

## PILOT ACTION FINAL REPORT - VERONA

---

DELIVERABLE D.T2.2.3

Version 1

03 2022

---



# Table of contents

- Table of contents .....0
- 1. Ex-ante situation .....1
- 2. Pilot action description.....4
- 3. Conclusions .....7
- 4. Annex I: SOAP Webservices types definitions .....9



## 1. Ex-ante situation

ZAILOG has taken part to European Union co-financed project for many years, with the aim of cooperating with other logistics nodes to improve the supply chain resilience and remove all major bottlenecks. One of the main areas of cooperation is the digitalization of the logistics processes, in order to prevent disruptions along the line and enhance the efficiency of the terminal operations.

Therefore, one of the main objectives of the Verona freight village is to foster the collaboration in the field of digitalization with all key ports and logistics players, at both national and European level, with the final goal to improve the interoperability among the IT systems of all public and private players along the supply chain.

Within this context of digital collaboration, great attention was given to the railway connection between the port of La Spezia and the Rail-Road terminal of Verona (Interporto Quadrante Europa, ZAILOG), which already features a service of 5 trains/week. Below, the table taken from the “Integrated planning of the railway shunting service in the port network of La Spezia and Marina di Carrara” details the connection between the two nodes.

Treno/Gr. carri	A/P	IF	MTO	Ora Arrivo/Partenza	Intervallo Orario	Provenienza/ Destinazione	L	Ma	Me	G	V	S	D	Giorni Soppressione	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

This existing rail connection could benefit from the implementation of ICT systems to enhance the efficiency of the service and contribute to create a seamless logistics chain on the SCANMED corridor. Towards this purpose, the main scope of the Pilot Action developed between La Spezia and Verona in the framework of the COMODALCE project is the data-sharing of messages containing more specific and reliable information of the train actual departure and expected arrival.

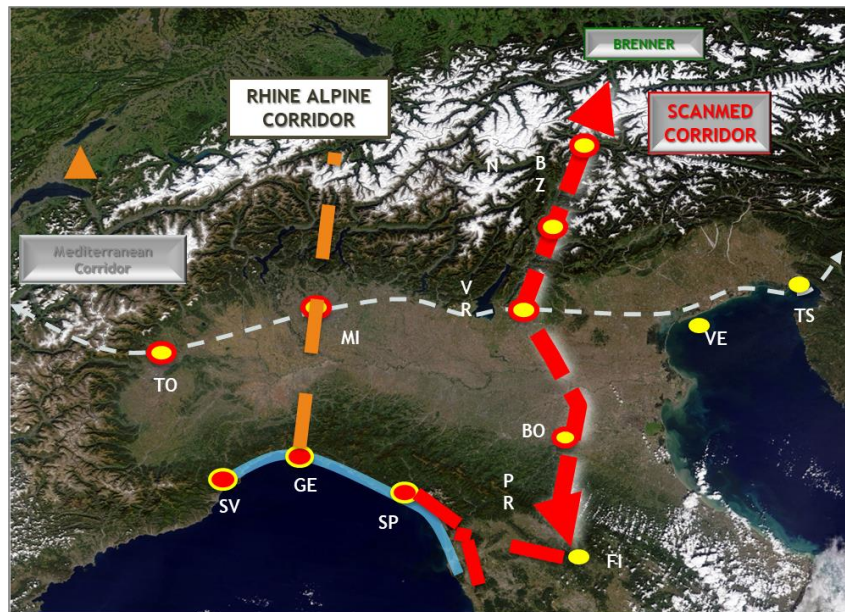


Figure 1 - La Spezia TEN-T connections

After a first stage of market study and analysis, the two partners (the Eastern Ligurian Sea Port Authority and ZAILOG) held several technical meetings with the potential players involved in the implementation of the Pilot Action, in order to define:

- the main features of the initiative
- the technical details and requirements
- the sub-activities required to carry out the project

Among the partners involved in the discussion, the main ones include:

- the technology provider;
- the Railway Undertaking;
- the Multimodal Transport Operator;
- the Port Terminal Manager;
- the Terminal Manager of the Verona Terminal.

The Port of La Spezia together with ZAILOG developed an ICT integrated platform to manage the rail freight connections between the two nodes. In this regard, it should be noted that the platform, namely the CMP - Corridor Management Platform, consists of a terrestrial extension of the ICT system already in place at the Port of La Spezia.

