Australian Government

Department of Agriculture, Fisheries and Forestry

NEXDOC

VIP04 – Software Developers Guide

Document Properties

Document version	3.0
Document status	**ISSUED**
Issue date	
Print date	
Author	Department of Agriculture, Fisheries and Forestry

Revision History

Date	Version	Description
01/08/2017	0.1	Draft
23/02/2018	1.0	Issued
24/04/2018	1.1	Updates to software registration section, addition of data change rules, addition of remote print disclaimer and insertion of sample requests.
17/07/2018	1.2	Minor updates to wording. Updates to WSDL references to include new fields. Updated type element for length example Inserted new section for reference data types
11/09/2018	1.3	Enforces commodity as mandatory Introduces the new declaration section Three additional fields added to submission service Updated samples for order and lodge based on new schema requirements
29/10/2018	1.4	Added commodity type to the endorsements interface Add additional fields for submission and read services for Dairy Add new service for request assistance Add new operation to the Read service for Read EDN
16/04/2019	1.5	Added new Transfer EDN service Added new REST API for retrieving full or partial codesets as CSV
8/05/2019	1.6	Added extra details around EU declaration codes and added Production urls.
17/06/2019	1.7	Removed additional EU declaration code text – the text will now be in the ECM_DECLARATION codeset. Updated the section on acknowledging warnings to provide examples when the error is a SOAP fault.
24/09/2019	1.8	Added clarification around Tokens required for Client and Exporter services
22/01/2020	1.9	Corrected page numbering and the rest api url. Re-embedded the soap request and response xmls into the document. Removed duplicate contents. Added urls of the respective services used in NEXDOC. Added notification sample messages. Updated the state diagram for no permit rex.
03/02/2020	2.0	Merged two versions of this document.
12/02/2020	2.1	Added "removeExistingSet" to imaDetails and messageAcknowledgements. Added missing elements to imaDetails, and moved the element to its correct place.
30/03/2020	2.2	Added the custom service to the document that details transfer, cancel and read rex edn.
15/09/2020	2.3	Added Cancel REX and Release REX to print services. Added more information regarding the Amend Request function. Updated references of HCRD to CTRD to reflect new status code.
16/09/2020	2.4	Updated the ReadRex response to return any certificate template details for a given product line that were not explicitly supplied by the exporter.

Date	Version	Description
06/11/2020	2.5	Updated references to AUSKEY to now reference myGovID. Updates to have Reissue Reason mandatory (RexCertificateService).
09/02/2021	2.6	Added section about AEPI authorisation and updated RexOwnership to clarify between transfer and forwards.
25/02/2021	2.7	Removed sample XML request/responses. For sample request/responses please refer to the test conformance suite now.
22/06/2022	2.8	Updated the changes for Paperless Certification for EU Traces
08/06/2023	2.9	Updated the Change Permissions Matrix for Eggs
05/09/2023	3.0	Added Composite Permission Matrix for Eggs and Dairy. Updated Farm fields for Eggs commodity.

Abbreviations and Acronyms

The following abbreviations and acronyms are used throughout this document.

Acronym	Description
CN	Combined Nomenclature
EDI	Electronic Data Interchange
EDN	Export Declaration Number
EU	European Union
ICS	Integrated Cargo System
NEXDOC	Export Document application. This system supersedes the EXDOC application. NEXDOC's core capability allows clients to search for a case, evaluate a case to identify specific import conditions, generate a view of the import conditions and export the information into a document.
REX	Request For Export
SLO	Software Liaison Officer
SOAP	Simple Object Access Protocol.
SSL	Secure Sockets Layer.
TRACES NT	TRACES New Technology is the European Commission's multilingual online sanitary and phytosanitary certification platform
WSDL	Web Services Description Language.
XML	Extensible Markup Language.
XMLDSIG	An XML Signature used for signing and verification.
XSD	XML Schema Definition.

Contents

1   Introduction

1.1   Overview

This document provides details of the Next Export Document (NEXDOC) external web services interfaces that enables third party software developers to create and maintain data communications with the NEXDOC application.

1.2   Services Overview and Standards Interface Testing

To allow software developers to test their interfaces the Department has established a Vendor testing environment. This environment will allow software developers to communicate with the NEXDOC web services and ensure that their applications are transmitting acceptable data before integration with the Production web services.

The services within this Vendor test environment will perform the same functionality as the Production web services. Access to the Vendor and Production environments is restricted. To gain access to the Production environment users will need to register with the Department using a myGovID via the Department's Client Portal. To gain access to lower environments Vendors will need to contact the Department's Software Liaison Officer.

1.3   Maintenance of Interfaces

It is the Department's intention to minimise all NEXDOC interface changes to allow for the ongoing operation of Vendor software. From time to time however it is expected that new business requirements will mandate an update to the NEXDOC interfaces. During such changes it is possible that new interfaces will be added or existing interfaces may change resulting in changes to their associated validation rules, the introduction of new validation rules or the deletion of current validation rules.

During these changes developers may find that their particular application requires further development to remain compliant with the NEXDOC interfaces. It is in the best interest of all software developers to design their applications with the flexibility to allow for interface evolution.

The Department will allow an adequate period of time for software developers to amend their software in response to interface changes. All software developers who wish to continue to utilise the NEXDOC interfaces may need to make the necessary changes to their application to comply with these validation rule changes and may be required to redeploy their applications.

1.4   Disclaimer

Any person who chooses to develop or use an application that communicates with NEXDOC does so on the basis that they acknowledge:

- schemas and validation rules and any other aspect of NEXDOC may be changed, and NEXDOC can be suspended, superseded, replaced, or withdrawn, at any time and for any reason;

- the Department has no involvement or responsibility of any kind in relation to such applications including, without limitation, in relation to their development, testing, promotion, support, training, use (including end user support, training and use) or any other matter relating to any such application (other than as a receiver of XML data);

- to the extent permitted by law the Department makes no representation, and provides no warranty, of any kind (whether express or implied) in respect to NEXDOC, or any such application, or any related matter, without limitation including in respect to condition, quality, performance, merchantability, durability or fitness for purpose; and

- the development or use of any such application is entirely at the developer's or user's own cost and risk and without any liability (whether direct, indirect or consequential, no matter how arising, including whether on the basis of contract, statute, promissory estoppel, quantum meruit or other quasi contractual or restitutionary grounds, in negligence or any other grounds whatsoever) on the part of Department or any other person.

It is mandatory that each Software Vendor using the NEXDOC web services display and enforce acknowledgement of the above disclaimer to all users of their application.

1.5   Remote Print Disclaimer

The following disclaimer must be shown to all users when attempting to remote print a document provided by NEXDOC.

To be able to print export documents at my own site, I agree to the following:

- The print site is located in Australia

- I have a software package to communicate with NEXDOC that has been registered by the Department

- I have a duplex laser printer which supports at least 600 dpi

- I will only print one set of export documents for each consignment

- I will not alter, vary or in any way change an export document generated in NEXDOC once it has been printed

- I understand that there are a number of criminal offences under the Criminal Code Act 1995 (Cth) relating to the making and use of a false Commonwealth document.

- The Department reserves the right to at any time suspend, cancel or amend the terms of this print agreement. Inability to demonstrate valid and legitimate use of the printed export document(s) may result in the Department revoking the option to print NEXDOC generated document(s) through the remote print function.

2   NEXDOC Overview

NEXDOC's core capability allows clients to submit Requests for Export to receive the required export documents that allow for the export of product of biosecurity concerns from Australia, or to allow entry of Australia's exported product into foreign countries.

To allow for business to government integration NEXDOC has exposed a number of web service end points to replicate this process. These ends points allow for the:

- Retrieval of available code set definitions to complete the Request for Export process;

- Submission and retrieval of Requests for Export;

- Forwards and transfer of ownership of Requests for Export;

- Retrieval of certificates for "In-Office" printing where appropriate.

- Retrieval and acknowledgment of NEXDOC related notifications.

The NEXDOC web services also provide a number of supporting functions which will be explained further in this document.

2.1   Software Vendor Registration

All Software Vendors who wish to integrate with the NEXDOC web services will be required to register as a Software Vendor via the Department's Client Portal. This can be accessed at https://online.agriculture.gov.au/portal.

In order to complete the registration process Software Vendors will be required to register as a myGovID client first. Information on how to obtain a myGovID can be obtained from the Australian Government website https://www.mygovid.gov.au/.

2.2   Software Registration

Prior to Software Vendors being able to interact with the web services in Production they will be required to test their software in the Vendor environment. During the testing process the Software Liaison Officer (SLO) will be available to provide assistance with any connectivity issues. Once Vendors believe they have successfully completed their testing, they will contact the SLO and advise that they have finalised their testing and would like to be registered. The SLO will then start the internal registration process and advise the Vendor if they will be registered or what further action is required to be registered.

Once registered the Vendor will then be able to generate Software Vendor tokens that allow integration with the Production web services. The SLO will also provide a set of tokens that will allow Vendors to access the end user training environment.

Vendors may at any time commence registration in the Production environment using their myGovID however their registration will not be finalised until approved by the SLO.

The document *NEXDOC VIP01 – Registration Processes* provides more information on the various pathways for the registration of Software Vendors and their software.

