Cyprus Import Control System (ICS)

Phase 1

SYSTEM ARCHITECTURE DOCUMENT

(SAD) - B2B Specification

Version 1

Department of
Information Technology
Services

Table of Contents

TABLE OF CONTENTS 2
LIST OF FIGURES 3
ACRONYMS / ABBREVIATIONS 4
1. INTRODUCTION 5
2. COMMUNICATION BASICS 6
3. SEQUENCE DIAGRAMS 10
4. XSDS FOR MESSAGE EXCHANGE 18
4.1 IcsRequest Message 18
4.2 IcsResponse Message........ 21
4.3 Ics Web Service Definintion 24

LIST OF FIGURES

Figure 1: IcsRequest Message Structure 7
Figure 2: IcsResponse Message Structure 9
Figure 3: Submission of an 315 Message 11
Figure 4: Submission of a 313 message 12
Figure 5: Submission of a 323 Message 13
Figure 6: Receive of a 351 message 14
Figure 7: Receive of a 328 message 15
Figure 8: Request for retrieving message by Date Range 16
Figure 9: Request for retrieving messages for a specific MRN 17

ACRONYMS / ABBREVIATIONS

Acronym/Abbreviations	Meaning
B2B	An electronic transaction between two organisations
DDNIA	Design Document for National Import Application
ENS	Entry summary declarations
FISS	Functional Import System Specifications
EC	European Communities
ICS	Import Control System
EO	Economic Operator
EU	European Union
ICT	Information and Communications Technologies
ISO	International Organisation for Standardisation
TBD	To be defined
UML	Unified Modelling Language
WP	Work Package
WWW	World Wide Web
XLS	Microsoft Excel file
XML	Extensible Markup Language
XSD	XML Schema Definition
SOAP	Simple Object Access Protocol
WSDL	An XML-based language that provides a model for describing Web services.

1. INTRODUCTION

The following paragraphs contain a description of the B2B communication between Traders applications and the ICS system. Its intended audience are developers of third party applications, aiming to communicate with the ICS System.

The document assumes that the reader is already familiar with the EU ICS specifications while good knowledge of FTSS and DDNIA is assumed.

2. COMMUNICATION BASICS

The B2B communication with the ICS System will take place through a unique Web Service that will act as a proxy to internal services (not exposed to traders) that serve a specific request.

Authentication between the Trader's Application and the ICS will be based on the WS-Security specification. The Trader's Application needs to authenticate itself to the ICS System through WS-Security information (Username, Password). This piece of information should be included in the header of the SOAP message. WS-Security offers a general-purpose mechanism, the 'UsernameToken Profile' which describes how a Web service client (Trader's Application) can supply a UsernameToken (Trader Username and Password) as a way to identify the requestor.

Communication between the Traders Application and ICS is always initiated by the former and by invoking the proxy Web Service and receiving the relevant response. The invocation of the Web Service and the corresponding response will be implemented based on the following two XML messages:

IcsRequest : Used for invoking the proxy Web Service (Figure 1)

IcsResponse: Used for propagating the Web Service results back to the caller (Figure 2)

The above, generic messages act as wrappers to actual XML messages defined and described by the EU ICS specification. In addition they contain certain metadata that enable the proxy Web Service to execute the internal service that materialises the request.

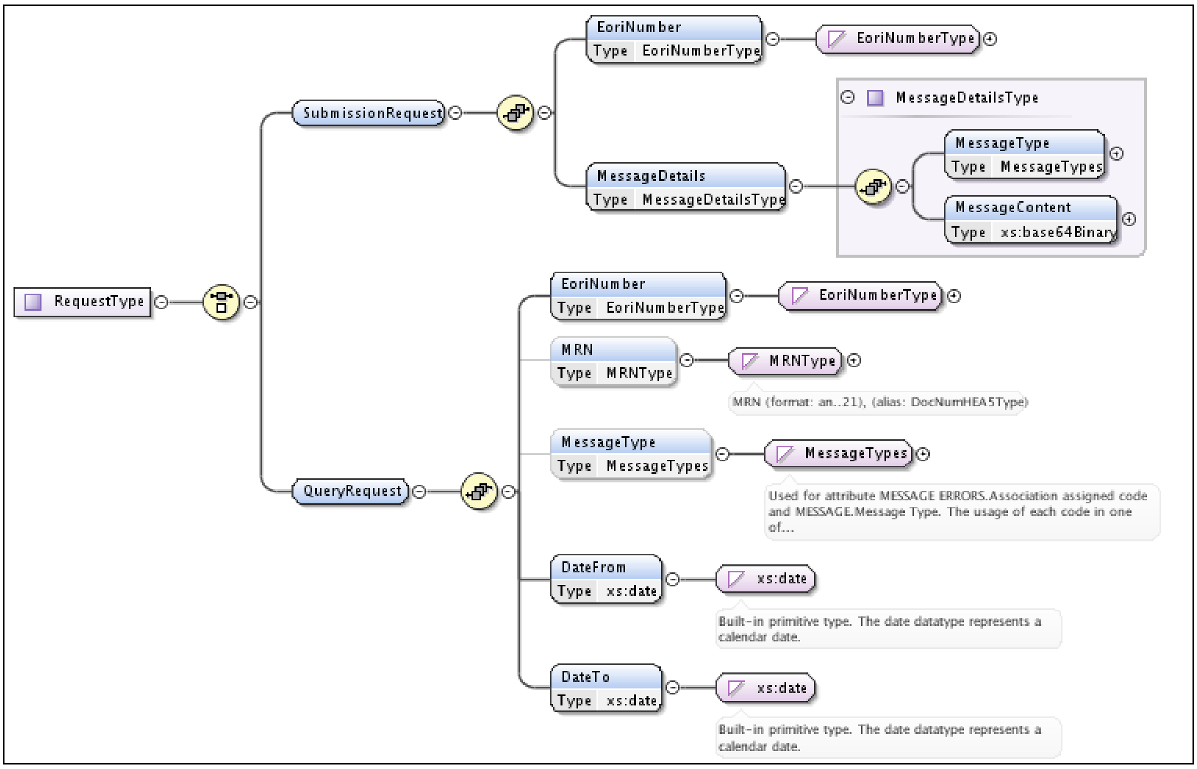

Figure 1: IcsRequest Message Structure

As already mentioned, invoking the proxy ICS Web Service implies sending an IcsRequest message.

