

## PILOT ACTION FINAL REPORT - PORT OF LA SPEZIA

---

DELIVERABLE D.T2.2.4

Version 1

03 2022

---



# Table of contents

- 1. Ex-ante situation ..... 1
- 2. Pilot action description..... 3
- 3. Conclusions ..... 6
- 2. Annex I: SOAP Webservices types definitions ..... 7



# 1. Ex-ante situation

In the last years the Port Authority of the Eastern Ligurian Sea has invested in the digitalization of logistics processes, exploiting the opportunities of European projects. Now the Port Authority has to combine together digitalization and sustainability in order to build a resilient port and logistics hub able to face the new challenges in this sector. For this reason, the Port Authority of the Eastern Ligurian Sea needs to improve the digital collaboration with all the port and logistics actors, strengthening the interoperability between the own IT systems and those of the others public and private administrations.

The railway connection between the port of La Spezia and the Rail-Road terminal of Verona (Zailog, PP02) is already characterized by frequent services. As we can see by the table coming from the M53 “Integrated planning of the railway shunting service in the port network of La Spezia and Marina di Carrara”, the La Spezia-Verona service has a frequency of 5 trains/week.

Treno/Gr. carri	A/P	IF	MTO	Ora Arrivo/Partenza	Intervallo Orario	Provenienza/Destinazione	L	Ma	Me	G	V	S	D	Giorni Sospensione	Ora MAD da IF a ATI	Ora MAD da ATI a IF	Annotazioni	Via
54105	P	MDW	MEDLOG	9:30	9-10	Verona Q.E.		Ma	Me	G	V	S		F		8:00	RID	Pisa
52113	A	MDW	MEDLOG	23:05	23-24	Verona Q.E.	L	Ma	Me	G	V			F	23:20		RID	Pisa

Table 1 - M53 planning - La Spezia - Verona trains

Along this rail network connection, the ICT technologies could help to create a seamless logistics chain on the actual rail corridor between La Spezia and Verona on the SCANMED corridor. Scope of the Pilot Action within the COMODALCE project is sharing some train messages along the connection between La Spezia and Verona.

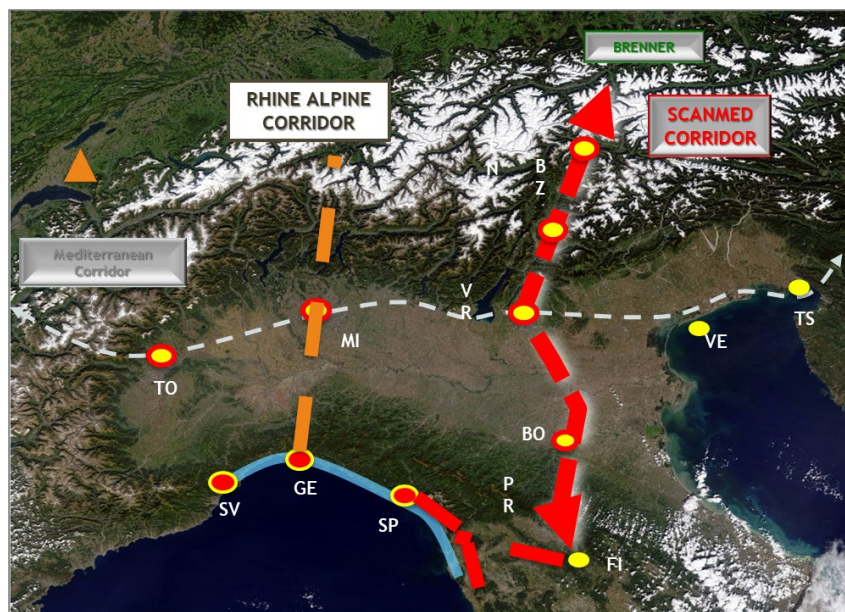


Figure 1 - La Spezia TEN-T connections

After the market analysis, the Eastern Ligurian Sea Port Authority therefore convened and conducted a series of conference calls with the actors involved:



- The technology provider;
- The Railway Company;
- The Multimodal Transport Operator;
- The Port Terminal;
- The Manager of the Inland Terminal in Verona.

The steps made it possible to highlight the details of the feasibility of the Project and of the sub-activities included, bearing in mind the current operational context and the needs expressed by the operators themselves.

The port of La Spezia capitalized an ICT integrated platform for the management of the rail freight corridor La Spezia - Verona. This logistic corridor information pipeline (namely CMP - Corridor Management Platform) represents one of the terrestrial extensions of the ICT systems already in use.