2.3   Software Developer Package

To assist software developers in their activities the Department will provide a Software Developer Package. This package will provide the Vendor with the required service endpoints, a compressed file version of the WSDL's and a number of tokens that will enable integration with the web services as well as a number of applicable Exporter Identifiers that will allow end to end testing of their software.

To obtain the Software Developer Package please contact the Department Software Liaison Officer using the following e-mail address: EnhancedTraceabilityProject@agriculture.gov.au

2.4   Services Overview and Standards

This information is provided to ensure that third parties can correctly extract the required information from the NEXDOC system and pass correctly structure message back.

Please note all XML shown within this document is provided for example purposes only and all software developers must download the currently released version from the web services via the WSDL.

2.5   Process Flow

The below diagram provides a suggested process flow for interaction with the NEXDOC services in a request that results in a remote ("In-Office") print outcome. The process flow is broken out into the relevant party's area of responsibility for one possible implementation of the NEXDOC Interface. Each Software Vendor would need to evaluate these steps to determine if they suit their needs.
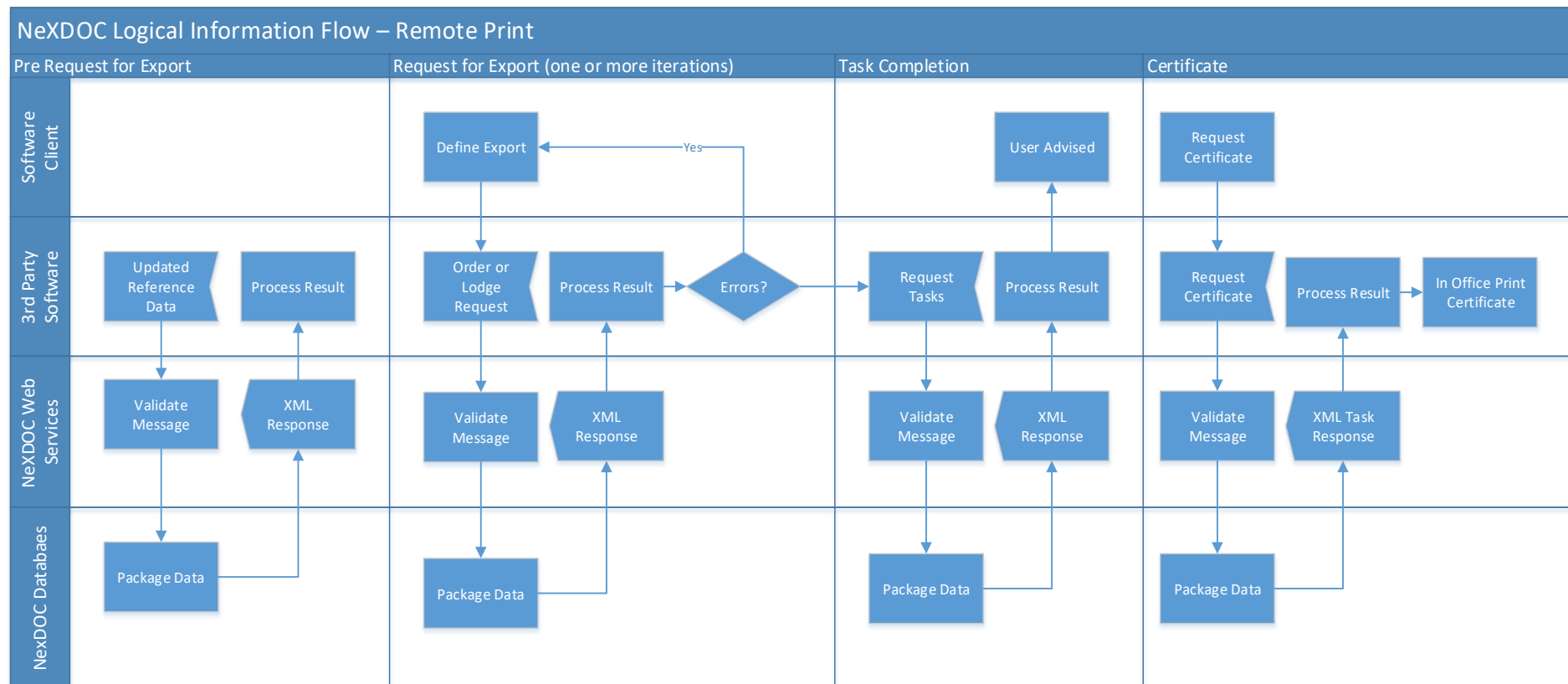
Figure 1 NEXDOC Interface process flow

3   Message Exchange

This section describes the implementation details that apply to every service and service operation and their use within all web service operations.

3.1   SOAP over HTTPS

All NEXDOC web services utilise the Simple Object Access Protocol (SOAP) and NEXDOC will support SOAP 1.1. Services that involve sending or receiving files will use SOAP Message Transmission Optimization Mechanism (MTOM). SOAP HTTP Binding is also utilised whilst SSL provides transport layer encryption of messages.

The Production url for the SOAP gateway is:

https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/(Port Type Name)

3.2   WSDL File

A Web Service Description Language (WSDL) is used to define the web service contracts and can be used by many development tools to generate code for connecting to and invoking the web services. All messages sent to and received from NEXDOC must comply with the WSDL.

All software developers wishing to use the NEXDOC web services will have access to the WSDL generated by the services by using the URL's provided in the Software Development Package.

Alternatively developers may extract the WSDL files provided in the software developers package which will also displayed for some XSD's additional documentation to assist with development. A sample documentation tag is show below.

```
<xsd:enumeration value="I">
    <xsd:annotation>
        <xsd:documentation>INFORMATION</xsd:documentation>
    </xsd:annotation>
</xsd:enumeration>
```

3.3   Authentication Rules

All interactions with the Department's web services will require an Installation Token (username) and Installation Password (password digest). This will be used to authenticate the credentials of the request against the NEXDOC installations register. Failure to include the installation token and password within the header will result in a rejection of the message.

To create an Installation Token users will need to be registered with the Department via the Client Portal and have requested the Installation Administrator role. It is envisaged that either Software Vendors or their clients may possibly be installation administrators depending on the chosen software installation.

An example of the Installation Token for the SOAP Headers is provide below.

```
<soapenv:Header>
    <wsse:Security>
        <wsse:UsernameToken wsu:Id="UsernameToken-1">
            <wsse:Username>test_ws</wsse:Username>
            <wsse:Password>G2CIbJ8XHn1B/6trq7VoHyDbX24=</wsse:Password>
```

```
        </wsse:UsernameToken>
      </wsse:Security>
    </soapenv:Header>
```

3.4   SOAP Faults

Common errors encountered during the actioning of a web service request will be returned to the caller as a SOAP fault. A custom SOAP fault 'NexdocSoapFault' has been defined to accommodate multiple errors to be reported for a given SOAP request. These faults will be embedded in the fault element of the response data contract for the web service method.

When an error occurs as a result of an internal NEXDOC system problem the fault code provided in the SOAP fault will indicate "InternalServiceFault" with a unique reference number. In these circumstances Vendor software can resend the same request at a later time. If the problem still persists then the Vendors can report the issue to the department with the reference number for investigation and resolution.

An example of such a fault with error reference number is

```
    <env:Body xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
      <soap-env:Fault xmlns:soap-env="http://schemas.xmlsoap.org/soap/envelope/">
        <faultcode>soap-env:Server</faultcode>
        <faultstring>Unexpected error. Error reference: 1cd9b030-29d1-4cb1-a27f-d57cc6e9d817</faultstring>
        <faultactor>online-dev.agriculture.gov.au:443/nexdoc-ws-ext-proxy/EndorsementService</faultactor>
        <detail/>
      </soap-env:Fault>
    </env:Body>
```

When business rule validations are being performed on an incoming message any non-compliance with the business rules will be reported via the above mentioned 'NexdocSoapFault'. The Vendor software should not attempt to resend the same message without firstly correcting the issues identified.

NEXDOC will provide additional information related to the SOAP fault in the detail element of the SOAP fault. For faults, NEXDOC will provide details of the specific business rule check that was violated in the detail element.

An example Fault is shown below.

```
    <ns0:Fault>
      <faultcode>ns0:Server</faultcode>
      <faultstring>VALIDATION FAIL</faultstring>
      <detail>
        <ns0:NexdocSoapFault>
          <ns0:FaultTimestamp>2017-08-22T15:19:43.575+10:00</ns0:FaultTimestamp>
          <ns0:NexdocSoapFaultItem>
            <ns0:FaultCode>100</ns0:FaultCode>
            <ns0:FaultReason>ERROR<ns0:FaultReason>
```

```
            <ns0:FaultMessage>The line items certificate templates are not valid.
    </ns0:FaultMessage>
            </ns0:NexdocSoapFaultItem>
        </ns0:NexdocSoapFault>
      </detail>
    </ns0:Fault>
```

3.5   Request-Response Message Exchange

All NEXDOC web services use the **Request-Response message exchange pattern**. Each operation has a specific schema for the request message and a schema for the corresponding response message returned upon successful completion of the web service operation.

All NEXDOC web service operations occur synchronously and provide an immediate response. In many cases, all NEXDOC activity required for the transaction invoked by the request message is completed before the response message is returned. Some actions requires additional activity to be performed which is performed asynchronously. The additional activity will be queued and the Vendor software package will have to intermediately poll the notification list to check for completion.