The role of the IcsRequest message is dual. It can be used either for submitting an XML message according to the EU Specifications or for querying the ICS system for messages received and are addressed to a particular trader.

In this context, each *IcsRequest* can contain either a *SubmissionRequest* or a *QueryRequest*. For example, when the user wants to submit an ICS message (e.g. CC315A, CC323A etc.), then the *IcsRequest* should contain the element *SubmissionRequest*, which consists of the user's *EoriNumber* and the *MessageDetails* (*MessageType* and *MessageContent*). The *MessageContent* is the actual xml message (as defined by the EU ICS specification), initially zipped (compressed) and then encoded in Base64 format.

When the user wants to retrieve messages from the ICS System then the *IcsRequest* should contain the element *QueryRequest*. Within the QueryRequest element the trader defines the criteria for retrieving specific messages.

The available criteria are one or a combination of the following:

- *MessageType*

- a date range (*DateFrom, DateTo)*
- and optionally an *MRN*

For example, in order for the Trader's application to check if there are any messages for the Trader Today, the date range criteria can be used.

Alternatively all the messages for a particular message type e.g. CC351A can be retrieved or all the messages related to a particular MRN.

For each request (Web Service Invocation), the ICS System will respond with an *IcsResponse* message. The structure of this message can be found in Figure 2.

The content of the *IcsResponse* message depends on the content of the corresponding *IcsRequest* message.

If the request message contained a *SubmissionRequest* element, then the response will contain a *SubmissionResponse* element. Accordingly, if the request message was a *QueryRequest*, then the response will contain a *QueryResponse* element.

The *SubmissionResponse* element consists of the user's *EoriNumber*, the *SubmissionResult* and optionally a standard ICS Message (*MessageType, MessageContent* zipped and encoded in base64 format) and according the EU ICS specification.

The *QueryResponse* element consists of the user's *EoriNumber*, the *QueryResult* and optionally as many ICS Messages as the query will return.

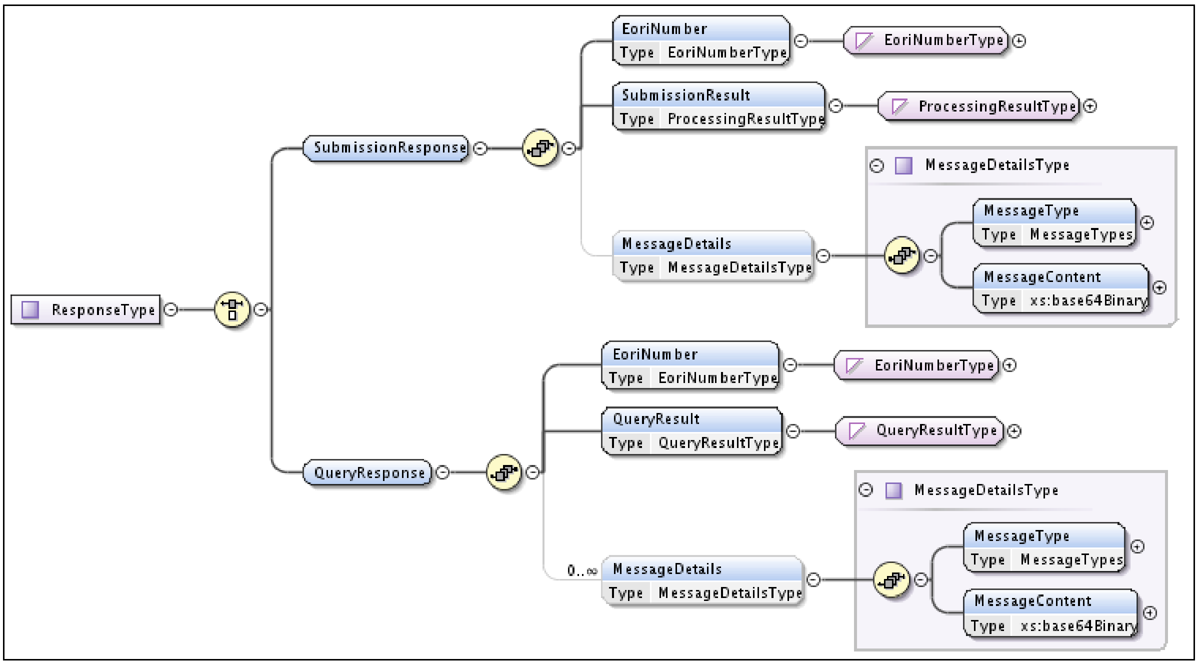

Figure 2: IcsResponse Message Structure

3. SEQUENCE DIAGRAMS

The following sequence diagrams depict the interaction between the Trader's application and the ICS System.