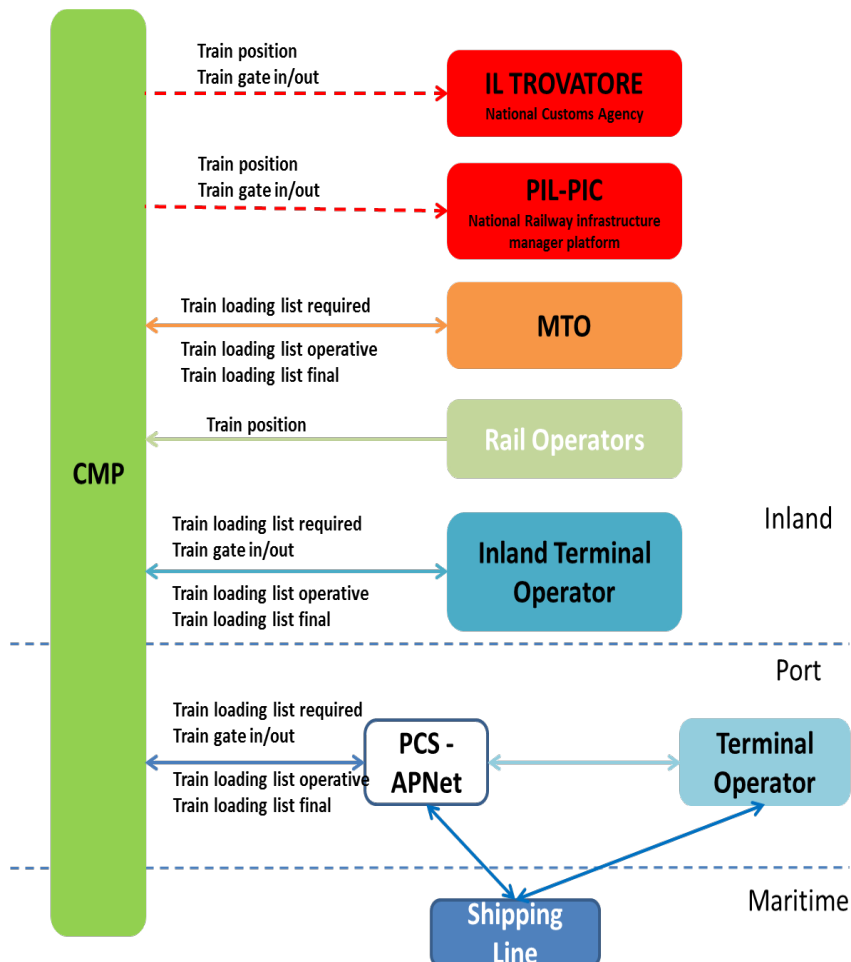


Figure 2 - Corridor Management Platform architecture



## 2. Pilot action description

As part of the COMODALCE European Project, aimed at the implementation of a pilot action consisting of the interoperability between the information systems of the supply chain actors operating on the La Spezia - Verona section, important advances were made, which made it possible to define in detail the operational and technical aspects of the data exchange, in line with the objectives of the Project.

Especially, it is emphasized that the COMODALCE Project fits punctually within an evolutionary context regarding the Port of La Spezia and its hinterland: the expansion of the **Corridor Management Platform**, a digital platform used by the Port Authority for the full digital management of information related to the railway customs corridors, will be expanded during the year, in order to cover all railway traffic to and from the Port, including new technological functions related to the operational and customs aspects of railway trains.

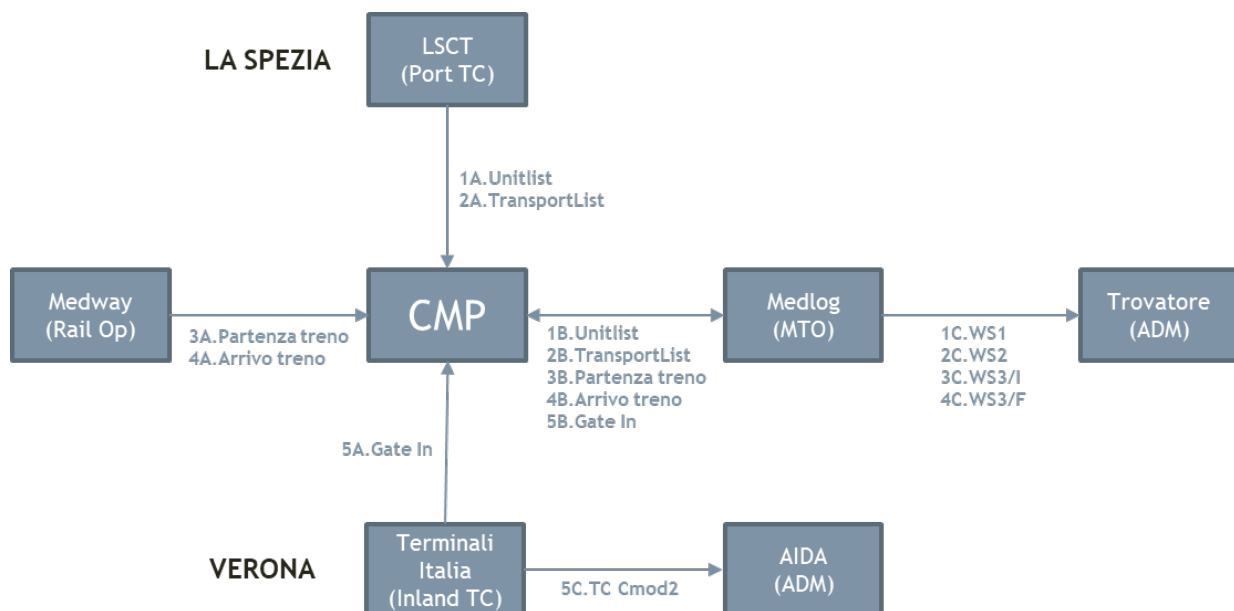


Figure 3 - Information flows managed by the COMODALCE Pilot Action

In this context, the COMODALCE project implemented a fully digitized information exchange between the players operating on the La Spezia - Verona railway section, in order to ensure a more efficient and effective information exchange, and to ensure greater added value to the intermodal corridor.

### Process features and technical details

Below is provided a high-level view on the processes that fall within the domain of the Project, and on the flows exchanged by operators thanks to the new technological solution implemented.

Subsequently, indications are also provided about the technical specifications of the Services that will be implemented.

First, as previously reported, the COMODALCE Project aims to establish interoperability between Medlog (MTO), Medway (Railway Company), LSCT (Port Terminal) and Terminali Italia (Inland Terminal), in Port Temporary Storage and the Inland one; with this in mind, interoperability with the ITrovatore and AIDA platforms, both managed by ADM, is also envisaged.

For each flow implemented, the mediation of the Corridor Management Platform (CMP) is envisaged.

The following are the main points of the TO BE Process:



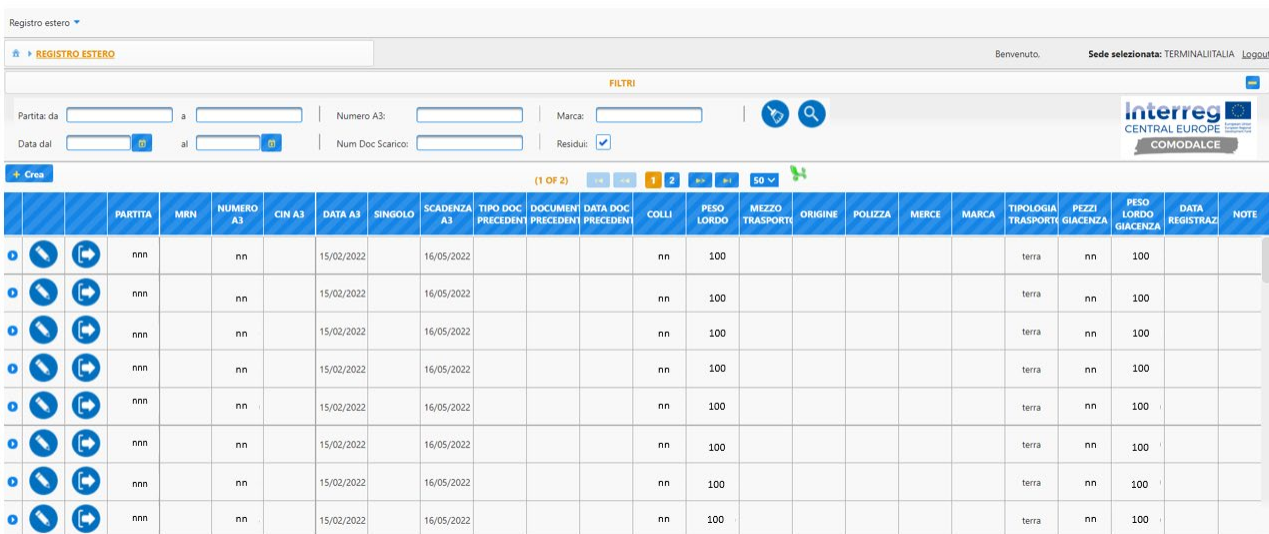
1. The MTO shares the railway Loading List with the CMP through the UnitList service; the Loading List is then forwarded by the CMP to the Port Terminal. At the same time, through the WS1 Service, the MTO communicates the same list to IITrovatore, for the enabling of the ITUs in the Customs Corridor, receiving a result.
2. The Port Terminal shares the Train Composition with the CMP through the TransportList Service; the CMP forwards the Train Composition to the MTO. At the same time, through the WS2 Service (Mission Registration), the composition is sent to IITrovatore, which returns a result.
3. The Railway Company communicates the Train Departure to the CMP, and the latter forwards the message to the MTO. The WS3 / I Service communicates the start of the Customs Mission to IITrovatore.
4. The Railway Company communicates the Arrival Train at the destination station to the CMP, and the latter forwards the message to the MTO. The WS3 / F Service communicates the end of the Customs Mission to IITrovatore.
5. The Inland Terminal communicates the Gate-In to the CMP, which forwards the message to the MTO. At the same time, the Inland Terminal uses the TC Mod2 Service to communicate the message to AIDA.

Data exchange between the CMP and the actors is implemented through a SOAP Webservice. Messages representing atomic events are implemented with the “SendEvent” element while more structured data is implemented with the “TransportList” element.

The SendEvent message relates an object with an event type, a location and a timestamp.

The TransportList message is composed by a header and a list of details. The header contains an identification of the message, the message type, identification of the sender and of the recipients and the common data of the message. Each detail element refer to a specific object and can contain a generic list of key-value pairs.

Through the systems implemented, the Customs Corridor can be managed in an automated and full digital way, pursuing criteria of interoperability between the actors, digitalization and standardization of flows, security of information and procedures, full visibility and monitoring of processes.

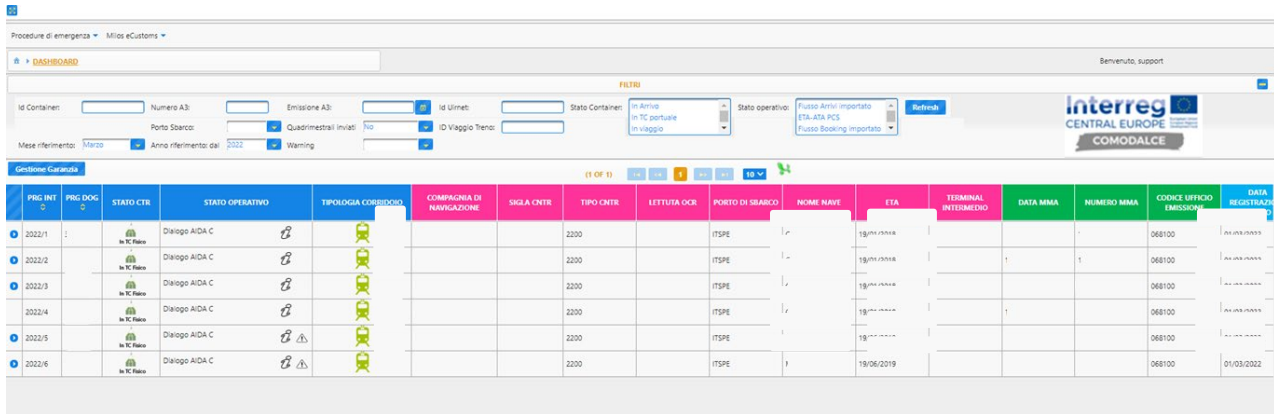


	PARTITA	MIRN	NUMERO A3	CIN A3	DATA A3	SINGOLO	SCADENZA A3	TIPO DOC PRECEDENT	DOCUMENT PRECEDENT	DATA DOC PRECEDENT	COLLI	PESO LORDO	MEZZO TRASPORTI	ORIGINE	POLIZZA	MERCE	MARCA	TIPOLOGIA TRASPORTI	PEZZI GIACENZA	PESO LORDO GIACENZA	DATA REGISTRAZ	NOTE
1	nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
2	nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
3	nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
4	nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
5	nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
6	nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
7	nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
8	nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		

Figure 4 - Inland terminal operators Dashboard



From its monitoring dashboard, the Inland Terminal can monitor in real time the status of its own customs temporary storage warehouse, through the indication of the remaining consignments, and an articulated series of information about the latter.



The dashboard includes a filter section with the following fields: Id Container, Numero A3, Emissione A3, Id Umet, Stato Container (In-Arrivo, In-TC portuale, In viaggio), Stato operativo (Passo Arrivo importato, ETA-ATA PCS, Passo Booking importato), Mese riferimento (Marzo), Anno riferimento dal (2022), and Warning. The main table displays the following data:

PRG INT	PRG DOC	STATO CTR	STATO OPERATIVO	TIPOLOGIA COORDINATO	COMPAGNIA DI NAVIGAZIONE	SIGLA CNTR	TIPO CNTR	LETTURA OCR	PORTO DI SBARCO	NUMERO NAVE	ETA	TERMINAL INTERMEDIO	DATA MMA	NUMERO MMA	CODICE UFFICIO EMISSIONE	DATA REGISTRAZIONE
2022/1		In TC Passo	Dialogo A/IDA C				2200		ITSPE		18/06/2019				068100	01/03/2022
2022/2		In TC Passo	Dialogo A/IDA C				2200		ITSPE		18/06/2019		1	1	068100	
2022/3		In TC Passo	Dialogo A/IDA C				2200		ITSPE		18/06/2019				068100	
2022/4		In TC Passo	Dialogo A/IDA C				2200		ITSPE		18/06/2019		1		068100	
2022/5		In TC Passo	Dialogo A/IDA C				2200		ITSPE		18/06/2019				068100	
2022/6		In TC Passo	Dialogo A/IDA C				2200		ITSPE		18/06/2019				068100	01/03/2022

Figure 5 - Multimodal Transport Operator Dashboard

From its interactive dashboard, the Customs Mission Manager (MTO) can monitor in real time the Units entrusted to it for the movement between the Port and Inland Temporary Storage. The Dashboard includes various information regarding the units to ensure full visibility and control, such as customs status, operational status, shipping company, container number, weight, ship ETA, Inland terminal, MMA number and date, registration date.



