage(s) of the time between [StationaryServiceObject]_completed and another [StationaryServiceObject]_commenced when at berth/anchorage</td>
</tr>
<tr>
<td></td>
<td>the time by taking the instances of berth visit(s) and anchorage(s) of if the time between [StationaryServiceObject]_completed and Departure_Vessel_AnchorageArea/Departure_Vessel_Berth (when no more service is to occur within berth visit or anchorage)</td>
</tr>
</tbody>
</table>

Table 3: Port CDM Process time Methods of calculation. Source: Own Elaboration
3.1.2 Data Collection in Ships

From the point of view of navigation, after the procurements launched based on the STM functionalities required specification, the providers are developing the main Voyage Management services that will be implemented for validation in the test beds.

STM compliant ships will exchange information using RTZ message format and AIS systems with STM ports and shore centres. The first route exchange format (RTZ 1.0) was published in Annex (S&T) of the ECDIS test standard IEC61174:2015 in August 2015. The RTZ format is now updated to version 1.1. In the STM Validation Project, the participants will use RTZ 1.1 with an STM extension.

Since the STM Project defined a need of a number of new attributes that was not covered in the Annex S, it was agreed with IEC and CIRM that those attributes could be included as extensions to the standard (see: Guidelines for the use of RTZ format).

In the case of ship-to-shore communication, the route exchange format is too large to be transmitted over AIS application specific message (ASM). However, the current development of VDES (VHF Data Exchange System) will provide an excellent communication channel for STM services to complement internet connection.

Four manufacturers have been selected by the leading organisation that previously set the functionalities requirements to install the STM module in the recruited ships among the fleet disposed by their clients: the ship-owners.

Not only the number of ships provided by each manufacturer is shown in this figure, but also the specific type of ships recruited by each one and the timeline scenario of STM module implementation for each procurement during the rest of the project. This information is essential to show the quantity of data that is going to be delivered in the different phases of the test bed and to remark that each manufacturer will provide different data and possibly in different formats.

At the end, there will be available data from 300 ships with STM module installed. These ships are being characterised by A&E and a summary sheet of each one will be produced. These sheets will contain relevant information about each ship such as the size, the engine power, the fuel consumption, the year of build, the flag…etc. The confidentiality of this information is pending to be confirmed.

The data foreseen to be collected from the ships in the test beds will be provided by three means:

- The log in VIS (Voyage Information Service).
- The log in SPIS (Ship-Port Information Service).
- AIS data.
### 3.1.2.1 VIS: VOYAGE INFORMATION SERVICE

Voyage Management services communications have been built based on Voyage Information Service (VIS) requirements. The main purpose with VIS is to handle the communication around voyage information and the main artefact Voyage Plan (VP) in RTZ format. VIS implements methods to expose new and updated VP's and to consume external VP's. VIS also supports subscription of voyage plans. In addition to voyage plans (RTZ), VIS also supports the exchange of the STM Text Message and area message (S124). In the figure below, a sequence example of the information flows between the actors in the exchange of voyage plans is described:

This service design is realised using REST (Representational State Transfer). REST is an architectural style, and an approach to communications that are often used in the development of Web services. The use of REST in VIS is preferred over the more heavyweight SOAP (Simple Object Access Protocol) style because REST does not leverage as much bandwidth, which makes it a better fit for use in communication between vessels and shore based representation of the same.
Figure 8: VIS Project Design. Source: STM Project

In this VIS project design figure, a double sense scheme of VIS is presented, both on public and private sides. On the left hand, the VIS between the ship and the manufacturer is located in a private REST, which means that the data is under the manufacturer's control and it may not be provided to the project for A&E.

If a shipping company agrees to transmit the voyage plans of their ships included in the STM test-beds through the SeaSWIM architecture, then these data will be directly available for analysis and Evaluation. The information flow processes have not been agreed today with the ship-owners included by the procurers.

VIS logs raw data exchanged between two services will need to be aggregated and messaged to meet KPIs. KPIs needs to be decomposed to meet measurements, post-preparation of logs and callService requests needs to be further extracted to get the RTZ exchanged.