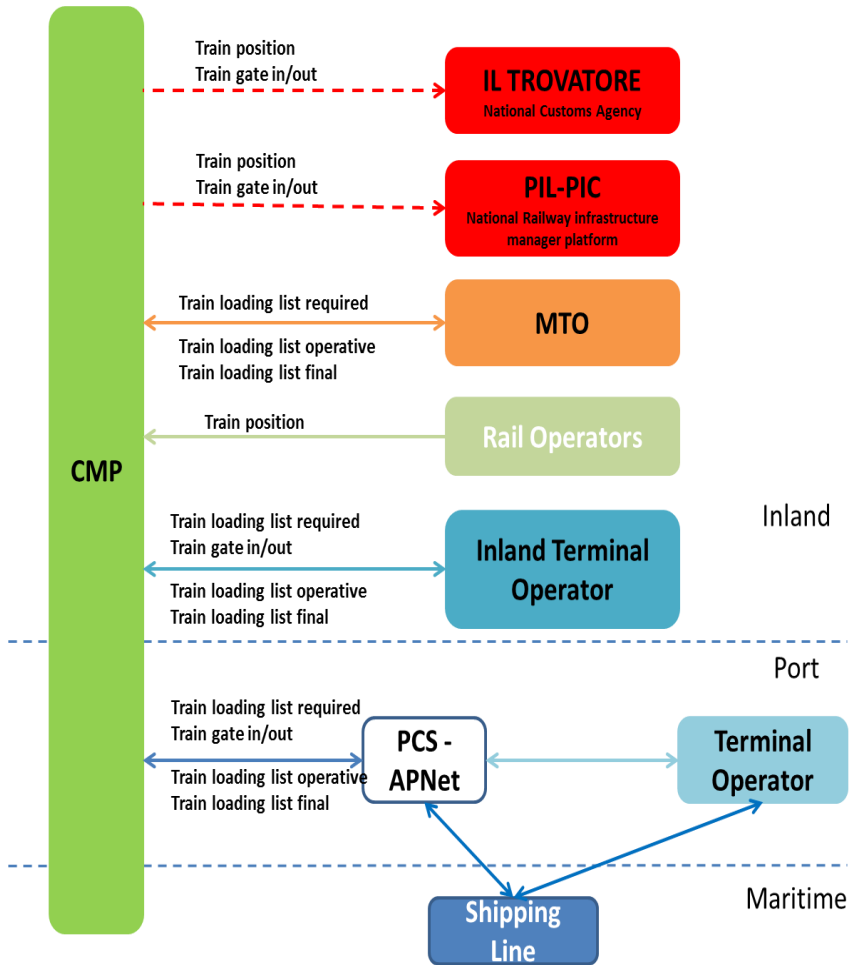


Figure 2 - Corridor Management Platform architecture



## 2. Pilot action description

The Pilot Action developed by a joint effort of La Spezia Port and Verona Terminal as part of the COMODALCE European Project is aimed at fostering the interoperability of the ICT systems among the supply chain players operating along the La Spezia - Verona nodes.