### 3. Conclusions

The COMODALCE Pilot Action tests showed several improvements within the logistic process, such as:

- better efficiency and flexibility of the processes, quality of information, goods traceability, and reduction of administrative costs and attribution errors;
- deeper integration and the information sharing between the different stakeholders of the entire logistic chain;
- more security and reliability by offering standardized and qualified services;
- interoperability ensured by implementing standard procedures for the information exchange;
- more and better services to the logistic corridor stakeholders in order to exploit economies of scale and be more competitive from an operating and economic point of view;
- reduction of the transport costs per unit;
- Reduction of cargo handling time.

The following table gives an overall qualitative impact assessment of the Pilot Action:

<i>Benefits</i>	<i>Qualitative evaluation</i>
Improve the efficiency and flexibility of processes, the quality of real-time information and traceability of cargo, reducing administrative costs and procedural errors.	Very important - CMP framework reduce administrative costs and improve the quality of real time information.
Improve the integration and sharing of information between different actors in the logistics chain.	Very important - The CMP framework connect the actors operating systems involved in the process by using specific connectors.
Increase visibility and predictability throughout the transport chain - Boosting reducing costs of production / stock, as well as the logistics costs, offering multimodal optimized transport solutions.	Important - CMP efficiency has a positive impact on the small/medium actors potentially involved in the process.
Improve security transparency and reliability, offering standardized and qualified services.	Important - It is the basis of the relationship between the partners involved in the transport chain.
Provide IT support services to ensure interoperability and implement standard procedures and mechanisms for data exchange	Very important - Standardization, exchange and interoperability are needed in the whole transport chain operations.

*Table 2 - Pilot Action qualitative impact assessment*

The development of functionalities of the Corridor Management Platform, Pilot Action of the COMODALCE project for the port of La Spezia, is satisfying this wish list of pre-identified ICT measures. The operators involved along the supply chain shared their needs the detailed analysis of the information flows of the actual IT systems managed by the companies involved and the design of the new solution through the COMODALCE Pilot Action.





## 2. Annex I: SOAP Webservices types definitions

In this section the messages definition is reported.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="unqualified"
targetNamespace="http://webservice.tt.cap.it/" xmlns:tns="http://webservice.tt.cap.it/"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="basicAuthenticate" type="tns:basicAuthenticate"/>
<xs:element name="basicAuthenticateResponse" type="tns:basicAuthenticateResponse"/>
<xs:element name="closeCycle" type="tns:closeCycle"/>
<xs:element name="closeCycleResponse" type="tns:closeCycleResponse"/>
<xs:element name="getEventSites" type="tns:getEventSites"/>
<xs:element name="getEventSitesResponse" type="tns:getEventSitesResponse"/>
<xs:element name="getEvents" type="tns:getEvents"/>
<xs:element name="getEventsResponse" type="tns:getEventsResponse"/>
<xs:element name="getObjectTypes" type="tns:getObjectTypes"/>
<xs:element name="getObjectTypesResponse" type="tns:getObjectTypesResponse"/>
<xs:element name="getQueueStatus" type="tns:getQueueStatus"/>
<xs:element name="getQueueStatusResponse" type="tns:getQueueStatusResponse"/>
<xs:element name="getSets" type="tns:getSets"/>
<xs:element name="getSetsResponse" type="tns:getSetsResponse"/>
<xs:element name="getTrainList" type="tns:getTrainList"/>
<xs:element name="getTrainListResponse" type="tns:getTrainListResponse"/>
<xs:element name="getTransportList" type="tns:getTransportList"/>
<xs:element name="getTransportListResponse" type="tns:getTransportListResponse"/>
<xs:element name="getUsers" type="tns:getUsers"/>
<xs:element name="getUsersInfo" type="tns:getUsersInfo"/>
<xs:element name="getUsersInfoResponse" type="tns:getUsersInfoResponse"/>
<xs:element name="getUsersResponse" type="tns:getUsersResponse"/>
<xs:element name="getYardPosition" type="tns:getYardPosition"/>
<xs:element name="getYardPositionResponse" type="tns:getYardPositionResponse"/>
<xs:element name="importTransportList" type="tns:importTransportList"/>
<xs:element name="importTransportListResponse" type="tns:importTransportListResponse"/>
<xs:element name="openCycle" type="tns:openCycle"/>
<xs:element name="openCycleResponse" type="tns:openCycleResponse"/>
<xs:element name="sendEvent" type="tns:sendEvent"/>
<xs:element name="sendEventByInspection" type="tns:sendEventByInspection"/>
<xs:element name="sendEventByInspectionResponse" type="tns:sendEventByInspectionResponse"/>
<xs:element name="sendEventByOffset" type="tns:sendEventByOffset"/>
<xs:element name="sendEventByOffsetResponse" type="tns:sendEventByOffsetResponse"/>
<xs:element name="sendEventResponse" type="tns:sendEventResponse"/>
<xs:element name="sendNotification" type="tns:sendNotification"/>
<xs:element name="sendNotificationResponse" type="tns:sendNotificationResponse"/>
<xs:element name="setExportedTrainList" type="tns:setExportedTrainList"/>
<xs:element name="setExportedTrainListResponse" type="tns:setExportedTrainListResponse"/>
<xs:element name="setRelation" type="tns:setRelation"/>
<xs:element name="setRelationResponse" type="tns:setRelationResponse"/>
<xs:element name="test" type="tns:test"/>
<xs:element name="testResponse" type="tns:testResponse"/>
<xs:complexType name="getTrainList">
<xs:sequence>
<xs:element minOccurs="0" name="inputTrainList" type="tns:inputTrainList"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="inputTrainList">
<xs:complexContent>
<xs:extension base="tns:inputBase">
<xs:sequence>
<xs:element minOccurs="0" name="callId" type="xs:string"/>
<xs:element minOccurs="0" name="exported" type="xs:string"/>

```