3.1.2.2 SPIS: SHIP-PORT INFORMATION SERVICE

The main purpose of Ship-Port Information Service (SPIS) is to support the ship system with communication with Port CDM services in port. The service may be integrated into the on-board system or at shore, depending on the situation. The main functionality of the service is to send updates on times at locations (e.g. PTA/TTA, ETA, and ATA) and receive recommendations. The purpose is to support a collaborative decision-making process with the port. The diagram below shows the primary use case for the STM Validation project’s voyage management validation.
The Ship-Port Information Service has one private service side towards the inside application (such as STM Module), and one exposed service side on SeaSWIM. The Ship-Port Information Service is dependent on a SeaSWIM Connector that supports with SeaSWIM specific functionality for authentication and supports with service and identity lookup in Service and Identity Registry. The Ship-Port Information Service consumes the Port CDM services for exchanging port call messages in the Port Call Message format.

### 3.1.2.3 AUTOMATIC IDENTIFICATION SYSTEM

The Automatic Identification System (AIS) is a shipboard broadcast system that acts like a transponder, operating in the VHF maritime band that is capable of handling over 4,500 reports per minute and updates every two seconds. It uses Self-Organizing Time Division Multiple Access (SOTDMA) technology to meet this high broadcast rate and ensure reliable ship-to-ship operation. This system is designed to be capable of providing information about the ship to other ships and to coastal authorities automatically. In 2000, IMO adopted a new requirement for all ships to carry AIS system under Regulation 19 of SOLAS, Chapter V (Carriage requirements for shipborne navigational systems and equipment). The regulation requires AIS to be fitted aboard all ships of 300 GT, upwards engaged on international voyages, cargo ships of 500 GT, and upwards not engaged on international voyages and all passenger ships irrespective of size. The regulation requires that AIS shall:

- Provide information about IMO No, name, type, position, course, speed, navigational status and hazard cargo information among others.
- Receive automatically such information from similarly fitted ships; monitor and track ships.
- Exchange data with shore-based facilities.

---

The information contained in each AIS-data packet can be divided into the following two main categories:

- Dynamic information automatically transmitted every short period, depending on the class of transponders installed on board.
  - MMSI (Maritime Mobile service identity number) is a nine-digit number used by maritime digital selective calling (DSC), AIS and certain other equipment uniquely used to identify a ship or a coast radio station. They are regulated and managed internationally by the International Telecommunications Union in Geneva, Switzerland, just as radio call signs are regulated. The format and use is documented in Article 19 of the ITU Radio Regulations and ITU-R Recommendation M.585-6, available from the ITU.
  - AIS navigational status such as “under way using the engine”, at anchor, not under command, restricted maneuverability, constrained by her draught, moored, aground, engaged in fishing, under way sailing, etc.
  - Rate of Turn, right or left.
  - Speed over ground (in knots).
  - Position coordinates (latitude/longitude) provided by the integrated GPS.
  - Course over Ground.
  - Heading.
  - Bearing at own position.
  - UTC time.

- Static & Voyage related information. This information is provided by the vessels’ crew; updating correct information regarding each voyage. IMO promulgate Resolution A.917 (22) Guidelines for the on board operational use of shipborne automatic identification systems.
  - IMO number associated to vessels’ names.
  - Call Sign.
  - Type of cargo.
  - Dimensions (length/beam/draught).
  - Type of positioning system.
  - Destination.
  - ETA (Estimated time of arrival).

Several different AIS data sources are expected during the validation process:

- Historical, accurate and real time data from at least STM ships selected for case studies along the project period, with a particular focus on those periods when the STM services will be installed on ship’s bridges. These periods will be properly analysed from the point of view of the implementation of the STM module in ship’s bridges.
- Yearly STM port calls data from AIS sources (port level and shore centre level) to build times schemes along the STM vessels calls. This information provides relevant information about anchoring times and waiting times during a port call. Port services included during a port call can be monitored and this data can be analysed in order to find out the resource utilisation management due to STM services implemented.

3.1.2.4 GLOBAL POSITIONING SYSTEM

GPS information about the tracking of the vessels during their voyages could be included in the analysis to find out the improvements of the use of STM services in navigation. The same concept will be taken into account when simulations data is stored. Moreover, specific information collected from Shore Centres will lead to better understanding of the advantages by using STM services during navigation. Services such as “enhanced monitoring” and “route Cross-check” might be evaluated using this compiled data. Voyage plans from sea carriers that provide vessels to the project will be collected and deeply analysed in order to see those changes in criteria during navigation evaluating the possibility of requiring sea carriers information about VDR (Voyage Data Recorder). The main aim with this data collection is to appraise and compare both non-STM services and STM services applied scenarios.