3.6   NEXDOC Schemas

A number of NEXDOC XML Schemas have been created to enable the transmission of data requests from external users to NEXDOC. These schemas are all generated from a common conceptual data model and form part of the each service WSDL. They are located within the <wsdl:types> section.

3.6.1   How do I extract the schemas?

Locate the <wsdl:types> section of the XML file by opening the WSDL in a web browser. Once complete XML similar to the below will be displayed.

```
<types>
….<xsd:schema>
……..<xsd:import schemaLocation="<<address>>/NEXDOC-ws/GenericCodeSetPort?xsd=1"
namespace="<<address>>//common/CommonTypes_1.0"/>
….</xsd:schema>
</types>
```

To utilise this information copy and paste the value in the schemaLocation attribute into your web browser and you will be presented with the Schema.

3.6.2   What is in the schemas?

Each schema contains a definition of all the data elements and their validation rules that make up a NEXDOC request. These rules include the type of data allowed to be entered, the range of acceptable values, the relationship between elements and whether each element is optional or mandatory. An example of this detail is shown below:

```
<xsd:element name="FaultCode" type="xsd:string" minOccurs="0"/>
```

3.6.3   What is not in the schemas?

Some business rules have been defined that must be enforced by the developer when capturing or submitting data. These rules arise from the need to cross validate the data against other data elements, or where a value is mandatory based on the inclusion of another value. To allow for the effective transmission of data, these business rules must form part of the Software Vendor validation process.

3.6.4   How to use the schemas

All Vendor software developers can choose how to design and develop their application to implement the required schemas and their validation rules.

Each schema (XSD file) contains all the data elements that make up a NEXDOC Request. The following attributes may exist on any element and need to be validated using the XSD. This will ensure that all data entered into the XML will meet the validation rules determined by the external interface into NEXDOC.

3.6.4.1 Element Attributes

Each type will have one or more element which must be used to validate the data being captured. An example element is detailed below.

```
<xsd:complexType name="ConsigneeType">
….<xsd:sequence>
……..<xsd:element name="consigneeName" type="xsd:string"/>
……..<xsd:element name="consigneeAddress" type="ns0:AddressType"/>
….</xsd:sequence>
</xsd:complexType>
```

3.6.4.2 Name

This is the name of the data field being entered. The name will be used to map the data captured to the required validation rules and business rules. An example name is detailed below.

```
name="ElementType"
```

3.6.4.3 Type

This is the type of data to be entered for this element. The type below shows an enumeration of ElementType which are defined in the schema.

```
type="tns:ElementType"
```

An example of the definition of the ElementType is detailed below:

```
<xsd:simpleType name="TelephoneType">
….<xsd:restriction base="xsd:string">
……..<xsd:enumeration value="W"/>
……..<xsd: enumeration value="M"/>
……..<xsd: enumeration value="F"/>
……..<xsd: enumeration value="P"/>
….</xsd:restriction>
```

</xsd:simpleType>

Within the XSD provided with the developer package additional details may also be provided as per the below

<xsd:simpleType name="NoticeType">

….<xsd:restriction base="xsd:token">

……..<xsd:enumeration value="I">

………..<xsd:annotation>

………….<xsd:documentation>INFORMATION</xsd:documentation>

………..</xsd:annotation>

……..</xsd:enumeration>

……..<xsd:enumeration value="W">

………..<xsd:annotation>

………….<xsd:documentation>WARNING</xsd:documentation>

………..</xsd:annotation>

……..</xsd:enumeration>

……..<xsd:enumeration value="E">

……..<xsd:annotation>

………….<xsd:documentation>ERROR</xsd:documentation>

……..</xsd:annotation>

….</xsd:enumeration>

</xsd:restriction>

The type field can also contain length restriction information as highlighted below where a restriction of seven characters is enforced.

type="ct:AlphaNumeric**7Type**"

3.6.4.4 MinOccurs

This is the minimum number of occurrences of this element in the log file. If this value is 1, then this is a mandatory field and must be entered at all times. By default, this value may not appear as an attribute in the element. If this is the case, then the default value is 1, making it a mandatory field. An example MinOccurs is detailed below

<xs:element minOccurs="1" maxOccurs="unbounded" name="MatchTypes" type="ns0:CountryGroupSpecies"/>

3.6.4.5 MaxOccurs

This is the maximum number of occurrences of the element that this element can occur in the log file. If this value is 1, then it can only occur once. If it is unbounded it can occur as many times as needed. An example MaxOccurs is detailed below

<xs:element minOccurs="0" maxOccurs="unbounded" name="MatchTypes" type=" ns0:CountryGroupSpecies"/>

3.6.4.6 Restrictions (Ranges)

The minInclusive and maxInclusive attributes control the range of values that can be entered into the element. If a value falls outside of this range then it will fail the schema validation process. An example restrictions is detailed below

```
<xs:restriction base="xs:int">

  <xs:minInclusive value="0"/>

  <xs:maxInclusive value="500000"/>

</xs:restriction>
```

3.6.4.7 Nillable

Types defined with nillable="true", allow for a null value to be explicitly defined for the supporting element type. For example

```
<xsd:element name="departureDate" type="xsd:date" nillable="true" minOccurs="0" />
```

May have a null value applied via the element data:

```
<com: departureDate xsi:nil="true"/>
```

This is commonly used to remove an existing value via a subsequent an amend service call.

3.7   Web Services

All Software Vendor applications will need to communicate with the NEXDOC Web Services developed for processing known XML schemas. There are several web services available that Vendor application can communicate with. The communication between the Vendor applications and NEXDOC will be over a secure sockets layer (SSL) and will be encrypted using the Department's digital certificate.

The testing (Test) site will allow for all of the same functionality as the Production (Prod) site and is a separate environment.

3.8   Validating Data against the Schema

Once the data has been captured in the Vendor application, it must be placed into an XML file. The schema can then be used to validate the XML file to ensure that the XSD rules have been satisfied before transmission to NEXDOC.

When the data is received by NEXDOC, it will be validated against the published schemas which reflect the current version of the NEXDOC web services to ensure that the data received is valid and that the schema used by the Vendor application is still correct.

3.9   Returned XML String

On successful transmission NEXDOC will return a valid response. An example of a returned XML string is provided below:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">

  <S:Body>

    <ns1:GenericCodeSetResponse
xmlns:ns1="http://agriculture.gov.au/NEXDOC/GenericCodeSetSoap_1.0">

      <ns1:genericCodeSetResponse>
```

```xml
<ns1:codeSetName>ECM_TREATMENT</ns1:codeSetName>
<ns1:listCodeSet>
    <ns1:listPosition>1</ns1:listPosition>
    <ns1:listItemCodeSet>
        <ns1:key>CODE</ns1:key>
        <ns1:value>DIPPED</ns1:value>
        <ns1:type>string</ns1:type>
    </ns1:listItemCodeSet>
    </ns1:listItemCodeSet>
        <ns1:key>DESCRIPTION</ns1:key>
        <ns1:value>DIPPED COLD</ns1:value>
        <ns1:type>string</ns1:type>
    </ns1:listItemCodeSet>
    </ns1:listItemCodeSet>
        <ns1:key>START_DATE</ns1:key>
        <ns1:value>2017-07-13T12:03:39.677</ns1:value>
        <ns1:type>dateTime</ns1:type>
    </ns1:listItemCodeSet>
    </ns1:listItemCodeSet>
        <ns1:key>END_DATE</ns1:key>
        <ns1:value>null</ns1:value>
        <ns1:type>dateTime</ns1:type>
    </ns1:listItemCodeSet>
    </ns1:listItemCodeSet>
        <ns1:key>UPDATED_DATE</ns1:key>
        <ns1:value>2017-07-13T12:03:39.677</ns1:value>
        <ns1:type>dateTime</ns1:type>
    </ns1:listItemCodeSet>
    </ns1:listCodeSet>
    </ns1:genericCodeSetResponse>
    </ns1:GenericCodeSetResponse>
    </S:Body>
</S:Envelope>
```

3.10   Validation Rules

3.10.1 Why validation rules are required

The NEXDOC XML packages contain data that must meet various validation rules to ensure consistency and accuracy of data being transmitted. Vendor software developers should ensure they address all validation rules defined within the schemas to ensure that all data, when tested against the NEXDOC services, will pass the required validation rules. This process will ensure successful data transmission to NEXDOC and save transmission costs due to the transmission of XML files that contain data errors.

3.10.2 When to apply the validation rules

It is up to the software developer to determine how and when the validation rules will be applied within their application. This may depend on the application being developed and the way the developer intends the user to enter data. At a minimum it is a requirement that all validation rules are applied before the data is transmitted to NEXDOC.

Once the data is received by the NEXDOC services any XML data that fails validation will be rejected by the service, and a fault message, if applicable, returned to the sender.

3.10.3 High Level Validation Rules

The following are the high level validation rules that have been defined for interaction with the NEXDOC web services.

3.10.3.1   XML File

The application must generate an XML file that contains the required data as validated by the schema.

3.10.3.2   XSD Schema

The application must use the appropriate schema to validate the data being captured and generated as an XML file. The version of the schema must be an active schema and contain no modifications to the schema supplied by the Department.

3.10.4 Enumerations Data

The allowable values for elements that are confined by specific discrete values are defined within the schema as enumerations.

These enumerations must be used to validate any data element defined within the schema as using a specific type. The value in the type attribute will match to a type element in the XML schema, and the developers must validate this data against the matching XML element.

For example, the below XML shows the type **TelephoneType** to be defined as a set of four possible string values:

```
<xsd:simpleType name="TelephoneType">
….<xsd:restriction base="xsd:string">
….…..<xsd:enumeration value="W"/>
….…..<xsd:enumeration value="M"/>
….…..<xsd:enumeration value="F"/>
```

```
........<xsd:enumeration value="P"/>
….</xsd:restriction>
</xsd:simpleType>
```

The element **TelephoneType** is defined as using the above type

```
<xsd:complexType name="TelephoneNumberType">
….<xsd:simpleContent>
........<xsd:extension base="xsd:string">
…….……<xsd:attribute name="type" type="ns1:TelephoneType"/>
........</xsd:extension>
….</xsd:simpleContent>
</xsd:complexType>
```

This method ensures that only the allowable values can be entered. If a user enters data that is not one of the four options, then the schema validation will fail and the Software Vendor application should not transmit the XML to NEXDOC for processing.
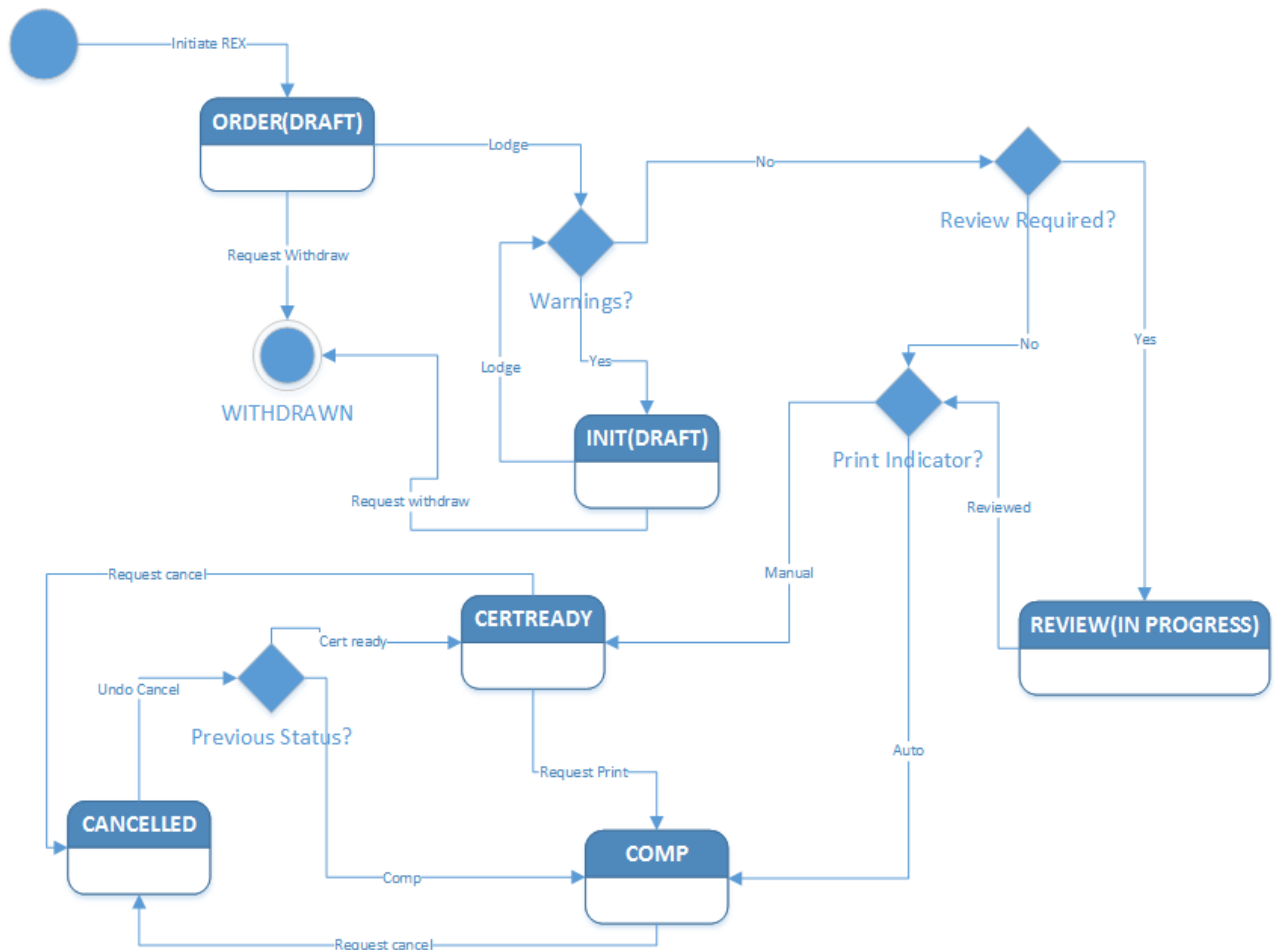
4 Interface Capability and Logical Information Flows

This section defines the interface capabilities and logical information flows between Vendor software applications and NEXDOC.
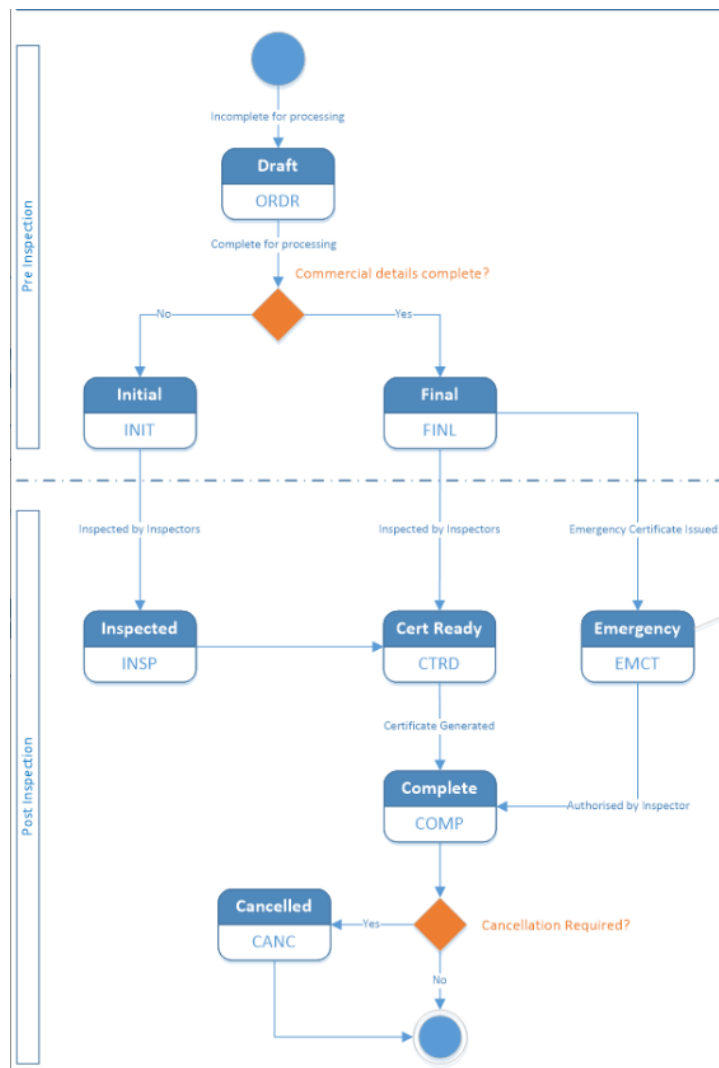
4.1 Logical Flow

Provided below are the logical flows for scenarios where a permit is and is not required.

4.1.1 Permit Not Required

4.1.2 Permit Required

5   Supporting Services

This section details the services that are required in support of submitting a request for export.

5.1   Reference Data

The reference data service provides Vendor software with a generic interface for retrieving NEXDOC reference data.

SOAP web service

The request has one mandatory field "codeSetName" that must match an entry from the "CodeSetName" enumeration.To get the different values please refer to EnumTypes.xsd in the sdl package provided. If a developer only passes the codeSetName in the request the response will retrieve everything from NEXDOC for the supplied code set.

When using the reference data service it is recommend that the entire list for a specific code set is rarely downloaded and instead the developer implements message exchange to use the last updated filter. This field is updated at creation of the item and on each subsequent update. Using this method ensures that only the items that have changed sense the last retrieve are returned. The required date format for these field are "yyyy-MM-ddTHH:mm:ss.S" for example "**2016-11-17T23:59:59.250**".

The reference data service also accepts other optional values that can be used to further filter search results. These include "description" and "code". All fields are case insensitive and the NEXDOC response is sorted ascending by the returned code value.

The url is https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ReferenceDataService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

REST web service

Full or partial code sets can be download as a CSV file using this service. For reference data, this is the preferred over the SOAP web service, specially for large code sets.

URL

GET https://online.agriculture.gov.au/nexdoc-rs/api/public/v1/reference-data/{codesetName}[?lastUpdated=2019-01-01T00:00:00]

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

Parameters

- codesetName
  - o Mandatory
  - o Style: template
  - o Type: xsd:string
- lastUpdated
  - o Optional
  - o Style: query

- o   Type: xsd:string

- o   Format: ISO 8601

List of available code sets

Refer to Reference Data Codesets for a list code sets.

5.2   Client

The client service provides Vendor software clients with the ability to request a new client identifier, request access to client groups or retrieve their client token and client group details for use within Vendor software.

Once successfully submitted the service will return the client token to the user. A further notification will be sent to all client group administrators of the organisation to approve access to the required client groups. The outcome of this request is notified back to the client via the e-mail address supplied in the initial request.

The URL is https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/PartyService?WSDL

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

5.3   Exporter Details

The exporter details service provides software clients with the ability to find the exporter identifier associated with an organisation. The service accepts a legacy exporter identifier (ExDOC exporter number) and returns the new identifier and the exporter address details. If a new exporter identifier is supplier it will return the details of the exporter to confirm the exporter's details are as expected.

The URL is https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/PartyService?WSDL

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

5.4   Notifications

A number of NEXDOC web services may result in events that involve human interactions or other actions which cannot always occur within the web service synchronise timeframes. Such events may include completion of long running tasks or events which users may need to act on such as a transfer of ownership of a request for export. In order to facilitate this notification requirement NEXDOC has implement a series of web services which allow Software Vendors to poll to determine if there are any tasks or events outstanding for their clients so they can be actioned. This process is described in the following web services.

The url for the below services - > https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/NotificationService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

Sample request and responses are provided in Appendix 3.

5.4.1 Notification List

This service is used to receive a list of outstanding events which are yet to be acknowledged by the user. The list is restricted to those events of the client groups of the calling software. The service has

been design to restrict the size of the returned payload to ensure performance of the service is optimal given the intended frequency of use.

The results are returned in ascending date order list and the list can be further filtered via a date range and previously acknowledged events can also be shown or hiding.

Even though the service has been designed for optimum performance the Department recommends that this service is polled only in response to a previous request such a lodge service which resulted in a status of complete or once every 60 seconds.

It should be noted that this service requires different header requirement as it accepts one or more client groups in its body. Refer to the contracts section of this document for more details.

5.4.2 Read Notification

Once a notification is received via the Notification list this service is used to gather details of the notification. By doing so this service provides a mechanism for clients to read the details relating to the notification placed on the registry and also provides a facility for the system to recognise that a user has acknowledged the notification and therefore it can be removed from the registry.

5.5   Request endorsement

This service is used to request a new endorsement from the Department should the need arise. The Department will review the request and determine if this is appropriate. The request identifier is returned in the message and the user will be advised via an e-mail notification of the outcome.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/EndorsementService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

5.6   Request assistance

This service is used to request assistance with a Request for Export from the Department should the need arise. The Department will review the request once received and correspond further if required. The request identifier is returned in the message and the user will be advised via an e-mail notification of the outcome.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/RequestAssistanceService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

5.7   Transfer EDN

This service is used to request an export declaration number (EDN) be transferred from one REX to another. The Department will generate and send the EDI message to ICS on behalf of the user.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ TransferEdnService.

Note: The users are encouraged to use the new customs service described in 'Custom Service' section of this document.

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

6 Request for Export Services

6.1 Validate

This service can be used to validate the details of a request for export against the current lodge business rules without persisting the data within the Departments application. This provide the ability for client to validate requests against the current business rules without submitting the request.

On completion of a Validate request NEXDOC will return an either a series of validation warning and errors or an informational message which indicates that no problems were received with the request.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexSubmissionService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

6.2 Order

This service can be used by Vendors to raise an Order request for export. All data submitted via the order service will be validated against the order business rules and persisted within the NEXDOC application. On successful completion of an Order request NEXDOC will return a Request for Export (REX) number that can then be used either in subsequent order requests or via the lodge service to request a formal submit of the request for processing.

If the REX number is provided in a subsequent order request, then this will amend the details of the existing record. The REX record must exist with an ORDR status for this to be permitted.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexSubmissionService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

6.3 Lodge

This service provides the process for which a client can formally lodge a request for export to the Department. This record would consist of details relating to the products being exported, the export details, identification details including the owner user, organisation and Vendor as well as potentially the certificate request details.

On submission a resulting state will be provided as part of the response to indicate the outcome of the submission. If this state is "Completed" and following receipt of the notification that the certificate is ready for retrieval, the client can then retrieve the certificate via the certificate request service.

Additionally on successful submission a Request for Export (REX) number will be provided that can then be used either in subsequent lodge requests or via the amend service to provide additional information or alter details.

If the REX number is provided in a subsequent lodge request, then this will amend the details of the existing record. The REX record must exist with an ORDR status or INIT for this to be permitted.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexSubmissionService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

6.4　Amend