```

<xs:element minOccurs="0" name="fromEta" type="xs:string"/>
<xs:element minOccurs="0" name="requestTime" type="xs:string"/>
<xs:element minOccurs="0" name="toEta" type="xs:string"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="inputBase">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="keyToSearch" type="xs:string"/>
<xs:element minOccurs="0" name="messageDateTime" type="xs:string"/>
<xs:element minOccurs="0" name="messageId" type="xs:long"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="baseBean">
<xs:sequence/>
</xs:complexType>
<xs:complexType name="getTrainListResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputTrainList"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="outputTrainList">
<xs:complexContent>
<xs:extension base="tns:outputBase">
<xs:sequence>
<xs:element maxOccurs="unbounded" minOccurs="0" name="trainList" nillable="true"
type="tns:bTrainListBean"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="outputBase">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="returnCode" type="xs:string"/>
<xs:element minOccurs="0" name="returnDateTime" type="xs:string"/>
<xs:element minOccurs="0" name="returnDescription" type="xs:string"/>
<xs:element minOccurs="0" name="returnId" type="xs:long"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="bTrainListBean">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="ata" type="xs:dateTime"/>
<xs:element minOccurs="0" name="atd" type="xs:dateTime"/>
<xs:element minOccurs="0" name="callId" type="xs:string"/>
<xs:element minOccurs="0" name="callStatus" type="xs:string"/>
<xs:element minOccurs="0" name="closeTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="descrizione" type="xs:string"/>
<xs:element minOccurs="0" name="eta" type="xs:dateTime"/>
<xs:element minOccurs="0" name="etd" type="xs:dateTime"/>
<xs:element minOccurs="0" name="id" type="xs:int"/>
<xs:element minOccurs="0" name="openTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="train" type="xs:string"/>
<xs:element maxOccurs="unbounded" minOccurs="0" name="trainListDetail" nillable="true"
type="tns:bTrainListDetailBean"/>

```



```

</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="bTrainListDetailBean">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="booking" type="xs:string"/>
<xs:element minOccurs="0" name="car" type="xs:string"/>
<xs:element minOccurs="0" name="carType" type="xs:string"/>
<xs:element minOccurs="0" name="carWeight" type="xs:string"/>
<xs:element minOccurs="0" name="containerId" type="xs:string"/>
<xs:element minOccurs="0" name="containerType" type="xs:string"/>
<xs:element minOccurs="0" name="esealId" type="xs:string"/>
<xs:element minOccurs="0" name="forwarder" type="xs:string"/>
<xs:element minOccurs="0" name="fullEmpty" type="xs:string"/>
<xs:element minOccurs="0" name="goods" type="xs:string"/>
<xs:element minOccurs="0" name="grossWeight" type="xs:string"/>
<xs:element minOccurs="0" name="imdg" type="xs:string"/>
<xs:element minOccurs="0" name="loadStatus" type="xs:string"/>
<xs:element minOccurs="0" name="mainVessel" type="xs:string"/>
<xs:element minOccurs="0" name="oog" type="xs:string"/>
<xs:element minOccurs="0" name="operator" type="xs:string"/>
<xs:element minOccurs="0" name="otherEsealId" type="xs:string"/>
<xs:element minOccurs="0" name="pod" type="xs:string"/>
<xs:element minOccurs="0" name="position" type="xs:string"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="sendEventByOffset">
<xs:sequence>
<xs:element minOccurs="0" name="InputEventByOffset" type="tns:inputEventByOffset"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="inputEventByOffset">
<xs:complexContent>
<xs:extension base="tns:inputEvent">
<xs:sequence>
<xs:element minOccurs="0" name="offsetTime" type="xs:string"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="inputEvent">
<xs:complexContent>
<xs:extension base="tns:inputBase">
<xs:sequence>
<xs:element minOccurs="0" name="event" type="xs:string"/>
<xs:element minOccurs="0" name="eventSite" type="xs:string"/>
<xs:element maxOccurs="unbounded" minOccurs="0" name="inputEventFields" nillable="true"
type="tns:inputEventFields"/>
<xs:element minOccurs="0" name="localTime" type="xs:string"/>
<xs:element minOccurs="0" name="objectInfo" type="tns:objectInfo"/>
<xs:element minOccurs="0" name="source" type="xs:string"/>
<xs:element minOccurs="0" name="gpSCoordinates" type="xs:string"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="inputEventFields">
<xs:sequence>
<xs:element minOccurs="0" name="fieldCode" type="xs:string"/>
<xs:element minOccurs="0" name="fieldForced" type="xs:string"/>

```



```

<xs:element minOccurs="0" name="fieldValue" type="xs:string"/>
<xs:element maxOccurs="unbounded" minOccurs="0" name="inputEventFieldsExt" nillable="true"
type="tns:inputEventFieldsExt"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="inputEventFieldsExt">
<xs:sequence>
<xs:element minOccurs="0" name="fieldCode" type="xs:string"/>
<xs:element minOccurs="0" name="fieldForced" type="xs:string"/>
<xs:element minOccurs="0" name="fieldValue" type="xs:string"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="objectInfo">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="code" type="xs:string"/>
<xs:element minOccurs="0" name="type" type="xs:string"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="sendEventByOffsetResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="closeCycle">
<xs:sequence>
<xs:element minOccurs="0" name="inputCycle" type="tns:inputCycle"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="inputCycle">
<xs:complexContent>
<xs:extension base="tns:inputBase">
<xs:sequence>
<xs:element minOccurs="0" name="BPROCESSIID" type="xs:string"/>
<xs:element minOccurs="0" name="endCycle" type="xs:string"/>
<xs:element minOccurs="0" name="objectInfo" type="tns:objectInfo"/>
<xs:element minOccurs="0" name="source" type="xs:string"/>
<xs:element minOccurs="0" name="startCycle" type="xs:string"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="closeCycleResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="test">
<xs:sequence/>
</xs:complexType>
<xs:complexType name="testResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="xs:string"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="sendEvent">
<xs:sequence>
<xs:element minOccurs="0" name="inputEvent" type="tns:inputEvent"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="sendEventResponse">
<xs:sequence>

```