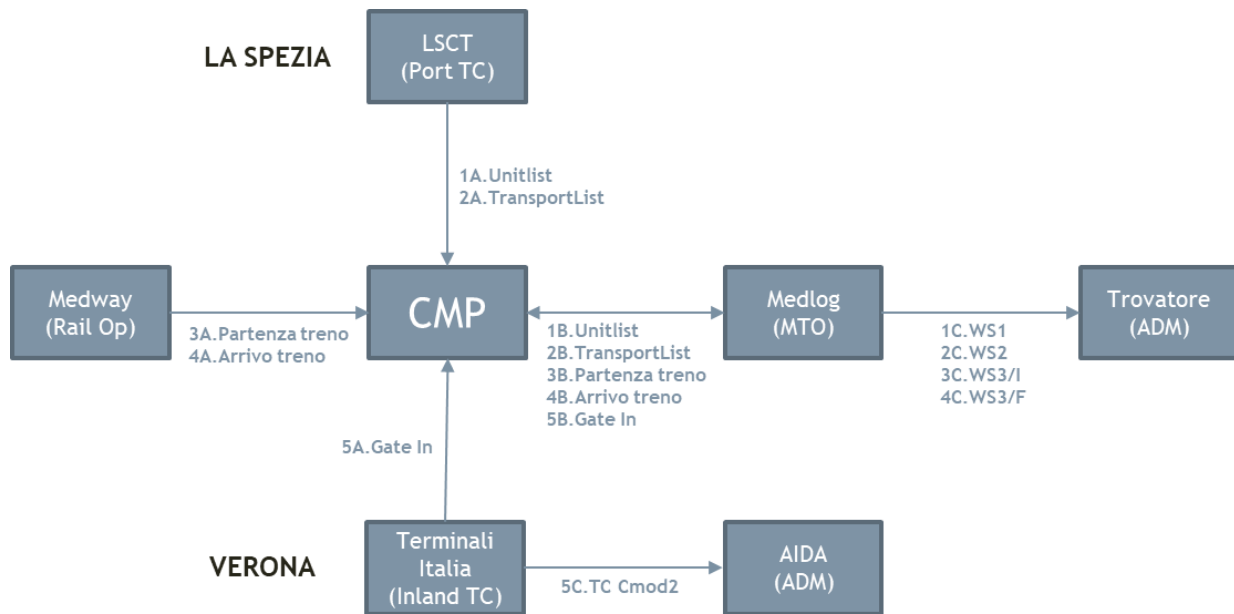


Figure 3 - Information flows managed by the COMODALCE Pilot Action

More in detail, the Pilot Action provides for the implementation of a data exchange system fully digitalized and automated, with the aim of ensuring a more efficient information exchange with evident benefits in the intermodal corridor management.

It should be noted that the present project falls within a wider masterplan defined by the Management Authority of Verona freight village, which is to strengthen the ICT system of the Rail-Road terminal for the benefit of all players based in the Rail-Road terminal area. The ultimate goal is to support with ICT tools and more detailed information the operational activities in the terminal, thus contributing to enhance the overall Verona node efficiency.

### Process features and technical details

The technological solution implemented provides for the exchange of data between the different operators involved. The technical specifications are explained in the paragraphs below.

First of all, it should be reminded that the main goal of the Pilot Action in the framework of the COMODALCE Project is to enhance the interoperability between Medlog (MTO), Medway (Railway Company), LSCT (Port Terminal Manager), and Terminali Italia (Inland Terminal Manager), in addition to the management platforms of the Managing Authorities of the Port of La Spezia and Verona freight village. In this regard, it is noteworthy to highlight that the CMP - Corridor Management Platform - will be interconnected with the two main management platforms of the Port of La Spezia, namely Il Trovatore and AIDA.



The data exchange process consists of the following main steps:

1. The railway Loading List is shared from the MTO to the CMP through the Unit List service; the CMP then forwards the Loading List to the Port Terminal manager. Simultaneously, the MTO then communicates the Loading List to the platform Il Trovatore through the WS1 Service, in order to enable the customs process of the ITUs in the Customs Corridor. A result of the process is also given.
2. The Port Terminal manager shares the Train Composition with the CMP through the Transport List Service; then, the CMP forwards the Train Composition to the MTO. At the same time, the Train Composition is also sent to Il Trovatore through the WS2 Service (Mission Registration), which returns a result.
3. The Train Departure is then communicated from the Railway Undertaking to the CMP, which then forwards the message to the MTO. Through WS3 Service then, the start of the Customs Mission procedures is communicated to Il Trovatore.
4. The Railway Undertaking informs the CMP of the Arrival Train, and the CMP then forwards the message to the MTO. Through WS3 Service, the end of the Customs Mission is transmitted to Il Trovatore.
5. The Inland Terminal manager communicates the Gate-In to the CMP, and the message is then forwarded to the MTO. Simultaneously, through the TC Mod2 Service the Inland Terminal manager communicates the message to AIDA.

On a more specific technical level, a SOAP Webservice provides for the implementation of the data exchange between the actors involved. For this purpose, both the “SendEvent” and the “TransportList” elements are used.

More specifically, on the one hand the “SendEvent” message connects an object with the event type, the location and the timestamp.

On the other hand, the TransportList contains a header and a list of details. The header includes information such as: the identification of the message, the message type, the identification of the sender and the message recipient, and also the message common data. Each of these elements refers to a specific object and contains a list of key-value pairs.

In so doing, the system implemented allows to manage the Customs Corridor in an automated and full digital way, thus contributing to enhance the interoperability among the supply chain players, to digitalize and standardize the data flows, to secure the information privacy, and to guarantee a full visibility and monitoring of the processes.



Registro estero

REGISTRO ESTERO

Benvenuto. Sede selezionata: TERMINALITALIA Logout

FILTRI

Partita da: [ ] a: [ ] Numero A3: [ ] Marca: [ ]

Data dal: [ ] al: [ ] Num Doc Scarico: [ ] Residui:

+ Crea

(1 OF 2)

		PARTITA	MRN	NUMERO A3	CIN A3	DATA A3	SINGOLO	SCADENZA A3	TIPO DOC PRECEDENT	DOCUMENT PRECEDENT	DATA DOC PRECEDENT	COLLI	PESO LORDO	MEZZO TRASPORTO	ORIGINE	POLIZZA	MERCE	MARCA	TIPOLOGIA TRASPORTO	PEZZI GIACENZA	PESO LORDO GIACENZA	DATA REGISTRAZ	NOTE
1		nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
1		nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
1		nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
1		nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
1		nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
1		nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
1		nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
1		nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		
1		nnn		nn		15/02/2022		16/05/2022				nn	100						terra	nn	100		

Figure 4 - Inland terminal operators Dashboard

The system implemented, namely the CMP (Corridor Management Platform), also provides for a monitoring dashboard, useful to monitor in real-time the status of customs temporary storage warehouses, as well as the status of the consignments. This information is also made available to the Multimodal Transport Operator, which can thus monitor the location of the loading units during the transport between the Port of La Spezia and Verona freight village, as well as their location when temporally stored in the warehouses.