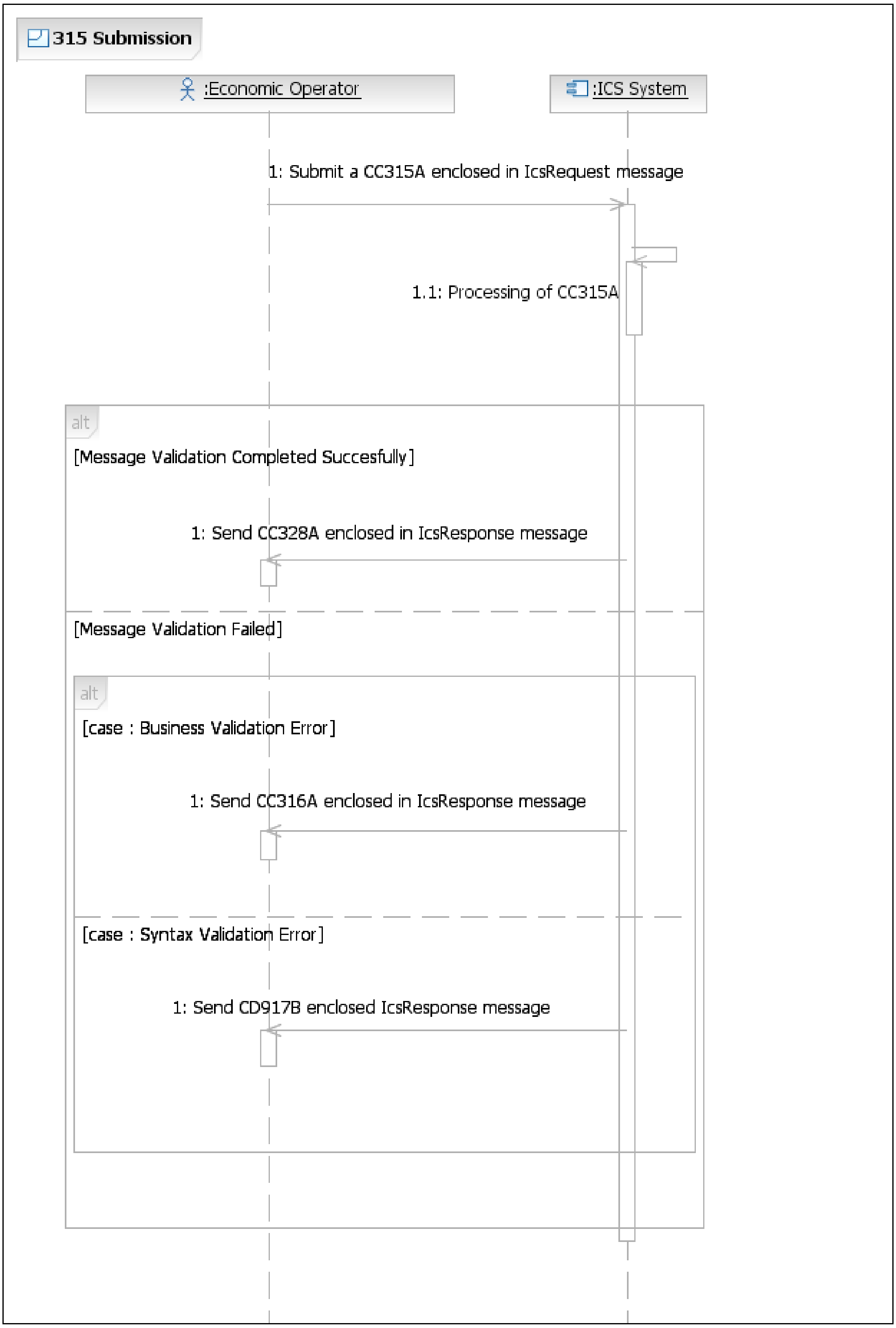
315 Submission
:Economic Operator
:ICS System
1: Submit a CC315A enclosed in IcsRequest message
1.1: Processing of CC315A
alt
[Message Validation Completed Succesfully]
1: Send CC328A enclosed in IcsResponse message
[Message Validation Failed]
alt
[case : Business Validation Error]
1: Send CC316A enclosed in IcsResponse message
[case : Syntax Validation Error]
1: Send CD917B enclosed IcsResponse message