3.1.3 Data Collection in Shore Centres

VIS and SPIS will be used as well for the quantitative analysis in the shore centres. Some qualitative data could be collected by making questionnaires to the users.
4 Data gathering and processing

This section describes the structure and the means by which the information from activities 1 and 2 of the present project will be gathered as well as how it will be initially stored.

4.1 Data from Activity 1: Port CDM

Activity 1, through PortCDM, stores data related to the different port calls of the vessels arriving at the ports involved in the project. In particular, Activity 1 registers different timestamps for different events or states occurring at the port for each port call. This information is collected for both STM and non-STM ships.

Despite the large amount of information collected in Activity 1, it has been estimated that with only a limited number of these states from each port call will be necessary to calculate the minimum set of indicators that will allow Activity 5 to evaluate the impact of STM on the maritime industry. These states capture the occurrence of different events happening during the call of a ship at a port, as depicted in Figure 10.

An initial set of states of interest could be the following:

- ETA: Estimated Time of Arrival of the vessel to the traffic area.
- ATA: Actual Time of Arrival of the vessel to the traffic area.
- Actual time of arrival at berth.
- Estimated time of Cargo Operations Commenced.
- Actual time of Cargo Operations Commenced.
- Estimated time of Cargo Operations Completed.
- Actual time of Cargo Operations Completed.
- Actual time of Vessel Departure from berth.

Figure 10: Timestamps/States reported by Activity 1, PortCDM. The associated events occur at different stages of the stay of a ship in a port.
• Actual time of Vessel Departure from port area.
• ETD: Estimated time of Departure of the vessel from the traffic area.
• ATD: Actual Time of Departure of the vessel from the traffic area.

Additionally, these states include information like the:
• STM port call ID.
• Local port call ID.
• Port ID (in the form of an unlocode).
• Vessel Id (IMO).
• Location associated to the state, if any (at, from or to location).
• Reporting time.
• Agent reporting the state.
• Type of time, e.g., actual or estimated.

4.1.1 Data gathering

The backend developed for PortCDM in Activity 1 already allows gathering the data collected in each port through a subscription process to potential data consumers. This process is pull-based and returns the data collected between calls to the service. Hence, a simple daily procedure polling each port can be set in order to retrieve this information in a daily basis.

The information from the PortCDM backend comes in XML format. Find below an example from the port of Valencia of how this service returns the data. The local port call, message, imo and times have been anonymized.

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<collection xmlns:ns2="urn:mrn:stm:schema:port-call-message:0.6.1:payload"
            xmlns:ns3="urn:mrn:stm:schema:port-call-message:0.6.1"
            xmlns:ns4="urn:mrn:stm:schema:port-call-message:0.6.1:entity">
  <ns3:portCallMessage>
    <ns3:portCallId>urn:mrn:stm:portcdm:port_call:LocalPortCallId</ns3:portCallId>
    <ns3:vesselId>urn:mrn:stm:vessel:IMO:X</ns3:vesselId>
    <ns3:messageId>urn:mrn:stm:portcdm:message:Y</ns3:messageId>
    <ns3:reportedAt>Z</ns3:reportedAt>
    <ns3:reportedBy>urn:mrn:stm:user:legacy:AIS</ns3:reportedBy>
    <ns3:payload xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns2:LocationState">
      <ns2:referenceObject>VESSEL</ns2:referenceObject>
      <ns2:time>ZZ</ns2:time>
      <ns2:timeType>ACTUAL</ns2:timeType>
      <ns2:arrivalLocation>
        <ns2:to>
          <ns4:position>
            <ns4:latitude>39.45458</ns4:latitude>
            <ns4:longitude>-0.3205503</ns4:longitude>
          </ns4:position>
        </ns2:to>
      </ns2:arrivalLocation>
    </ns3:payload>
  </ns3:portCallMessage>
</collection>
```
As can be seen, data comes as a list where each element is a different PortCDM message. It is possible to provide a filter in the input when subscribing to the service. By using this filter, it is possible to limit the information received. In this case, the information received is limited to the states of interest expressed above. One example of such a filter would be the following:

```json
[{
  "type": "stateDefinition", "element": "Departure_Vessel_TrafficArea"},
  {
  "type": "stateDefinition", "element": "Arrival_Vessel_TrafficArea"},
  {
  "type": "stateDefinition", "element": "Departure_Vessel_PortArea"},
  {
  "type": "stateDefinition", "element": "Arrival_Vessel_PortArea"},
  {
  "type": "stateDefinition", "element": "Departure_Vessel_Berth"},
  {
  "type": "stateDefinition", "element": "Arrival_Vessel_Berth"},
  {
  "type": "stateDefinition", "element": "CargoOp_Comenced"},
  {
  "type": "stateDefinition", "element": "CargoOp_Completed"}
]
```

The data gathering process will, then, consist in a script scheduled to be run once a day that will poll the backends from the different ports to collect the latter messages.

### 4.1.2 Data storage

The information collected when polling the port backends will be dumped into a SQL-like relational database. In particular, there will be a table per port and each port call message will be a register in it. Each register will store the following fields from the port call message: messageId, portCallId, reportedAt, reportedBy, stateDefinition, time, timeType, atLocation, fromLocation, toLocation and vesselId. These fields contain the following information:

- **messageId**: unique identifier for a message.
- **portCallId**: unique PortCDM based port call identifier. Note that localportcall id, which is the port call identifier for the port, may not be unique across multiple ports.
- **reportedAt**: time at which a timestamp/event was reported.
- **reportedBy**: who reported the event.
- **stateDefinition**: the event being reported.
- **Time**: time at which the event took place.
- **timeType**: describes whether the “time” field refers to an estimated, actual, planned, time.
atLocation, fromLocation, toLocation: locations at, from where or to where an event is occurring.

vesselId: unique identifier for the vessel. Currently, the IMO number is used.

An alternative is to keep all the messages in the same table adding a code for each port, in order to differentiate the messages from the different sources. In this case, the “port” code will be its unlocode.

The fields (not in the example above) isWithdrawn and withdrawnStatement may be included in the future in the standard messages. Upon their inclusion, they would be used to delete data from previously stored messages. Finally, note that the fields atLocation, fromLocation, toLocation may not be present empty/NULL and will result in a NULL value.

4.1.3 Data processing

Once the collection and storage script finishes downloading and storing the data from the different ports, another script triggers to process it. This second script processes the data received and enriches it.

The goal of the data processing is to compute the time it took for a service or event to complete, in the case of ACTUAL timestamps, or, in the case of ESTIMATED, the time that was originally estimated for it. The enrichment of the data will consist in collecting and storing data from the vessels calling at the different ports for further analysis and classification.

Summarizing, the script will perform the following tasks:

- Group the port call messages per portCallId, stateDefinition, timeType to remove potential duplicates of the same state. Duplicates are the result of multiple agents reporting the same state multiple times due to corrections or because there were multiple working at the same time on the vessel, e.g., multiple tugboats reporting the entrance of the vessel to port waters, or moorers its arrival at berth. Alternatively, there can be different agents reporting the same state, e.g., the AIS or the pilots reporting the entrance to traffic waters. Note that there may be multiple criteria to perform this removal of duplicates due to the possible causes, e.g., taking the earliest message in “commenced”/“arrival” states and the latest in “completed”/“departure” states, take the one with the latest reportedBy, etc.

- Compute the actual or estimated service time. This time is computed by subtracting the “commenced”/“arrival” state reported timestamp from the “completed”/“departure” one.

- Store the output in a new table state_duration including the VesselId, portCallId, stateName, timeType and serviceTime. By stateName we refer to the descriptive part of a stateDefinition. For instance, in locationStates like Arrival_Vessel_PortArea or Departure_Vessel_PortArea states, the stateName would be Vessel_PortArea. With serviceStates the stateName would be the result of removing the commenced or completed suffixes. With administrative states, we would remove the requested, request_received, denied... suffixes.
The adequate location has to be taken into account as well, as there may be multiple events of the same type in a port call due to berth shifting, e.g., two load/unload operation periods, or simply because the event happens twice, e.g., pilotage when arriving at the port and when leaving. Depending on the state the ‘at’, ‘from’ or ‘to’ location will be used.