The amend service allows clients to modify an existing REX record in order to amend existing details or alternatively to add details which were previously omitted from the request. This may include changes to the print indicator or where possible updating the consignee information. There are restrictions around what data can be changed at what status (see Appendix A). If an error code of 2049 is returned in the amend response, then this means that some (or all) of the updated data was not permitted to be changed. There is a boolean called "submitAmendmentRequest" at the root of the SOAP request, which allows the changes to be sent for approval by a departmental officer rather than have the changes applied straight away. When the "submitAmendmentRequest" flag is set to true a reason for the changes (free text) must be provided as well in the amendmentReason field. Once a single amendment request has been submitted, the REX cannot be updated (nor a new request submitted) until that request is either approved or rejected. A notification of type LCA is sent when the amendment request is approved, and type LCJ is sent when the request is not approved. Note that the submitted data is not updated until request is approved.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexSubmissionService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

6.5　Withdraw

This service allows a client to withdraw a request if it is no longer required. Withdrawal of a request can only be requested up to the status of "Certificate Ready". Post this status this service will return a SOAP error stating the action is not allowed.

Once the service returns with a success response, the record will be updated to have the status of Withdrawn and will no longer be able to be updated via the order, lodge or amend services.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexSubmissionService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

6.6　Release REX to print

If a REX is at CTRD (Certificate Ready) then this service allows the REX to be progressed to COMP, where the certificates will be issued either at the selected regional office, or to the selected remote print client group. Note that this service has the same effect as using the Amend service and updating the print indicator to AUTO, however the advantage with this service is that the REX data does not need to be submitted – only the REX number and last updated timestamp do.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexSubmissionService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

6.7   Cancel REX

If a permit number has been issued on a REX, then the REX cannot be withdrawn but must be cancelled instead. The permit number will be issued if the REX is for a prescribed good and the status is INSP, CTRD or COMP (it is also returned in the lodge/amend/read services). This service allows the user to request that a REX be cancelled. Note that the cancel REX service does not apply the cancellation straight away – it must be approved by a departmental officer first. Once the cancel request is approved the status of the REX will be updated to CANC. Please note: Cancelling a REX does not cancel the EDN from ICS. If you wish to also cancel the EDN then this must be performed using the Cancel REX EDN operation on Customs Services endpoint.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexSubmissionService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

6.8   Read

The Read service allows a user to Read the contents of a previously submitted REX record. This service can be used for a variety of reason such as:

- to confirm the details of the request previously submitted;
- to retrieve the latest version of a request;
- to download a copy of the record if it has been transferred or forwarded;
- to check the timestamp matches their current records; or
- to record the final details of the request.

The response payload conforms to the same schema used to order, lodge or amend a record, but also includes an optional list of DefaultLineCeriticates, which will contain any certificate template details that were derived for a given line number, when the certificate template was not specified.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ ReadRexService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

6.9   Read REX EDN

The Read REX EDN service allows user to collect their allocated EDN numbers. Notifications are placed on notification queue when EDN's can be read.

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ ReadRexService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

7 Custom Services

This is the new service developed to allow clients to cancel, read and transfer rex edn.The manually entered EDN overrides the EDN generated from SEW integration.

Note the clients are encouraged to use the new service as opposed to the transferEDN service -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ TransferEdnService

URL for this service is -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/CustomsService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

7.1 Cancel REX EDN

The Cancel REX EDN service allows user to withdraw their EDN numbers. Notifications are placed on notification queue when EDN's can be read.

7.2 Read REX EDN

The Read REX EDN service allows user to read their EDN numbers. Notifications are placed on notification queue when EDN's can be read.

7.3 Transfer REX EDN

The transfer REX EDN service allows user to transfer their EDN numbers. Notifications are placed on notification queue when EDN's can be read.

8 REX Certificate

URL for this service -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexCertificateService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

8.1 Replace

The replacement service allows clients to request a replacement certificate. The service accepts all fields as per the lodge service however no changes are made until the replacement request is accepted.

8.2 Reissue

This service allows clients to request a reprint of a current certificate. The reprint will only occur if the request is accepted and the outcome is notified via the notification service.

9   Certificate Read

This service is responsible for issuing a certificate for a referenced NEXDOC record. The requested record must meet the conditions to have a certificate issued. Where the specific conditions are meet for remote print the response will include the certificate on the first call*. On all subsequent calls this will not be included. It is the Vendor's responsibility to ensure they enforce the print once protocol and do not store the original or allow multiple prints to occur.

If a request for export as in "CTRD" status this service can be used to retrieve a preview of the certificate. This can be used numerous times and is still available at COMP.

*If the certificate is still generating at the time the first call is made, the certificate file will not be returned. A subsequent call once generation has been completed will return the certificate. A 5 second (or more) delay between REX Lodgement and Certificate Read should prevent this situation from ever being encountered.

URL for this service -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ ReadCertificateService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

10   Rex Ownership

NEXDOC supports the ability of clients to forward a request to another party or to transfer ownership of the record to an external party. The services that support these processes are outlined below.

URL for this service -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexOwnershipService

For Vendor test environment please change context root from online.agriculture.gov.au to **online-vnd.agriculture.gov.au**

10.1   Forward

The forward service allows a client to forward ownership of a request for export to another party. Upon receiving a request the service will validate that the submitting client is authorised to make the request and that the external identifier supplied is valid. If both entries are validated the request will be processed.

The forward service also accepts two optional field being status and requires acceptance. When the forward status is used the service will hold the forward request until the submitted status is achieved. When the acceptance indicator is set to true the receiving client group must accept the request before the ownership is forwarded to the new client group. The submitting client group can at any time prior to acceptance withdraw the request.

10.2   Transfer

The transfer service allows a client change the exporter of the REX, and optionally transfer ownership to another party. The service requires both a new exporter identifier and a new client group identifier, although if only the exporter needs updating the client group identifier can be the same as the current owner. The transfer service will return a success message and a notification will be issued to the new client group to accept or reject the transfer request.

Should the user which to cancel the request they can send a subsequent request to the "withdraw" service to cancel the request at any time prior to the new client group actioning the request.

10.3   Withdraw

The withdraw service is used to withdraw any transfer requests or forward requests that require approval that have not yet been actioned by the receiving client group.

10.4   Acknowledgement

Once notified of a transfer or optionally a forward that requires approval the receiving client group will use this service to either accept or reject the transfer. If accepted the ownership of the request will be updated to the new client group. If rejected a notification will be sent via the notification list to the owning client group informing them of the outcome of the request.

11 Request and Response Contracts

Provided below is a summary of all request types, request and responses contractors. Where required each contract will be further explained in the common request types section of this document. Vendors are encouraged to refer to the latest wsdl and xsd files to for the request and response contracts.Alternatively the Vendors can download the wsdl and xsd from the urls provided, by appending' ?WSDL'.

11.1 Authorisation and Authentication

Each web service in the NEXDOC request for export suite needs to include the authentication type which allows the services to authenticate the request against the following user types:

- **Vendor Token** – The Vendor Token must represent an organization that has an approved Vendor registration and must be flagged as active in the target environment. Refer to the Software Vendor Registration section of this document for further information on how to request a Vendor token.

- **Client Group Token** - The client group token represent the approved owner of the request for export. To receive a client group token a user must have successfully applied for the client group administrator role and created a client group token. The client group token must be flagged as active and must be flagged as active in the target environment.

- **Client Token** – To create or interact with any request for exports the client token, which represents the end user, must be approved by the client group administrator and be flagged as active in the target environment. Client tokens are created using the Client Group Access service or are automatically assigned as part of the client group administrator approval process.

An example of the authentication type along with the username and password in the header section of soap messages is shown below:

For more information on getting the username and password