Figure 3: Submission of an 315 Message

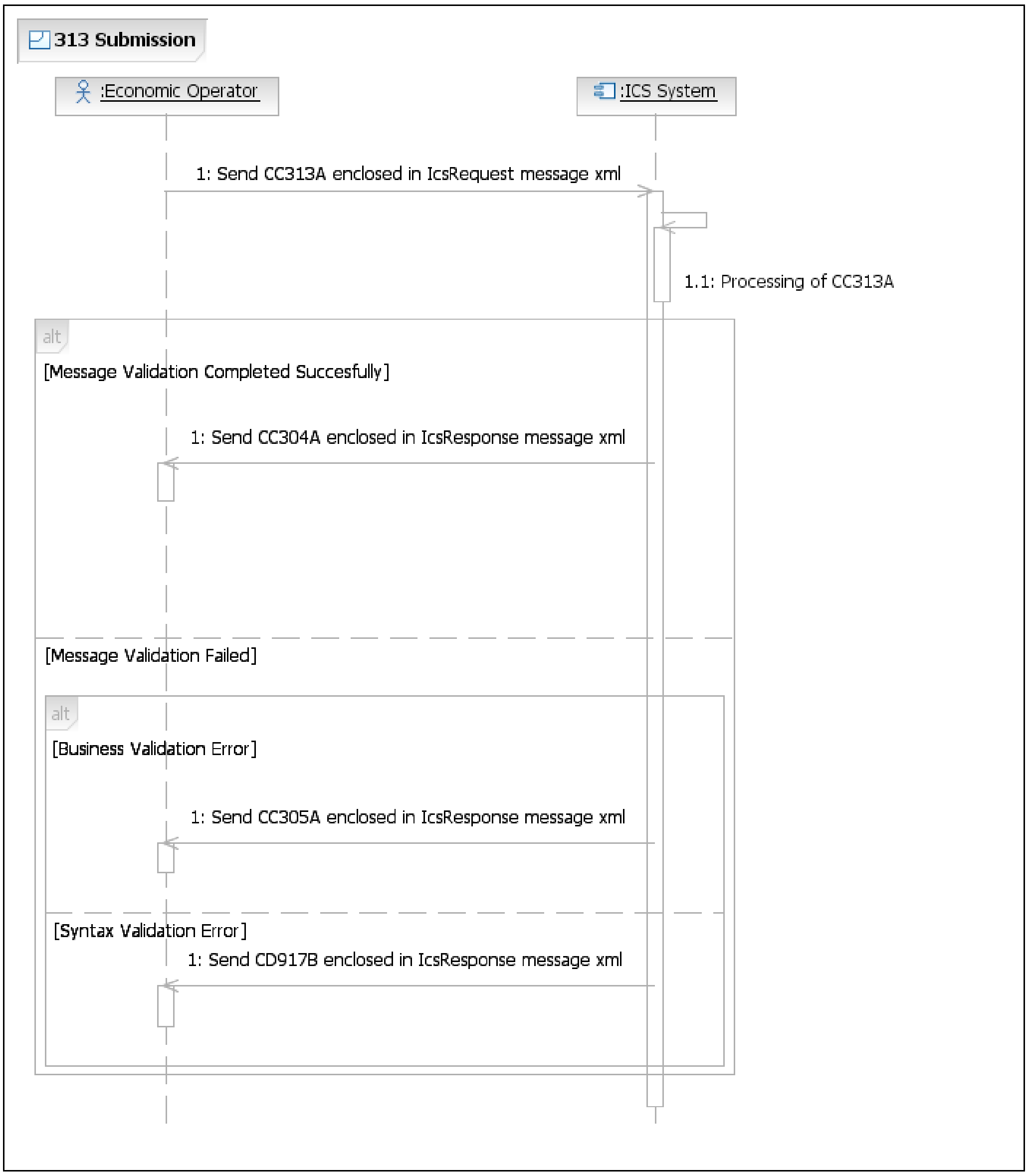
313 Submission
:Economic Operator
:ICS System
1: Send CC313A enclosed in IcsRequest message xml
1.1: Processing of CC313A
alt
[Message Validation Completed Succesfully]
1: Send CC304A enclosed in IcsResponse message xml
[Message Validation Failed]
alt
[Business Validation Error]
1: Send CC305A enclosed in IcsResponse message xml
[Syntax Validation Error]
1: Send CD917B enclosed in IcsResponse message xml

Figure 4: Submission of a 313 message

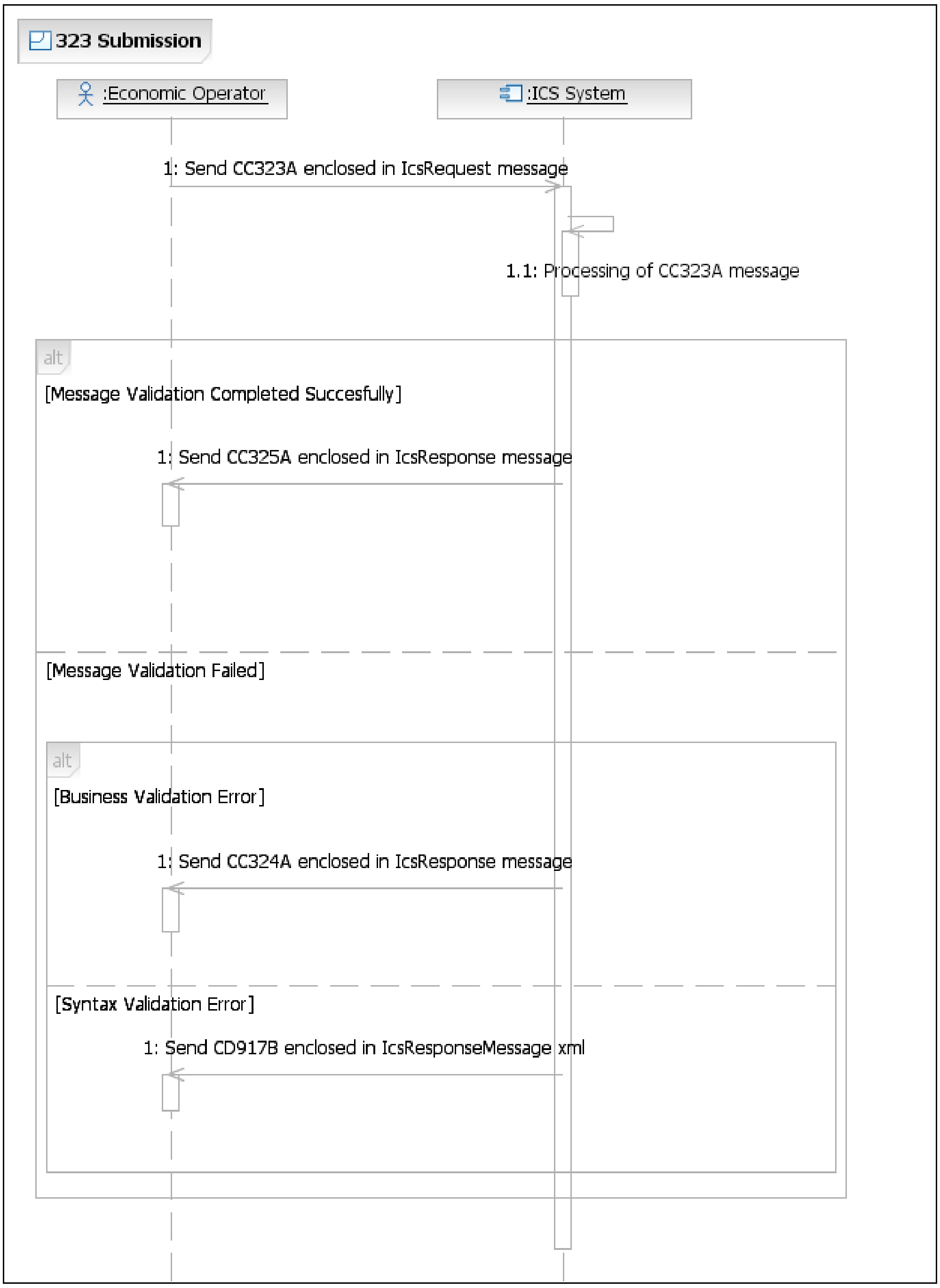
323 Submission
:Economic Operator
:ICS System
1: Send CC323A enclosed in IcsRequest message
1.1: Processing of CC323A message
alt
[Message Validation Completed Succesfully]
1: Send CC325A enclosed in IcsResponse message
[Message Validation Failed]
alt
[Business Validation Error]
1: Send CC324A enclosed in IcsResponse message
[Syntax Validation Error]
1: Send CD917B enclosed in IcsResponseMessage xml