- Similarly, store in a Portcall info table the basic information of the port call, like the portcallid, localportcallid, vessel and port.
- Check if the vesselId is in a table vessel_data. The table vessel_data includes data from the vessel including its vessel type, dead-weight tonnage (DWT), gross tonnage (GT), or maximum number of transported TEUs, cars, passengers or its lane meters, for instance. If the vessel is already in the table, no action is performed. Otherwise, the data associated for vessel is extracted from sources like the Act. One vessel registry or Act. Five VSSL databases.
- Every message used to compute a service time and its duplicates are copied to a complete_states table.
- Bear in mind that some states may take multiple days to complete. Those port call messages, which are not completed, i.e., arrivals without departures or commenced without completed, are left in the initial table, so they are used in the following day with the next batch of incoming data. These messages are copied as well to a secondary table incomplete_states. If a message from a different day than the present one is used to complete a state, it is looked up in the incomplete_states table and deleted.
- Those portCall messages associated to states that are not completed are deleted if they remain in the original table for more than one month, remaining only in the incomplete_states table.

This is an example of a process-taking place already. Other processes that are being proposed are the computation of certain KPIs for ports that would follow a process similar to the one described above. Figure 11 shows an overview of the whole gathering and storage process for data coming from activity 1.

Figure 11: Overview of the process of gathering and storing data from activity 1
4.2 Data from Activity 2: VIS logs

The information from activity 2 comes from both STM enabled vessels and shore centres. This information corresponds to the message exchange between both types of entity. These messages are registered in both sides by the VIS logging service. Hence, instead of analysing the information coming from vessels or shore centres, this subsection analyses the information obtained from VIS logs.

The different types of messages that can be found in VIS logs are:

- STM enabled vessels and shore centres: s124, RTZ and TXT messages.
- STM enabled vessels and PortCDM ports: SPIS messages.
- Among STM enabled vessels: RTZ and TXT messages.

In principle, Activity 5 will consume information related to the RTZ messages. The interest will focus on the number of times the route changes while the vessel has already sailed, the waypoints for each route or the variations on the estimated time of arrival to port. For other types of message, there will be an accounting process, to estimate the amount of messages of each type exchanged over time and per voyage.

4.2.1 Data gathering

Vessels and shore centres generate log files for different periods, usually one day, which are sent and stored by Activity 2, and made available via FTP. The FTP file structure has multiple layers. These layers depend on the manufacturer of the STM device installed in the vessel, the vessel itself, and the type of message. In particular, the root folder contains different folders, one per manufacturer. Each of these folders contains the VIS files containing the events related to a vessel during a period, usually a day, and a series of folders per type of message, i.e., RTZ, S124, TXT and OTHER. The VIS files are not, necessarily, self-contained. Some of the events appearing in the VIS files are linked to individual files in those folders. Figure 12, Figure 13 and Figure 14 show examples of this structure.

![Figure 12: Root directory for VIS logs FTP. Logs are classified per manufacturer.](image-url)
<table>
<thead>
<tr>
<th>Filename</th>
<th>Date</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADVETO003_3_21_2018_12_00_00_AM-3_21_2018_11_59_59_PM</td>
<td>22/03/2018</td>
<td>2:00</td>
</tr>
<tr>
<td>ADVETO003_3_22_2018_12_00_00_AM-3_22_2018_11_59_59_PM</td>
<td>23/03/2018</td>
<td>2:00</td>
</tr>
<tr>
<td>ADVETO003_3_23_2018_12_00_00_AM-3_23_2018_11_59_59_PM</td>
<td>24/03/2018</td>
<td>2:00</td>
</tr>
<tr>
<td>ADVETO003_3_24_2018_12_00_00_AM-3_24_2018_11_59_59_PM</td>
<td>25/03/2018</td>
<td>1:00</td>
</tr>
<tr>
<td>ADVETO003_3_25_2018_12_00_00_AM-3_25_2018_11_59_59_PM</td>
<td>26/03/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_3_26_2018_12_00_00_AM-3_26_2018_11_59_59_PM</td>
<td>27/03/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_3_27_2018_12_00_00_AM-3_27_2018_11_59_59_PM</td>
<td>28/03/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_3_28_2018_12_00_00_AM-3_28_2018_11_59_59_PM</td>