```

<xs:element minOccurs="0" name="return" type="tns:outputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="sendEventByInspection">
<xs:sequence>
<xs:element minOccurs="0" name="InputEventByInspection" type="tns:inputEventByInspection"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="inputEventByInspection">
<xs:complexContent>
<xs:extension base="tns:inputBase">
<xs:sequence>
<xs:element minOccurs="0" name="eventSite" type="xs:string"/>
<xs:element minOccurs="0" name="localTime" type="xs:string"/>
<xs:element minOccurs="0" name="memoryRead" type="xs:string"/>
<xs:element minOccurs="0" name="objectInfo" type="tns:objectInfo"/>
<xs:element minOccurs="0" name="openClosed" type="xs:string"/>
<xs:element minOccurs="0" name="source" type="xs:string"/>
<xs:element minOccurs="0" name="tampReg" type="xs:string"/>
<xs:element minOccurs="0" name="gPSCoordinates" type="xs:string"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="sendEventByInspectionResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="getObjectTypes">
<xs:sequence>
<xs:element minOccurs="0" name="inputBase" type="tns:inputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="getObjectTypesResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputObjectsType"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="outputObjectsType">
<xs:complexContent>
<xs:extension base="tns:outputBase">
<xs:sequence>
<xs:element maxOccurs="unbounded" minOccurs="0" name="objectsType" nillable="true"
type="tns:bTipoOggettoBean"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="bTipoOggettoBean">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="BTIPIOGGETTOId" type="xs:int"/>
<xs:element minOccurs="0" name="codTipoOggetto" type="xs:string"/>
<xs:element name="createBUTENTIId" type="xs:int"/>
<xs:element minOccurs="0" name="createTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="desTipoOggetto" type="xs:string"/>
<xs:element minOccurs="0" name="tipoCruscotto" type="xs:string"/>
<xs:element name="updateBUTENTIId" type="xs:int"/>
<xs:element minOccurs="0" name="updateTime" type="xs:dateTime"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```



```

<xs:complexType name="getTransportList">
  <xs:sequence>
    <xs:element minOccurs="0" name="inputTransportList" type="tns:inputTransportList"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="inputTransportList">
  <xs:complexContent>
    <xs:extension base="tns:inputBase">
      <xs:sequence>
        <xs:element minOccurs="0" name="codLuogo" type="xs:string"/>
        <xs:element minOccurs="0" name="idLista" type="xs:string"/>
        <xs:element minOccurs="0" name="idProcesso" type="xs:string"/>
        <xs:element minOccurs="0" name="requestTime" type="xs:string"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="getTransportListResponse">
  <xs:sequence>
    <xs:element minOccurs="0" name="return" type="tns:outputTransportList"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="outputTransportList">
  <xs:complexContent>
    <xs:extension base="tns:outputBase">
      <xs:sequence>
        <xs:element maxOccurs="unbounded" minOccurs="0" name="transportList" nillable="true"
          type="tns:bTransportListBean"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="bTransportListBean">
  <xs:complexContent>
    <xs:extension base="tns:baseBean">
      <xs:sequence>
        <xs:element minOccurs="0" name="codLuogo" type="xs:string"/>
        <xs:element minOccurs="0" name="codOggetto" type="xs:string"/>
        <xs:element minOccurs="0" name="descrizione" type="xs:string"/>
        <xs:element minOccurs="0" name="idLista" type="xs:string"/>
        <xs:element minOccurs="0" name="idProcesso" type="xs:string"/>
        <xs:element minOccurs="0" name="requestTime" type="xs:string"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="setExportedTrainList">
  <xs:sequence>
    <xs:element minOccurs="0" name="inputTrainList" type="tns:inputTrainList"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="setExportedTrainListResponse">
  <xs:sequence>
    <xs:element minOccurs="0" name="return" type="tns:outputTrainList"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="getYardPosition">
  <xs:sequence>
    <xs:element minOccurs="0" name="qrCode" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="getYardPositionResponse">
  <xs:sequence>
    <xs:element minOccurs="0" name="return" type="tns:outputYardPosition"/>
  </xs:sequence>

```



```

</xs:complexType>
<xs:complexType name="outputYardPosition">
<xs:complexContent>
<xs:extension base="tns:outputBase">
<xs:sequence>
<xs:element minOccurs="0" name="yardPosition" type="xs:string"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="getSets">
<xs:sequence>
<xs:element minOccurs="0" name="inputBase" type="tns:inputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="getSetsResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputSets"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="outputSets">
<xs:complexContent>
<xs:extension base="tns:outputBase">
<xs:sequence>
<xs:element maxOccurs="unbounded" minOccurs="0" name="insiemi" nillable="true"
type="tns:iInsiemeBean"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="iInsiemeBean">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element name="createBUTENTIId" type="xs:int"/>
<xs:element minOccurs="0" name="createTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="desInsieme" type="xs:string"/>
<xs:element name="updateBUTENTIId" type="xs:int"/>
<xs:element minOccurs="0" name="updateTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="bTIPIOGGETTOId1" type="tns:bTipoOggettoBean"/>
<xs:element minOccurs="0" name="bTIPIOGGETTOId2" type="tns:bTipoOggettoBean"/>
<xs:element minOccurs="0" name="iINSIEMIId" type="xs:int"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="getEvents">
<xs:sequence>
<xs:element minOccurs="0" name="inputBase" type="tns:inputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="getEventsResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputEvents"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="outputEvents">
<xs:complexContent>
<xs:extension base="tns:outputBase">
<xs:sequence>
<xs:element maxOccurs="unbounded" minOccurs="0" name="events" nillable="true"
type="tns:bEventoBean"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
    
```



```

</xs:complexType>
<xs:complexType name="bEventoBean">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="BEVENTIIId" type="xs:int"/>
<xs:element minOccurs="0" name="codEvento" type="xs:string"/>
<xs:element name="createBUTENTIIId" type="xs:int"/>
<xs:element minOccurs="0" name="createTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="desEvento" type="xs:string"/>
<xs:element maxOccurs="unbounded" minOccurs="0" name="fieldList" nillable="true" type="tns:bField"/>
<xs:element minOccurs="0" name="mobile" type="xs:string"/>
<xs:element name="updateBUTENTIIId" type="xs:int"/>
<xs:element minOccurs="0" name="updateTime" type="xs:dateTime"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="bField">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="descrizione" type="xs:string"/>
<xs:element minOccurs="0" name="label" type="xs:string"/>
<xs:element name="maxVal" type="xs:int"/>
<xs:element name="minVal" type="xs:int"/>
<xs:element minOccurs="0" name="obbligatorio" type="xs:string"/>
<xs:element name="ordine" type="xs:int"/>
<xs:element minOccurs="0" name="regExp" type="xs:string"/>
<xs:element minOccurs="0" name="tipoCampo" type="xs:string"/>
<xs:element minOccurs="0" name="bCAMPIId" type="xs:int"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="basicAuthenticate">
<xs:sequence/>
</xs:complexType>
<xs:complexType name="basicAuthenticateResponse">
<xs:sequence>
<xs:element name="return" type="xs:boolean"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="getQueueStatus">
<xs:sequence>
<xs:element name="id" type="xs:int"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="getQueueStatusResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:queueStatus"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="queueStatus">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="createTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="jobNote" type="xs:string"/>
<xs:element minOccurs="0" name="jobStatus" type="xs:string"/>
<xs:element minOccurs="0" name="updateTime" type="xs:dateTime"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```