Figure 5: Submission of a 323 Message

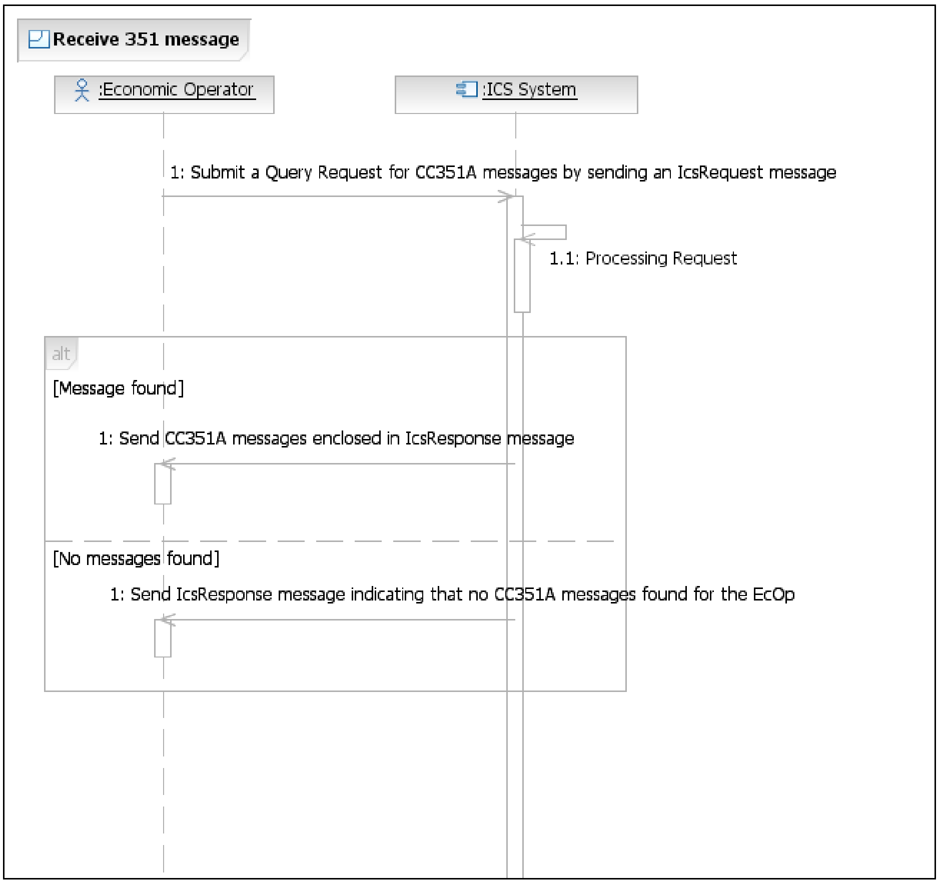
Receive 351 message
:Economic Operator
:ICS System
1: Submit a Query Request for CC351A messages by sending an IcsRequest message
1.1: Processing Request
alt
[Message found]
1: Send CC351A messages enclosed in IcsResponse message
[No messages found]
1: Send IcsResponse message indicating that no CC351A messages found for the EcOp

Figure 6: Receive of a 351 message

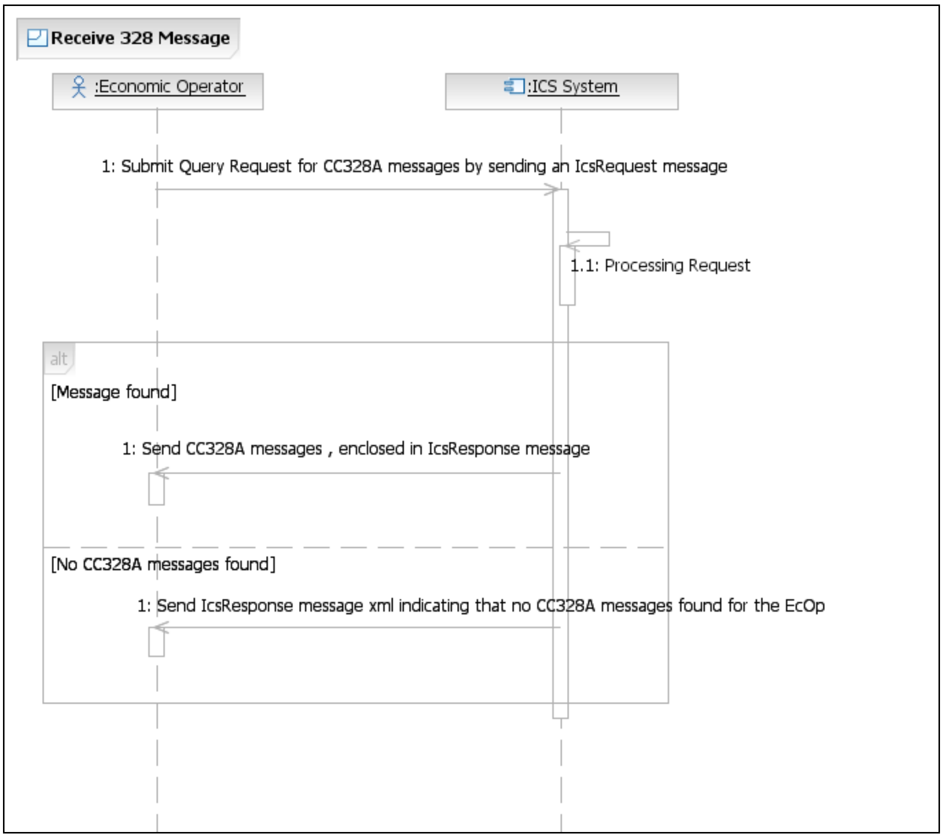
Receive 328 Message
:Economic Operator
:ICS System
1: Submit Query Request for CC328A messages by sending an IcsRequest message
1.1: Processing Request
alt
[Message found]
1: Send CC328A messages , enclosed in IcsResponse message
[No CC328A messages found]
1: Send IcsResponse message xml indicating that no CC328A messages found for the EcOp