<td>29/03/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_3_29_2018_12_00_00_AM-3_29_2018_11_59_59_PM</td>
<td>30/03/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_3_30_2018_12_00_00_AM-3_30_2018_11_59_59_PM</td>
<td>31/03/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_3_31_2018_12_00_00_AM-3_31_2018_11_59_59_PM</td>
<td>01/04/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_4_1_2018_12_00_00_AM-4_1_2018_11_59_59_PM</td>
<td>02/04/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_4_2_2018_12_00_00_AM-4_2_2018_11_59_59_PM</td>
<td>03/04/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_4_3_2018_12_00_00_AM-4_3_2018_11_59_59_PM</td>
<td>04/04/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_4_4_2018_12_00_00_AM-4_4_2018_11_59_59_PM</td>
<td>05/04/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_4_5_2018_12_00_00_AM-4_5_2018_11_59_59_PM</td>
<td>06/04/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_4_6_2018_12_00_00_AM-4_6_2018_11_59_59_PM</td>
<td>07/04/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_4_7_2018_12_00_00_AM-4_7_2018_11_59_59_PM</td>
<td>08/04/2018</td>
<td>3:00</td>
</tr>
<tr>
<td>ADVETO003_4_8_2018_12_00_00_AM-4_8_2018_11_59_59_PM</td>
<td>09/04/2018</td>
<td>3:00</td>
</tr>
</tbody>
</table>

**Figure 13:** Manufacturer level for Vis logs FTP. There is a VIS file per period of time and vessel. Vessels, in this example, are named as ADVETO003, ADVETO004, etc. The Other/RTZ/s124/TXT folders store the messages mentioned in the VIS files.

As mentioned, the VIS files are not necessarily self-contained. This is an excerpt of the content of a VIS file.

**Figure 14:** Last level in the Act. Two FTP. Example of the content of the RTZ folder.
<?xml version="1.0" encoding="utf-8"?>
<Log xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xmlns:xsd="http://www.w3.org/2001/XMLSchema"
     xmlns="StmLog">
  <Event>
    <ServiceInstanceId>urn:mrn:stm:service:instance:adveto:imo8917601</ServiceInstanceId>
    <ExternalOrgId>urn:mrn:stm:org:sma</ExternalOrgId>
    <ExternalEntityId>urn:mrn:stm:service:instance:sma:stmvalidation</ExternalEntityId>
    <EventNumber>1</EventNumber>
    <EventType>7</EventType>
    <EventDataType>8</EventDataType>
    <EventParameters>uvid = , routeStatus = "</EventParameters>
  </Event>
  <Event>
    <ServiceInstanceId>urn:mrn:stm:service:instance:adveto:imo8917601</ServiceInstanceId>
    <ExternalOrgId>urn:mrn:stm:org:sma</ExternalOrgId>
    <ExternalEntityId>urn:mrn:stm:service:instance:sma:stmvalidation</ExternalEntityId>
    <EventNumber>1</EventNumber>
    <EventType>6</EventType>
    <EventDataType>8</EventDataType>
    <EventParameters>uvid = , routeStatus = "</EventParameters>
    <EventDataExternalFileRef>Other\2018-01-04_12_15_28.txt</EventDataExternalFileRef>
  </Event>
  <Event>
    <ServiceInstanceId>urn:mrn:stm:service:instance:adveto:imo8917601</ServiceInstanceId>
    <ExternalOrgId>urn:mrn:stm:org:adveto</ExternalOrgId>
    <ExternalEntityId>urn:mrn:stm:service:instance:adveto:imo9129263</ExternalEntityId>
    <EventNumber>3</EventNumber>
    <EventType>7</EventType>
    <EventDataType>1</EventDataType>
    <EventParameters>deliveryAckEndPoint = , callbackEndpoint = "</EventParameters>
    <EventDataExternalFileRef>RTZ\2018-01-31_23_32_28_urn_mrn_stm_voyage_id_adveto_Charterfrakt20904-Rostock-Rauma-1801302103.rtz</EventDataExternalFileRef>
  </Event>
  <Event ... />
</Log>

This excerpt shows three different events. If the event has an associated message the field EventDataExternalFileRef contains the path. The first message is self-contained, notice that the field EventDataExternalFileRef is absent. The second one refers to a TXT file stored in "Other\2018-01-04_12_15_28.txt". Finally, the third one refers to an RTZ event stored in ="RTZ\2018-01-31_23_32_28_urn_mrn_stm_voyage_id_adveto_Charterfrakt20904-Rostock-Rauma-1801302103.rtz". It is important to remark that the different types of file may come in different formats. In general, the files have the following format:
- VIS: XML format.
- RTZ: XML format.
- S124: XML format.
- TXT: JSON format.
- OTHER: JSON format.

Activity 5 will gather these files via a script that triggers at regular time intervals, in principle on a daily basis, and downloads in a local repository those files that have not been yet downloaded.

4.2.2 Data storage

A second script will load those newly downloaded files and perform the following operations:

- Load new VIS files and convert them to JSON format.
- Find in the list of events those linking to a file. In case a find is linked, the script will load such file, convert it to JSON if necessary and expand the VIS message with its content.
- Once the entire VIS log file is self-contained it is dumped on a non-SQL database. In particular, to a “new data” collection.