```

<xs:complexType name="getEventSites">
<xs:sequence>
<xs:element minOccurs="0" name="inputBase" type="tns:inputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="getEventSitesResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputEventSites"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="outputEventSites">
<xs:complexContent>
<xs:extension base="tns:outputBase">
<xs:sequence>
<xs:element maxOccurs="unbounded" minOccurs="0" name="eventList" nillable="true"
type="tns:bLuogoBean"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="bLuogoBean">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="BLUOGHIId" type="xs:int"/>
<xs:element minOccurs="0" name="codLuogo" type="xs:string"/>
<xs:element minOccurs="0" name="coordinateGPS" type="xs:string"/>
<xs:element name="createBUTENTIId" type="xs:int"/>
<xs:element minOccurs="0" name="createTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="desLuogo" type="xs:string"/>
<xs:element minOccurs="0" name="gap" type="xs:string"/>
<xs:element name="updateBUTENTIId" type="xs:int"/>
<xs:element minOccurs="0" name="updateTime" type="xs:dateTime"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="getUsers">
<xs:sequence>
<xs:element minOccurs="0" name="inputBase" type="tns:inputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="getUsersResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputUsers"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="outputUsers">
<xs:complexContent>
<xs:extension base="tns:outputBase">
<xs:sequence>
<xs:element maxOccurs="unbounded" minOccurs="0" name="users" nillable="true" type="tns:bUtenteBean"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="bUtenteBean">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="BUTENTIId" type="xs:int"/>
<xs:element minOccurs="0" name="codUtente" type="xs:string"/>
<xs:element name="createBUTENTIId" type="xs:int"/>
<xs:element minOccurs="0" name="createTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="desUtente" type="xs:string"/>

```



```

<xs:element minOccurs="0" name="email" type="xs:string"/>
<xs:element minOccurs="0" name="password" type="xs:string"/>
<xs:element name="updateBUTENTIIId" type="xs:int"/>
<xs:element minOccurs="0" name="updateTime" type="xs:dateTime"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="setRelation">
<xs:sequence>
<xs:element minOccurs="0" name="inputRelation" type="tns:inputRelation"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="inputRelation">
<xs:complexContent>
<xs:extension base="tns:inputBase">
<xs:sequence>
<xs:element minOccurs="0" name="description" type="xs:string"/>
<xs:element minOccurs="0" name="end" type="xs:string"/>
<xs:element minOccurs="0" name="firstObjectInfo" type="tns:objectInfo"/>
<xs:element minOccurs="0" name="GPSCoordinates" type="xs:string"/>
<xs:element minOccurs="0" name="secondObjectInfo" type="tns:objectInfo"/>
<xs:element minOccurs="0" name="site" type="xs:string"/>
<xs:element minOccurs="0" name="source" type="xs:string"/>
<xs:element minOccurs="0" name="start" type="xs:string"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="setRelationResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="getUsersInfo">
<xs:sequence>
<xs:element minOccurs="0" name="inputBase" type="tns:inputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="getUsersInfoResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputUsersInfo"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="outputUsersInfo">
<xs:complexContent>
<xs:extension base="tns:outputBase">
<xs:sequence>
<xs:element maxOccurs="unbounded" minOccurs="0" name="userInfoList" nillable="true"
type="tns:bUserInfoBean"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="bUserInfoBean">
<xs:sequence>
<xs:element minOccurs="0" name="codUtente" type="xs:string"/>
<xs:element minOccurs="0" name="desProcesso" type="xs:string"/>
<xs:element minOccurs="0" name="desSede" type="xs:string"/>
<xs:element minOccurs="0" name="bPROCESSIIId" type="xs:int"/>
<xs:element minOccurs="0" name="bSEDIId" type="xs:int"/>
<xs:element minOccurs="0" name="bUTENTIIId" type="xs:int"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="sendNotification">

```