```
<soapenv:Header>                                                    <read:authToken>
    <com:vendorToken>12C2A-XXXX-4B58- XXXX-1C9521551</com:vendorToken>
    <com:clientToken>82553A-XXXX-41BD- XXXX-0AB250A5C</com:clientToken>
    <com:clientGroupToken>EDF0- XXXX-40E3- XXXX-B1A0BC1</com:clientGroupToken>
</read:authToken>
    <wsse:Security>
        <wsse:UsernameToken wsu:Id="UsernameToken-1">
            <wsse:Username>test_ws</wsse:Username>
            <wsse:Password>G2CIbJ8XHn1B/6trq7VoHyDbX24=</wsse:Password>
        </wsse:UsernameToken>
    </wsse:Security>                                                 <soapenv:Header>
```

Any requests that fail validation against this schema will return the standard schema validation to the requestor whilst all other data validation, that have a valid xml structure, will return the soap fault formatted with the error/warning messages.

Request Type	Request Contract	Response Contract
Reference Data (SOAP) URL = https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ReferenceDataService	security • InstallationToken and Password codeSetName (m) codeSetDescription codeSetValue codeSetUpdateRangeDate • codeSetUpdateTimestampStart • codeSetUpdateTimestampEnd	codeSetName listCodeSet listPosition listItemCodeSet • key • value • type
Reference Data (REST) URL = https://online.agriculture.gov.au/ nexdoc-rs/api/public/v1/reference-data/{codesetName}[?lastUpdated=2019-01-01T00:00:00]	security – Basic Authentication • InstallationToken and Password attached to the HTTP Request Header in Authorisation • codeSetName (m) lastUpdated	**Media type:** text/csv **File name:** [codeSetName].csv First row contains column headings All values enclosed within double-quote characters
Client URL -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/PartyService	security • vendorToken • InstallationToken and Password newClientDetails • firstName • lastname • email • description clientGroupAdminEmail	clientToken message
Exporter Details URL -> https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/PartyService	security • vendorToken • InstallationToken and Password readExporterDetails • identifier • identifierType	readExporterDetails • exporterID • exportername • exporteraddress
Request Endorsement URL = https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/nexdoc-ws-ext-proxy/EndorsementService	Authentication (m) EndorsementInquiry (m) • reason • endorsementText product Details • country • commodityType • productType • packType • preservationType document • name • mimeType • data	servicerequestIdentifer

Request Type	Request Contract	Response Contract
Request Assistance URL = https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RequestAssistanceService	Authentication (m) • Rex • Description • Document	servicerequestIdentifer

Transfer EDN (Old supported)
 	
URL =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ TransferEdnService
	Authentication (m)
	TransferEdn
	• transferEdnFrom
	   o rexNumber
	   o lastAmendDateTime
	• transferEdnTo
	   o rexNumber
	   o lastAmendDateTime
		TransferEdnResponse
		• rexResponseDetails
		• validationNotices

Transfer EDN (New Custom Service)
 	
URL =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ CustomsService
	Authentication (m)
	TransferRexEdn
	• transferEdnFrom
	   o rexNumber
	   o lastAmendDateTime
	• transferEdnTo
	   o rexNumber
	   o lastAmendDateTime
		TransferRexEdnResponse
		• rexResponseDetails
		• validationNotices

Read Rex EDN URL = https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/CustomsService	Authentication (m) ReadRexEDN • rexNumber	ReadRexEdnResponse • edn
Cancel Rex EDN URL = https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/CustomsService	Authentication (m) CancelRexEDN • rexNumber • lastAmendDateTime	CancelRexEdnResponse • validationNotices
Notification List URL = https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ NotificationService	security • vendorToken • installation Token and Password clientgroups • clientgroup filterDateRange • startDateTime • endDateTime showPreviouslyAcknowledged	Notification • Notification Id • Notification type • Notification title • ClientGroup • Date

Request Type	Request Contract	Response Contract
Read Notification URL = https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ NotificationService	Authentication (m) Notification Id (m)	notification • notificationId • date • notificationType • notificationTitle • notificatoinText

URL for Validate, Order, Lodge, Copy and Amend =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexSubmissionService
 	
Validate
 	
Order
 	
Lodge
 	
Amend
 	
Replacement (URL =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexCertificateService)
	identification
	  rexNumber
	  lastAmendDateTime
	exportDetails
	  commodityType
	  priority
	  departureDate
	  transportDetails
	    transportMode
	    voyageOrFlightNumber
	    vesselName
	    shippingCompany
	    storeTransportTemperature unit
	    vesselHoldSeals
	      sealStartNumber
	      sealEndNumber
	  destinationCity
	  dischargePorts
	    dischargePort
	    removeExistingSet
	  destinationCountry
	  transitCountries
	    transitCountry
	    removeExistingSet
	  borderInspectionPort
	  loadingPorts
	    loadingPort
	    removeExistingSet
		Identification
		• rexNumber
		ComplianceStatus
		lastAmendDateTime
		exporterReference

Request Type	Request Contract	Response Contract
	importPermits	
	importPermit	
	importPermitNumber	
	importPermitDate	
	removeExistingSet	
	sew	
	customsAgentIndicator	
	edn	
	currency	
	consigneeName	
	ownerExporterId	
	exporterDeclaration	
	exporterDeclarationCode	
	removeExistingSet	
	imaDetailsList	
	 imaDetails	
	   serialNumber	
	   containerNumber	
	   sealNumber	
	   invoiceDate	
	   invoiceNumber	
	   grossWeight unit	
	   netWeight unit	
	   productDescription	
	   quotaYear	
	 removeExistingSet	
	exporterReference	
	consigneeDetails	
	consigneeName	
	consigneeAddress	
	streetAddress	
	streetLine	
	city	
	state	
	country	

Request Type	Request Contract	Response Contract
	postalCode	
	consigneePhoneNumber type	
	consigneeRepresentative	
	messageAcknowledgements	
	messageId	
	removeExistingSet	
	preferredWeightUnit	
	tracesApprovalId	
	importedProductFlag	
	manufacturedTreatedPackagedLabelledInAustralia	
	legallyImported	
	lotNumber	
	storageEstablishmentNumber	
	quotaYear	
	shipStoresFlag	
	exemptionCode	
	certificateDetails	
	certificatePrintControls	
	certificatePrintIndicator	
	certificatePrintRegion	
	certificateRequiredClientGroup	
	separateBy	
	certificates	
	certificate	
	removeEntry	
	lineNumbers	
	lineNumber	
	certificateDetails	
	certificateTemplate	
	certificateEndorsement	
	certificatePrintDetails	
	certificatePrintRegion	
	certificateRequiredClientGroup	
	productLines	

Request Type	Request Contract	Response Contract
	productLine	
	lineNumber	
	removeEntry	
	productDetails	
	productType	
	category	
	packType	
	preservationType	
	cutType	
	suppCode	
	outerProductPackaging	
	quantity packageType	
	unitAmount unit	
	packageMeasureAccuracy	
	shippingMarks	
	intermediateProductPackaging	
	quantity packageType	
	unitAmount unit	
	packageMeasureAccuracy	
	innerProductPackaging	
	quantity packageType	
	unitAmount unit	
	packageMeasureAccuracy	
	netMetricWeight unit	
	netImperialWeight unit	
	manualCertificateProductDescription	
	farmCode	
	farmType	
	containers	
	container	
	containerNumber	
	containerSeals	
	containerSeal	
	sealNumber	
	sealStartNumber	

Request Type	Request Contract	Response Contract
	sealEndNumber	
	containerSeal	
	removeExistingSet	
	treatments	
	treatmentType	
	treatmentCode	
	treatmentStartDate	
	treatmentEndDate	
	treatmentInformation	
	removeExistingSet	
	productionProcesses	
	productionProcess	
	processGroup	
	processingStartDate	
	processingEndDate	
	establishmentIndicator	
	processingEstablishmentNumber	
	processingEstablishment	
	establishmentName	
	establishmentAddress	
	streetAddress	
	streetLine	
	streetAddress	
	city	
	state	
	country	
	postalCode	
	establishmentAddress	
	contactPhoneNumber type	
	contactName	
	removeExistingSet	
	sew	
	netCustomsWeight unit	
	grossMetricWeight unit	
	fobAmount	

Request Type	Request Contract	Response Contract
	productSourceState	
	relatedPermitType	
	relatedPermitNumber	
	relatedPermitDate	
	durabilityStartDate	
	durabilityEndDate	
	aheccCode	
	batchCode	
	additionalTexts	
	additionalText	
	code	
	text	
	productSourceCountries	
	productSourceCountry	
	removeExistingSet	
	attachments	
	attachment	
	name	
	removeEntry	
	description	
	mimeType	
	data	
	importAuthorityCode	
	natureOfCommodity	
	euTreatmentType	
	manufacturers	
	manufacturer	
	lineNumbers	
	lineNumber	
	manufacturerDetails	
	name	
	address	
	streetAddress	
	streetLine	
	city	

Request Type	Request Contract	Response Contract
	state	
	country	
	postalCode	
	additionalTexts	
	additionalText	
	code	
	text	
	euTransit	
	personResponsible	
	address	
	streetAddress	
	streetLine	
	city	
	state	
	country	
	postalCode	
	firstName	
	lastName	
	phoneNumber	
	placeOfDestinationDetails	
	name	
	address	
	streetAddress	
	streetLine	
	city	
	state	
	country	
	postalCode	
	phoneNumber	
	approvalNumber	
	transitLocationType	
	authorisationDetails	
	authorisationDate	
	authorisingEntityId	
	comments	

Request Type	Request Contract	Response Contract
	attachments	
	attachment	
	name	
	removeEntry	
	description	
	mimeType	
	data	

CopyRex
 	
URL =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexSubmissionService
	Authentication (m)
	Identification (m)
	  • RexNumber
	  • lastAmendDateTime
		value
		newRexNumber
		lastAmendDateTime

Read URL = https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ ReadRexService	Authentication (m) Identification (m) • RexNumber(m)	Identification Export Details productLines (1 or more) Manufacturers Certificate Details Attachments
ReadREXEDN URL = https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ ReadRexService	RexNumber (m)	EDN

Withdraw
 	
URL =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexOwnershipService
	Authentication (m)
	Identification (m)
	  • RexNumber
	  • lastAmendDateTime
		withdrawalRexResponse
		  • value

Certificate Read
 	
URL for this service ->
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ ReadCertificateService
	Authentication (m)
	Identification (m)
	  • RexNumber
		Certificates
		  • Consignee
		  • Departuredate
		  • Certificates
		    • Certificate number
		    • certificate Payloads

Request Type	Request Contract	Response Contract

Reissue
 	
URL =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexCertificateService
	Authentication (m)
	Identification (m)
	   • RexNumber
	Reason (m)
		ReprintResponse
		   • value

REX Forward Ownership
 	
URL =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexOwnershipService
	Authentication (m)
	Identification (m)
	   • RexNumber
	clientGroup (m)
	RequiresAcceptance (m)
		rexOwnershipResponse
		   • value

REX Transfer Ownership
 	
URL =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexOwnershipService
	Authentication (m)
	Identification (m)
	   • RexNumber
	clientGroup (m)
	exporter (m)
		rexOwnershipResponse
		   • value

REX Withdrawn Ownership
 	
URL =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexOwnershipService
	Authentication (m)
	Identification (m)
	   • RexNumber
		rexOwnershipResponse
		   • value

REX Acknowledge Ownership
 	
URL =
 	
https://online.agriculture.gov.au/nexdoc-ws-ext-proxy/ RexOwnershipService
	Authentication (m)
	Identification (m)
	   • RexNumber
	isAccepted(m)
		rexOwnershipResponse
		   • value

12   Additional Information

12.1   Request Timestamp

To assist end users identifying that they may not be able to update the latest version of a request for export NEXDOC has included a request timestamp. This timestamp will be used to address potential problems where two or more authenticated users with the ability to update the same request cause concurrency issues.

The timestamp is generated whenever the Lodge, Amend or Order services are called by an external client and is returned in the corresponding response. The timestamp can also be accessed via the Read service. The timestamp is updated regardless of what is added, modified or delete for that particular record. An example of the return is provided below:

```
<ns1:rexResponseDetails>
    <ns2:complianceStatus>COMP</ns2:complianceStatus>
    <ns2:lastAmendDateTime>2017-08-10T14:16:45.524+10:00</ns2:lastAmendDateTime>
</ns1:rexResponseDetails>
```

Subsequent amendments made by either the Lodge, Amend or Order services are required to present this timestamp, to ensure that the user has the latest revision of this REX record. An example is provide below:

```
<amen:identification>
    <basecom:rexNumber>12100003182</basecom:rexNumber>
    <basecom:lastAmendDateTime>2017-08-10T14:16:45.524+10:00</basecom:lastAmendDateTime>
</amen:identification>
```

12.2   Remove Existing Set

To allow clients to remove existing values for repeated element types, for example TransitCountries or LoadingPorts, NEXDOC has provided the removeExistingSet element type. When this flag is set to true during an amend operation NEXDOC will action the request by removing all of the current values.

An example of this might be where a client uses the Order service to submit a request and provide one transit counties as per the below:

```
<com:transitCountries>
    <com:transitCountry>FR</com:transitCountry>
</com:transitCountries>
```

On a subsequent request the user may wish to remove all existing transit countries by specifying the removeExistingSet element with a value of 'true'.

```
<com:transitCountries>
<com1:removeExistingSet>true</com1:removeExistingSet>
</com:transitCountries>
```

Importantly if no TransitCountries element was provided during this amend operation, then the Transit countries would remain intact, as they were prior to the amend service call.

12.3   Altering Existing Set

For these repeated element types (where multiple values are permitted), changes to these values are made by replacing the entire set with the new values provided. For example for an existing record with values:

```
<com:transitCountries>
```

```
        <com:transitCountry>FR</com:transitCountry>

    </com:transitCountries>
```

If an amend operation is provided with the new values of:

```
    <com:transitCountries>

        <com:transitCountry>NZ</com:transitCountry>

        <com:transitCountry>JP</com:transitCountry>

    </com:transitCountries>
```

Then thre resulting transit countries after this operation has been completed would be NZ and JP. Note FR would have been removed.

12.4   Existing Set Behaviour

Provided below is an overview of expected behaviours when using the existing set funcationality.

If submitting an Order or Lodge request without an identification element and the:

- list container element (eg. transitCountries) is omitted completely *then* an empty list is stored;

- list container element is included but has no children then an empty list is stored;

- list container element is included and has 1 or more children but does not include the removeExistingSet then the provided list is stored

- list container element is included and contains a single removeExistingSet element *then* an empty list is stored

When submitting an Order, Lodge or Amend request with an identification element ie. for an existing rex record and the:

- list container element is omitted completely *then* existing list remains unchanged;

- list container element is included but has no children *then* existing list remains unchanged

- list container element is included and has 1 or more children but not a removeExistingSet existing list remains unchanged;provided list replaces existing list

- list container element is included and contains a single removeExistingSet element *then* empty list replaces existing list

12.5   Message Acknowledgements

As part of normal process NEXDOC will return some business rule validations as "Warnings". In order to successfully submit a request these warnings must be resolved or acknowledged via the messageacknowldgement section.

Provided below is a sample of a warning that can be returned as part of a lodge request:

```
    <ns2:validationNotices>

        <ns0:notice>

            <ns0:noticeId>241</ns0:noticeId>

            <ns0:noticeType>W</ns0:noticeType>

            <ns0:noticeMessage>Consignee Address must be present before H/C can be
    printed</ns0:noticeMessage>