In this regard, it should be noted that in order to guarantee a full visibility and control on the status of the loading units, the Dashboard provides several information, such as: customs status, operational status, shipping company, container number, weight, ship ETA, inland terminal, MMA number and date, and registration date.

Procedure di emergenza Mito e Customs

DASHBOARD

Benvenuto, support

FILTRI

Id Container: [ ] Numero A3: [ ] Emissione A3: [ ] Id Umit: [ ] Stato Container: In Arrivo In TC portuale In viaggio

Stato operativo: Passo Arrivi importato ETA-ATA PCS Passo Booking importato Refresh

Mese riferimento: Marzo Anno riferimento: dal 2022 Warning

Gestione Caricanti

(1 OF 1)

PRG INT	PRG DOC	STATO CTR	STATO OPERATIVO	TIPOLOGIA COORDINATA	COMPAGNIA DI NAVIGAZIONE	SIGLA CNTR	TIPO CNTR	LETTURA OCR	PORTO DI SBARCO	NOME NAVI	ETA	TERMINAL INTERMEDIO	DATA MMA	NUMERO MMA	CODICE UFFICIO EMISSIONE	DATA REGISTRAZ
2022/1		In TC Fuso	Dialogo AIDA C				2200		ITSPE		18/01/2019				068100	
2022/2		In TC Fuso	Dialogo AIDA C				2200		ITSPE		18/01/2019			1	068100	
2022/3		In TC Fuso	Dialogo AIDA C				2200		ITSPE		18/01/2019				068100	
2022/4		In TC Fuso	Dialogo AIDA C				2200		ITSPE		18/01/2019			1	068100	
2022/5		In TC Fuso	Dialogo AIDA C				2200		ITSPE		18/01/2019				068100	
2022/6		In TC Fuso	Dialogo AIDA C				2200		ITSPE		18/06/2019				068100	01/03/2022

Figure 5 - Multimodal Transport Operator Dashboard





### 3. Conclusions

The present Pilot Action within the framework of the COMODALCE project has led to successful outcomes in the logistics procedures, including:

- better efficiency and flexibility of the processes, quality of information, goods traceability, and reduction of administrative costs and attribution errors;
- more integrated data sharing between the different stakeholders of the entire logistics chain;
- enhanced security and reliability by offering standardized and qualified services;
- implementation of standard procedures for the information exchange which led to the interoperability of the systems;
- improved services for the benefit of the logistic corridor stakeholders, which has provided for the exploitation of economies of scale and more competition from an operating and economic point of view;
- reduction of the transport costs per loading unit;
- reduction of cargo handling time.

The following table lists the main benefits produced by the Pilot Action and recognizes a qualitative impact assessment for each of them:

<b>Benefits</b>	<b>Qualitative evaluation</b>
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 table above shows the improvements achieved by the implementation of the Pilot Action, with specific regards to:

- Improvement of the integration and data sharing among the actors of the supply chain;
- Improvement of efficiency and flexibility of logistics processes;
- Increase of security, reliability and transparency.

In conclusion, the Pilot Action has thus met the main objectives set at the beginning of the project, with specific regard to the cooperation among the partners for the digitalization of the logistics processes, with the aim of preventing disruptions along the supply chain and enhancing the efficiency of the terminal operations.

In fact, in the framework of the railway services between the Port of La Spezia and Verona freight village, the present Pilot Action has successfully served the purpose of enhancing the efficiency of the connection between the two nodes and contributing to create a seamless logistics chain on the SCANMED corridor.



## 4. Annex I: SOAP Webservices types definitions

An example of the messages definition is reported below.

```

<xs:schema          attributeFormDefault="unqualified"          elementFormDefault="unqualified"
targetNamespace="http://webservice.tt.cap.it/"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://webservice.tt.cap.it/"
<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 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="BPROCESSIIId" 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="BEVENTIId" type="xs:int"/>
<xs:element minOccurs="0" name="codEvento" type="xs:string"/>
<xs:element name="createBUTENTIId" 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="updateBUTENTIId" 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="updateBUTENTIId" 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="bPROCESSIId" type="xs:int"/>
<xs:element minOccurs="0" name="bSEDIId" type="xs:int"/>
<xs:element minOccurs="0" name="bUTENTIId" 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>
```