```

<xs:sequence>
<xs:element minOccurs="0" name="notificationInputBean" type="tns:notificationInputBean"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="notificationInputBean">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element minOccurs="0" name="codiceEvento" type="xs:string"/>
<xs:element maxOccurs="unbounded" minOccurs="0" name="params" nillable="true" type="xs:anyType"/>
<xs:element name="sede" type="xs:int"/>
<xs:element name="userid" type="xs:int"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="sendNotificationResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="importTransportList">
<xs:sequence>
<xs:element minOccurs="0" name="transportListBean" type="tns:transportListBean"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="transportListBean">
<xs:complexContent>
<xs:extension base="tns:baseBean">
<xs:sequence>
<xs:element maxOccurs="unbounded" minOccurs="0" name="additionalHeaderField" nillable="true"
type="tns:additionalHeaderField"/>
<xs:element minOccurs="0" name="arrivalCode" type="xs:string"/>
<xs:element minOccurs="0" name="ata" type="xs:dateTime"/>
<xs:element minOccurs="0" name="atd" type="xs:dateTime"/>
<xs:element minOccurs="0" name="closeDateTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="code" type="xs:string"/>
<xs:element minOccurs="0" name="departureCode" type="xs:string"/>
<xs:element minOccurs="0" name="departureOffice" type="xs:string"/>
<xs:element minOccurs="0" name="description" type="xs:string"/>
<xs:element minOccurs="0" name="destinationOffice" type="xs:string"/>
<xs:element minOccurs="0" name="eta" type="xs:dateTime"/>
<xs:element minOccurs="0" name="etd" type="xs:dateTime"/>
<xs:element minOccurs="0" name="openDateTime" type="xs:dateTime"/>
<xs:element minOccurs="0" name="ownerCode" type="xs:string"/>
<xs:element minOccurs="0" name="processId" type="xs:string"/>
<xs:element minOccurs="0" name="ptd" type="xs:dateTime"/>
<xs:element minOccurs="0" name="status" type="xs:string"/>
<xs:element maxOccurs="unbounded" minOccurs="0" name="transportListDetail" nillable="true"
type="tns:transportListDetail"/>
<xs:element minOccurs="0" name="voyageId" type="xs:string"/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="additionalHeaderField">
<xs:sequence>
<xs:element minOccurs="0" name="fieldCode" type="xs:string"/>
<xs:element minOccurs="0" name="fieldForced" type="xs:string"/>
<xs:element minOccurs="0" name="fieldValue" type="xs:string"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="transportListDetail">
<xs:sequence>

```



```

<xs:element maxOccurs="unbounded" minOccurs="0" name="additionalDetailField" nillable="true"
type="tns:additionalDetailField"/>
<xs:element minOccurs="0" name="booking" type="xs:string"/>
<xs:element minOccurs="0" name="car" type="xs:string"/>
<xs:element minOccurs="0" name="carPos" type="xs:string"/>
<xs:element minOccurs="0" name="carType" type="xs:string"/>
<xs:element minOccurs="0" name="carWeight" type="xs:string"/>
<xs:element minOccurs="0" name="companyCode" type="xs:string"/>
<xs:element minOccurs="0" name="companyDescription" type="xs:string"/>
<xs:element maxOccurs="unbounded" minOccurs="0" name="customsDocuments" nillable="true"
type="tns:customsDocuments"/>
<xs:element minOccurs="0" name="customsState" type="xs:string"/>
<xs:element minOccurs="0" name="deliveryOrder" type="xs:string"/>
<xs:element minOccurs="0" name="departureSiding" type="xs:string"/>
<xs:element minOccurs="0" name="destinationSiding" type="xs:string"/>
<xs:element minOccurs="0" name="destinationStation" type="xs:string"/>
<xs:element minOccurs="0" name="errorCode" type="xs:string"/>
<xs:element minOccurs="0" name="errorDescription" type="xs:string"/>
<xs:element minOccurs="0" name="finalDestination" type="xs:string"/>
<xs:element minOccurs="0" name="forwarder" type="xs:string"/>
<xs:element minOccurs="0" name="forwarderDescription" type="xs:string"/>
<xs:element minOccurs="0" name="fullEmpty" type="xs:string"/>
<xs:element minOccurs="0" name="goods" type="xs:string"/>
<xs:element minOccurs="0" name="grossWeight" type="xs:string"/>
<xs:element minOccurs="0" name="imdg" type="xs:string"/>
<xs:element minOccurs="0" name="loadStatus" type="xs:string"/>
<xs:element minOccurs="0" name="mainVessel" type="xs:string"/>
<xs:element minOccurs="0" name="meanType" type="xs:string"/>
<xs:element minOccurs="0" name="merchantCarrier" type="xs:string"/>
<xs:element minOccurs="0" name="missionId" type="xs:string"/>
<xs:element minOccurs="0" name="missionStatus" type="xs:string"/>
<xs:element minOccurs="0" name="oog" type="xs:string"/>
<xs:element minOccurs="0" name="operator" type="xs:string"/>
<xs:element minOccurs="0" name="otherSealId" type="xs:string"/>
<xs:element minOccurs="0" name="pod" type="xs:string"/>
<xs:element minOccurs="0" name="pos" type="xs:string"/>
<xs:element minOccurs="0" name="processId" type="xs:string"/>
<xs:element minOccurs="0" name="seal1" type="xs:string"/>
<xs:element minOccurs="0" name="transportUnit" type="xs:string"/>
<xs:element minOccurs="0" name="unitId" type="xs:string"/>
<xs:element minOccurs="0" name="unitSize" type="xs:string"/>
<xs:element minOccurs="0" name="unitType" type="xs:string"/>
<xs:element minOccurs="0" name="vesselDescription" type="xs:string"/>
<xs:element minOccurs="0" name="vesselVoyageId" type="xs:string"/>
<xs:element minOccurs="0" name="eSealId" type="xs:string"/>
<xs:element minOccurs="0" name="pCheckTime" type="xs:dateTime"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="additionalDetailField">
<xs:sequence>
<xs:element minOccurs="0" name="fieldCode" type="xs:string"/>
<xs:element minOccurs="0" name="fieldForced" type="xs:string"/>
<xs:element minOccurs="0" name="fieldValue" type="xs:string"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="customsDocuments">
<xs:sequence>
<xs:element minOccurs="0" name="customsCode" type="xs:string"/>
<xs:element minOccurs="0" name="docDate" type="xs:dateTime"/>
<xs:element minOccurs="0" name="docNum" type="xs:string"/>
<xs:element minOccurs="0" name="docTypeCode" type="xs:string"/>
<xs:element minOccurs="0" name="packageQty" type="xs:string"/>
<xs:element minOccurs="0" name="weight" type="xs:string"/>
</xs:sequence>
</xs:complexType>

```



```
<xs:complexType name="importTransportListResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputBase"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="openCycle">
<xs:sequence>
<xs:element minOccurs="0" name="inputCycle" type="tns:inputCycle"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="openCycleResponse">
<xs:sequence>
<xs:element minOccurs="0" name="return" type="tns:outputBase"/>
</xs:sequence>
</xs:complexType>
<xs:element name="Exception" type="tns:Exception"/>
<xs:complexType name="Exception">
<xs:sequence>
<xs:element minOccurs="0" name="message" type="xs:string"/>
</xs:sequence>
</xs:complexType>
</xs:schema>
```