At this stage, there is a first enrichment of the data, adding human-readable labels to the data. These labels complement the numerical codes in fields like EventNumber or EventType. This information can be stored in the non-SQL database or in a SQL like one, depending on whether it is used for anything else.

4.2.3 Data processing

The processing goals regarding activity 2 data are currently twofold. On the one hand, there is an aim on building statistics based on the number of messages and types of event, total numbers and per port call, and their evolution over time. On the other hand, the waypoints, speeds and times are another source of valuable information that will be analysed separately.

With these goals in mind, on a daily basis, a script runs to process the data in the new data collection. This script will crawl over the different VIS log documents in it and perform the following actions:

- For each event, extract descriptive information like EventTime, number, instanced or type among others.
- Store this information in the SQL-like database in an events data table. It is important that this table contains the imo number of the involved vessel and the date, so it can be linked to other data eventually.
If the event is of the RTZ type, an additional processing is performed. In such case, the script will first store the vessel IMO, route name, vesselVoyage and route version in the SQL-like database in a rtz_data table. Secondly, it will store the waypoints, the speed at them and the route in a second table rtz_waypoints relating them to the particular route stored in rtz_data.

Finally, once all the events have been processed, the VIS log document is moved from the “new data” collection to a “processed data” collection.

The goal when storing RTZ information is to be able to track the paths followed by the vessels between ports and study the variation on the navigation speed depending on the type of vessel or the time of the year, for instance. Figure 15 shows an overview of the entire process for data gathered from activity 2.

![Figure 15: Overview of the extraction and storage processes of data coming from activity 2.](image)

### 4.3 Expected outputs

The goal of the data gathering and processing stage is to collect the data from the different sources and have it imported into the same database or pool of data. Moreover, these data must contain information that is common to every data source so data from different sources can be linked eventually. Once all the data has been extracted, a set of different processes will be applied on it in order to take advantage of this common information and integrate the data from the different sources, producing a joint output.

This section has described the process followed to collect and import the data from the different data sources and prepare the pool of data from where the final output will be defined.
4.4 Software decisions

This section is not only descriptive, but also based on a series of scripts developed to perform these tasks. In this case, these scripts have been implemented using Python 3, merely due to the skills of the team and its flexibility. Other languages, e.g., Java or C#, could have been used as well.

Similarly, in the case of the database where the Act. Two data is imported, it was internally decided to use a non-relational database, particularly, MongoDB. The reason for opting for a non-relational database is the flexibility it offers to import different types of data with different structures with varying number of fields per register. Relational databases could have been used as well, but the lack of rigidity in the message structure seemed to fit better the non-relational paradigm. Among the available non-relational databases MongoDB was also selected because of the skills of the team. Other databases, like Cassandra or ElasticSearch, could have been used as well.

For the SQL-like database where the pre-processed data from activities 1 and 2 are stored, it was decided to use MySQL. The reasons for using it were mainly that it was the database used in Activity 1, its ease of use and the familiarity of the team with it. Additionally, a SQL-like database was selected because the aim was to produce a set of structured data, with a fix structure.

5 Data Integration

The integration phase aims at merging the data from activities 1 and 2. The goal of this phase is to relate the information of the two independent data sets from Activity 1 and 2. As a result, the output will contain a set of tables with registers including information for end-to-end trips defined by a port of origin, a port of arrival and the navigation in between. In addition to these tables, the previously processed data will help to build statistics that may be relevant for the project by linking these port calls and trips to statistics from the departure and origin ports as well as to the path followed in between.