Figure 7: Receive of a 328 message

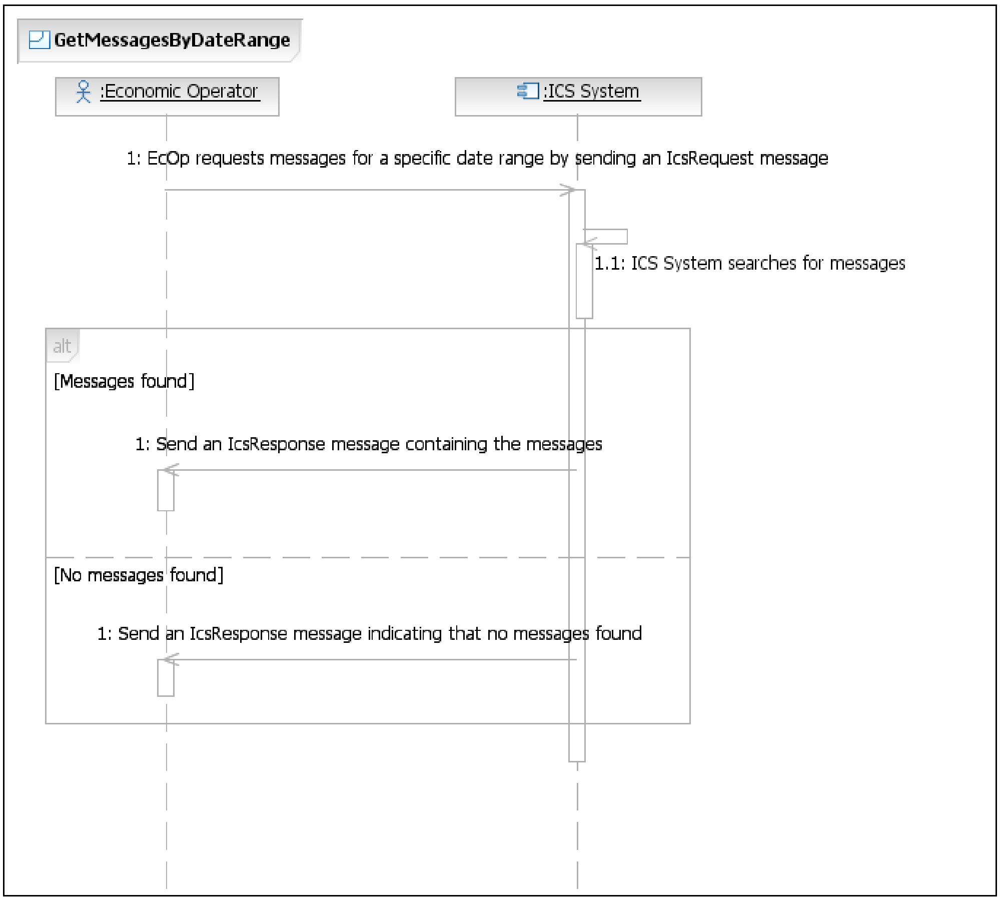
GetMessagesByDateRange
:Economic Operator
:ICS System
1: EcOp requests messages for a specific date range by sending an IcsRequest message
1.1: ICS System searches for messages
alt
[Messages found]
1: Send an IcsResponse message containing the messages
[No messages found]
1: Send an IcsResponse message indicating that no messages found