            </ns0:notice>

        </ns2:validationNotices>
```

NOTE: Currently when the lodge fails to update/create the REX, a SOAP fault is returned instead. To read the message type from a soap fault check the fault reason. Provided below is a sample of a warning that can be returned as part of a failed lodge request:

<ns0:NexdocSoapFaultItem>

  <ns0:FaultCode>**241**</ns0:noticeId>

  <ns0:FaultReason>WARNING</ns0:noticeType>

  <ns0:FaultMessage>Consignee Address must be present before H/C can be printed</ns0:FaultMessage>

  </ns0:NexdocSoapFaultItem>

To acknowledge this warning the following is included in the order request.

<com1:messageAcknowledgements>

  <com:messageId>**241**</com:messageId>

</com1:messageAcknowledgements>.

12.6 Exporter Declaration

When a declaration is required for the submission of a Request for Export clients have two options. Either the client can provide a free text declaration using the exporterDeclaration tag which will be verified on or post submission or they can use the declaration codes.

If on lodge no free text or declarations codes are provided NEXDOC will return the required declaration codes as part of the lodge message response. An example of the response is provide below:

<ns0:FaultMessage>Exporter declaration code(s) [DEC02 – This product is wholly made in Australia] must be provided</ns0:FaultMessage>

If the client wish to use the provided declaration the Vendor software must present these codes along with the description for the client to certificate compliance. Once compliance is confirmed the codes should be provided within the lodge request as per the following example:

<com1:exporterDeclaration>

<com1:exporterDeclarationCode>DEC02</com1:exporterDeclarationCode>

</com1:exporterDeclaration>

12.7 AEPI Authorisation

When an authorised user is also an AEPI authorised user, they can self authorise a REX so that it does not have to be manually processed by a departmental officer. To do this, an authorisation date must be provided (under exportDetails/authorisationDetails). The authorisationFlag (under exportDetails) must also be set to "Y". If these two conditions are met, and the current user is AEPI authorised, then the REX will be granted a permit immediately upon lodgement (if REX is SEW it may take a short time for the permit number to reach ICS). The REX can also be AEPI authorised post lodgement using the Amend service. Note that if the authorisationFlag is set to "Y" but no date is provided, then the REX will remain at FINAL however no departmental officer will attempt to authorise the REX. If you wish for a departmental officer to authorise the REX then the authorisationFlag should be set to "N". Doing this will notify a departmental officer that the REX requires authorisation.

13   Paperless Certification

TRACES is the European Commission's online platform for sanitary and phytosanitary certification required for the importation of animals, animal products, food and feed of non-animal origin and plants into the European Union, and the intra-EU trade and EU exports of animals and certain animal products.

It facilitates the progression to paperless certification through the exchange of data, information and documents between all involved trading parties and control authorities and therefore simplifies and speeds up the administrative procedures.

For Australia to progress to paperless certification for export commodities to the European Union, NEXDOC electronic certificates must comply with and conform to the new TRACES NT system by the 21 August 2021.

The transition to paperless certification will require changes to the NEXDOC Web Services Interface and underlying business logic including new REX fields and REX related messages. Software Vendors will be required to update their applications in support of these NEXDOC changes:

1. Field - Combined Nomenclature (CN)
    a. This element will be at the Product Level
    b. If commodity type is Dairy, and Destination Country is a member of the Country Group European Union then CN is mandatory
    c. CN is a free text data element and must be six or eight characters long and contain values 0 to 9 only
2. Field - Final Consumer
    a. This element will be at the Product Level
    b. If commodity type is Dairy, and Destination Country is a member of the country group European Union then Final Consumer is mandatory
    c. Final Consumer is a Boolean data element
3. Field - Place of Destination (Group)
    a. This element will be at the REX Header Level
    b. If commodity type is Dairy, and Destination Country is a member of the Country Group European Union then the Place of Destination group of elements is mandatory
    c. This group is expected to include the following data elements:
        i. ID
        ii. Name
        iii. Address Line
        iv. City Name
        v. Country ID
4. The Permissions table will be updated for these new elements

Note 1: Data Element can only be changed at COMP if the Customs Agent Indicator on the REX is equal to 'Y'.

Note 1: Data Element can only be changed at COMP if the Customs Agent Indicator on the REX is equal to 'Y'.

Appendix A Dairy and Eggs - Change Permissions Matrix

COMMODITY	KEY
Dairy	D
Eggs	E

Note: Where a commodity does not appear against the field in any status, it is because that field is not relevant to the commodity.

CHANGES PERMITTED BY EXPORTER WITHOUT APPROVAL

REX Header

FIELDS	ORDR	Before Authorised	Authorised	COMP
Additional Information	D	D	D	
Additional Product Description	DE			
Authorising Estab Number	DE			
Border Inspection Port	DE	DE		
Certificate Print Indicator	DE	DE	DE	
Certificate Print Location	DE	DE	DE	
Certificate Required Location	DE	DE	DE	
Consignee City	DE	DE	DE	
Consignee Country	DE	DE	DE	
Consignee Name	DE	DE	DE	
Consignee Phone Number	DE	DE	DE	
Consignee Postcode	DE	DE	DE	
Consignee State	DE	DE	DE	
Consignee Street	DE	DE	DE	
Consignee TRACES Approval ID	D	D	D	
Customs Agent Indicator	DE	DE	DE	
Departure Date	DE	DE	DE	
Destination City	DE	DE	DE	
Destination Country	DE			
Discharge Port	DE	DE	DE	
EDN – Exporter supplied	DE	DE	DE	
Exporter Declaration	DE	DE	D	
Exporter Reference	DE	DE	DE	
FOB Currency Unit (see note 1)	DE	DE	DE	D
Forwarding Instructions	DE	DE	DE	
Forwarding Status	DE	DE	DE	
Import Permit Date	DE	DE	DE	

FIELDS	ORDR	Before Authorised	Authorised	COMP
Import Permit Number	DE	DE	DE	
Imported Product Flag	D			
Header Storage Establishment	D	D		
Loading Port	DE	DE	DE	
Notify Party Text	D	D	D	
Person Responsible for Consignment	D	D	D	
Person Responsible Address	D	D	D	
Person Responsible Contact Details	D	D	D	
Place of Destination Details	DE	DE	DE	
Print Location	DE	DE	DE	
Product Use Indicator	D	D		
Separate HC Container Indicator	DE	DE	DE	
Separate HC Marks Indicator	DE	DE	DE	
Separate HC Processor Indicator	DE	DE	DE	
Transferee Exporter Number	DE	DE	DE	
Transit Country	DE	DE	DE	
Transport Mode	DE	DE	DE	
Transport Temperature Min Max	DE	DE	D	
Transport Temperature Unit	DE	DE	D	
True and Complete Indicator	DE			
Vessel Name	DE	DE	DE	
Voyage/Flight Number	DE	DE	DE	
IMA 1 Information	D	D	D	

REX Line

FIELDS	ORDR	Before Authorised	Authorised	COMP
AHECC Code	DE	DE	DE	
Airfreight Seals	DE	DE	DE	
Batch Code	D	D	D	
Category Code	DE	DE		
Certificate Product Description	D	D	D	
Certificate Template & Endorsement	DE	DE	DE	
Container Number (Sea) [see note 2]	DE	DE	DE	
Container Number (Air)	DE	DE	DE	
Container Seals (see note 2)	DE	DE	DE	
Extra Certificate	DE	DE	DE	
Farm Code	E	E		
Farm Type	E	E		
Import Authority Code	D	D	D	
Line Customs Weight Amount	D	D	D	
Line Customs Weight Unit	D	D	D	
Line Metric Gross Weight Amount	DE	DE	DE	
Line Metric Gross Weight Unit	DE	DE	DE	
Line Metric Net Weight Amount	DE	DE	DE	
Line Metric Net Weight Unit	DE	DE	DE	
Line Imperial Weight Amount	E	E	E	
Line Imperial Weight Unit	E	E	E	
Package Metric Net Weight Amount	DE	DE	DE	
Package Metric Net Weight Unit	DE	DE	DE	
Package Quantity	DE	DE	DE	
Package Type	DE	DE	DE	
Package Weight Accuracy	D	D	D	
Packaging Level	DE	DE	DE	
Process Date	DE			
Processing Establishment Number/Name	DE			
Product User Indicator	D	D	D	
Product Code	DE			
Product Source State	DE	DE	DE	
Quota Year	D	D	D	
Quota Exporter Number	D	D	D	
Shipping Marks	DE	DE	DE	
Treatments	E	E		
Use By/Best Before Date	DE	DE	DE	

1   Data Element can only be changed at COMP if the Customs Agent Indicator on the REX is equal to 'Y'.

2   In some cases the Data Element can only be changed after Authorisation if the authenticated user is an Approved Export Permit Issuer (AEPI user).

3   If a status is not mapped for a given attribute, then if that attribute is changed at that status it must be requested and approved first. See the Amend service description (section 7.4) to see how an amendment request can be submitted.

Reference Data Codesets

Please refere to **CodeSetName** in xsd/common/**EnumTypes.xsd** for current code set names in the wsdl zip file provided

The **ECM_PORT_LIST** code set contain more than 10,000 records.