At this stage, there are three available databases, two SQL-like databases from with data from activities 1 and 2, and a non-SQL database with data from activity 2. The integration phase will not use the latter one, as it acts like a storage database for messages coming from activity 2 and the pre-processed data is already in the SQL-like database. In the case of the database with data from activity 1, the complete states and incomplete states tables are not used either, as they are also message repositories.

The minimum set of information to be put together in the integration phase is:

- Information from the port call of departure.
- Information from the port call of destination.
- Vessel information.
- Information from the route followed by the vessel.
- Events exchanged during the trip.
The first three items can be easily connected from data from activity 1. However, the only link between the information from activities 1 and 2 is the IMO number of the vessel, which is not enough to link the data. Hence, a second field is required to perform this connection, the date. It is easy to track the port calls of a vessel and the dates at which those took place. The time in between can be considered as voyage time. Hence, all the events occurring between those dates for a particular vessel will be associated to the voyage between the previous and posterior ports in its associated list of calls.

Therefore, with access to the port call IDs it is possible to retrieve the dates at which they took place, with those and the imo of the vessel, we perform a date based searched and collect all the events occurring in that period. Similarly, if available, we collect information from the voyage from any associated RTZ file. As a result, we can create a Port-2-Port_data table where each register would have, at least, the following fields:

- Port-2-Port_id: Unique identifier, that acts as primary key of the table, to identify each of the registers of this table.
- Departure_portcallid: unique identifier of the portcallid at the port of departure. It will allow collecting any information related to such call.
- Destination_portcallid: unique identifier of the portcallid at the port of destination. It will allow collecting any information related to such call.
- VesselId: unique identifier, currently uses the IMO number. Used to identify the vessel and collect its data.
- Voyage number: If there is any associated RTZ it will be possible to extract the voyage identifier, which is a unique identifier for the voyage. The voyage number will also allow recovering its waypoints.
- Exchanged events_type_X: It contains the total number of events of type x that were exchanges/sent during that voyage. There are as many fields of this type as different types of event.

A second table named voyage_events is created as well. Each register in this table will contain the Port-2-Port_id and the event id of one of the events accounted for that Port-2-Port_id register. This table will be used to easy locating any particular event related to a port-2-port voyage. Figure 16 summarizes the integration process.
Figure 16: Overview of the data integration process
6 Reference Material

Heurlin H., Andreasson B. Procurement 1.0 of STM ship systems. STM project. Swedish Maritime Administration, 2017.
Heurlin H., Andreasson B. Procurement 2.0 of STM ship systems. STM project. Swedish Maritime Administration, 2017.
Orange. Orange, 2017. Available at: https://orange.biolab.si/
GNU. GNU Operation System. 2017. Available at: https://www.gnu.org/software/spss/
Tableau. Tableau Software. 2017. Available at: https://www.tableau.com
Neo4j. Neo4j software. 2017. Available at: https://neo4j.com/
38 partners from 13 countries -
Creating a safer more efficient and environmentally friendly maritime sector

Demonstrating the function and business value of the Sea Traffic Management concept and its services.

SAFETY - ENVIRONMENT - EFFICIENCY

Swedish Maritime Administration ◦ SSPA ◦ RISE Viktoria ◦ Transas/ Wärtsilä Voyage ◦ Chalmers University of Technology ◦ The Swedish Meteorological and Hydrological Institute ◦ Danish Maritime Authority ◦ Navicon ◦ Novia University of Applied Sciences ◦ Fraunhofer ◦ Carnival Corp. ◦ Italian Ministry of Transport ◦ SASEMAR ◦ Valencia Port Authority ◦ Valencia Port Foundation ◦ CIMNE ◦ University of Catalonia ◦ Norwegian Coastal Administration ◦ GS1 ◦ Cyprus University of Technology ◦ Port of Barcelona ◦ Costa Crociere ◦ Svitzer ◦ OFFIS ◦ Finnish Transport Agency ◦ Southampton Solent University ◦ Frequentis ◦ Wärtsilä SAM Electronics ◦ University of Flensburg ◦ Airbus ◦ Maritiem Instituut Willem Barentsz ◦ SAAB TransponderTech AB ◦ University of Oldenburg ◦ Magellan ◦ Furuno Finland ◦ Rörvik ◦ University of Southampton ◦ HiQ

www.stmvalidation.eu

Co-financed by the Connecting Europe Facility of the European Union