Figure 8: Request for retrieving message by Date Range

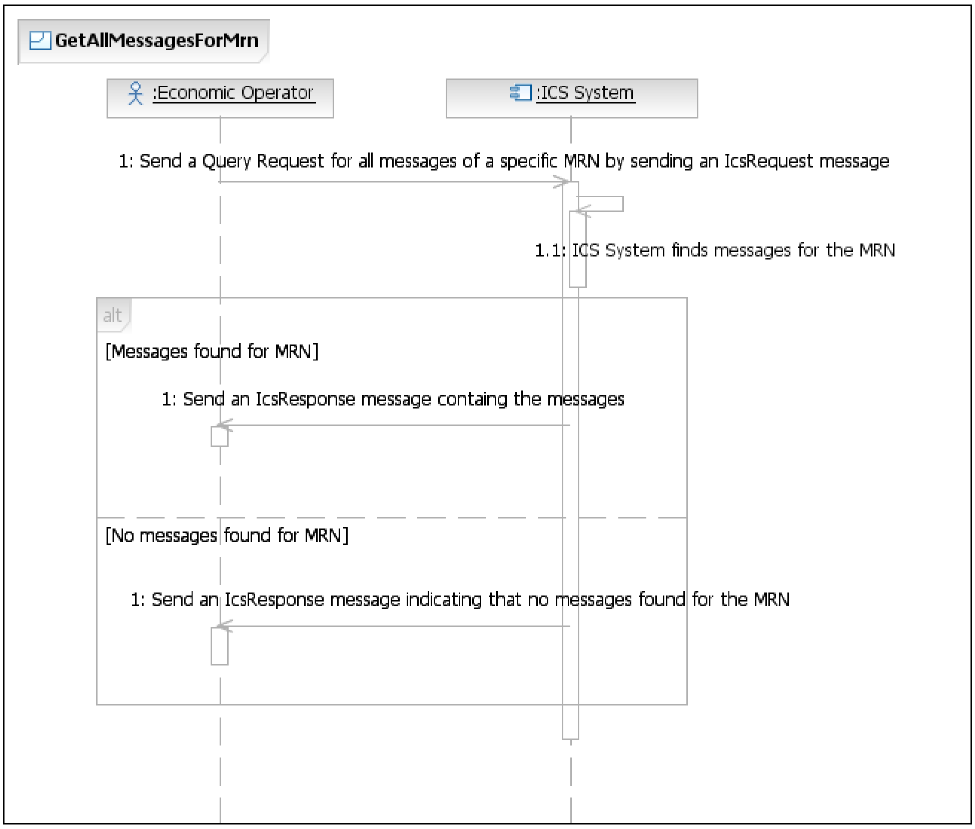
GetAllMessagesForMrn
:Economic Operator
:ICS System
1: Send a Query Request for all messages of a specific MRN by sending an IcsRequest message
1.1: ICS System finds messages for the MRN
alt
[Messages found for MRN]
1: Send an IcsResponse message containg the messages
[No messages found for MRN]
1: Send an IcsResponse message indicating that no messages found for the MRN

Figure 9: Request for retrieving messages for a specific MRN

4. XSDs FOR MESSAGE EXCHANGE

4.1 IcsRequest Message

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://eurodyn.com/ws/IcsRequest"
   xmlns="http://eurodyn.com/ws/IcsRequest">
  <xs:element name="Request" type="RequestType"/>
  <xs:complexType name="RequestType">
    <xs:sequence>
      <xs:element name="SubmissionRequest" type="Submission" minOccurs="0"
maxOccurs="1"/>
      <xs:element name="QueryRequest" type="Query" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="MessageDetailsType">
    <xs:sequence>
      <xs:element name="MessageType" type="MessageTypes"/>
      <xs:element name="MessageContent" type="xs:base64Binary"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="EoriNumberType">
    <xs:restriction base="StringLatin1">
      <xs:minLength value="3"/>
      <xs:maxLength value="17"/>
      <xs:pattern value="[A-Z]{2}[^\n\r]*"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="StringLatin1">
    <xs:restriction base="xs:string">
      <xs:pattern value="\p{IsBasicLatin}+"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="MRNType">
    <xs:annotation>
      <xs:documentation>MRN (format: an..21), (alias:
        DocNumHEA5Type)</xs:documentation>
    </xs:annotation>
    <xs:restriction base="AlphanumericCapitalType">
      <xs:pattern value="[0-9]{2}[A-Z]{2}[A-Z0-9]{13}[0-9]"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="AlphanumericCapitalType">
    <xs:restriction base="AlphaNumType">
      <xs:pattern value="[A-Z0-9]*"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="AlphaNumType">
    <xs:annotation>
      <xs:documentation>Base class for all anN and an..N
        types</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:token"/>
  </xs:simpleType>
  <xs:simpleType name="MessageTypes">
    <xs:annotation>
      <xs:documentation>Used for attribute MESSAGE ERRORS.Association assigned code and
MESSAGE.Message Type. The usage of each code in one of the UNSM templates is defined in the
Remarks column. CC should in all cases be instantiated to the value of the particular domain in
which the message is created. [Format: an..6]</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:token">
      <xs:enumeration value="CC304A">
        <xs:annotation>
          <xs:documentation>Entry Summary Declaration Amendment
Accepted</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC305A">
        <xs:annotation>
          <xs:documentation>Entry Summary Declaration Amendment
Rejection</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC313A">
        <xs:annotation>
          <xs:documentation>Entry Summary Declaration Amendment</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC315A">
        <xs:annotation>
          <xs:documentation>Entry Summary Declaration</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC316A">
        <xs:annotation>
          <xs:documentation>Entry Summary Declaration Rejected</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC323A">
        <xs:annotation>
          <xs:documentation>Diversion Request Import</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC324A">
        <xs:annotation>
          <xs:documentation>Diversion Request Rejected</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC325A">
        <xs:annotation>
          <xs:documentation>Diversion Request Acknowledgement</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC328A">
        <xs:annotation>
          <xs:documentation>Entry Summary Request Acknowledgement</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC351A">
        <xs:annotation>
          <xs:documentation>Advanced Intervention Notification</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CD906B">
        <xs:annotation>
          <xs:documentation>Functional NACK</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CD917B">
        <xs:annotation>
          <xs:documentation>XML NACK</xs:documentation>
        </xs:annotation>
```

```
      </xs:enumeration>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="Submission">
    <xs:sequence>
      <xs:element name="EoriNumber" type="EoriNumberType" maxOccurs="1" minOccurs="1"/>
      <xs:element name="MessageDetails" type="MessageDetailsType" maxOccurs="1"
minOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="Query">
    <xs:sequence>
      <xs:element name="EoriNumber" type="EoriNumberType" maxOccurs="1" minOccurs="1"/>
      <xs:element name="MRN" nillable="false" type="MRNType" minOccurs="0"/>
      <xs:element name="MessageType" nillable="false" type="MessageTypes" minOccurs="0"/>
      <xs:element name="DateFrom" nillable="false" type="xs:date"/>
      <xs:element name="DateTo" nillable="false" type="xs:date"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

4.2 IcsResponse Message

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="http://eurodyn.com/ws/IcsResponse"
   targetNamespace="http://eurodyn.com/ws/IcsResponse">
  <xs:element name="Response" type="ResponseType"/>
  <xs:complexType name="ResponseType">
    <xs:sequence>
      <xs:element name="SubmissionResponse" type="SubmissionRes" minOccurs="0"
maxOccurs="1"/>
      <xs:element name="QueryResponse" type="QueryRes" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="ProcessingResultType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="PROCESS OK"/>
      <xs:enumeration value="PROCESS FAILED"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="QueryResultType">
    <xs:restriction base="xs:string">
```

```
      <xs:enumeration value="QUERY OK"/>
      <xs:enumeration value="NO RESULTS"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="MessageDetailsType">
    <xs:sequence>
      <xs:element name="MessageType" type="MessageTypes"/>
      <xs:element name="MessageContent" type="xs:base64Binary"/>
    </xs:sequence>
  </xs:complexType>
  <xs:simpleType name="EoriNumberType">
    <xs:restriction base="StringLatin1">
      <xs:minLength value="3"/>
      <xs:maxLength value="17"/>
      <xs:pattern value="[A-Z]{2}[^\n\r]*"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="StringLatin1">
    <xs:restriction base="xs:string">
      <xs:pattern value="\p{IsBasicLatin}+"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="MessageTypes">
    <xs:annotation>
      <xs:documentation>Response Message Types</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:token">
      <xs:enumeration value="CC304A">
        <xs:annotation>
          <xs:documentation>Entry Summary Declaration Amendment
Accepted</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC305A">
        <xs:annotation>
          <xs:documentation>Entry Summary Declaration Amendment
Rejection</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC313A">
        <xs:annotation>
          <xs:documentation>Entry Summary Declaration Amendment</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC315A">
        <xs:annotation>
          <xs:documentation>Entry Summary Declaration</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC316A">
        <xs:annotation>
          <xs:documentation>Entry Summary Declaration Rejected</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC323A">
        <xs:annotation>
          <xs:documentation>Diversion Request Import</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC324A">
        <xs:annotation>
          <xs:documentation>Diversion Request Rejected</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC325A">
        <xs:annotation>
          <xs:documentation>Diversion Request Acknowledgement</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC328A">
        <xs:annotation>
          <xs:documentation>Entry Summary Request Acknowledgement</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CC351A">
        <xs:annotation>
          <xs:documentation>Advanced Intervention Notification</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CD906B">
        <xs:annotation>
          <xs:documentation>Functional NACK</xs:documentation>
```

```
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="CD917B">
        <xs:annotation>
          <xs:documentation>XML NACK</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
      <xs:enumeration value="ALL">
        <xs:annotation>
          <xs:documentation>Query for All Messages</xs:documentation>
        </xs:annotation>
      </xs:enumeration>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="SubmissionRes">
    <xs:sequence>
      <xs:element name="EoriNumber" type="EoriNumberType" maxOccurs="1" minOccurs="1"/>
      <xs:element name="SubmissionResult" type="ProcessingResultType"/>
      <xs:element name="MessageDetails" type="MessageDetailsType" maxOccurs="1"
minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="QueryRes">
    <xs:sequence>
      <xs:element name="EoriNumber" type="EoriNumberType" maxOccurs="1" minOccurs="1"/>
      <xs:element name="QueryResult" type="QueryResultType"/>
      <xs:element name="MessageDetails" type="MessageDetailsType" maxOccurs="unbounded"
minOccurs="0"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

4.3 Ics Web Service Definintion

```
<definitions
   targetNamespace="http://eurodyn.com/IcsService"
   xmlns="http://schemas.xmlsoap.org/wsdl/"
   xmlns:tns="http://eurodyn.com/IcsService"
   xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
   xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
   xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
   xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
   xmlns:req="http://eurodyn.com/ws/IcsRequest"
   xmlns:res="http://eurodyn.com/ws/IcsResponse"
 >
   <types>
     <schema xmlns="http://www.w3.org/2001/XMLSchema">
       <import namespace="http://eurodyn.com/ws/IcsRequest"
schemaLocation="IcsRequestMessage.xsd"/>
     </schema>
     <schema xmlns="http://www.w3.org/2001/XMLSchema">
       <import namespace="http://eurodyn.com/ws/IcsResponse"
schemaLocation="IcsResponseMessage.xsd"/>
     </schema>
   </types>
   <message name="IcsRequestMessageType">
     <part name="parameters" element="req:Request"/>
   </message>
   <message name="IcsResponseMessageType">
     <part name="parameters" element="res:Response"/>
   </message>
   <portType name="IcsService">
     <operation name="process">
       <input message="tns:IcsRequestMessageType"/>
       <output message="tns:IcsResponseMessageType"/>
     </operation>
   </portType>
   <binding name="IcsService" type="tns:IcsService">
     <soap12:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
     <operation name="process">
       <soap12:operation soapAction="process" soapActionRequired="false"/>
       <input>
         <soap12:body use="literal"/>
       </input>
       <output>
         <soap12:body use="literal"/>
       </output>
     </operation>
   </binding>
   <service name="IcsService">
```

```
      <port name="IcsService" binding="tns:IcsService">
        <soap12:address location="http://127.0.0.1:8888/ics-ws/IcsService"/>
      </port>
    </service>
  </definitions>
```

END OF DOCUMENT