



AP E-cert Submissions using SOAP

Technical User Guide

Copyright

Work that is produced relating to this project will remain the property of the Crown. MAF will hold all copyright and other intellectual property rights in respect of such work on behalf of the Crown. Individuals wishing to publish work commissioned by MAF must first seek prior written approval from MAF to do so.



Version History

Date	Version	Description
4 October 2007	0.1	Draft
31 October 2007	0.2	Updated Declaration Section for Export Certificates
05 December 2007	0.3	Updated post review by Anita Patel, NZFSA
23 June 2008	0.4	Updated data standards
12 September 2008	0.5	Include Extra SOAP Methods: Cancel, Amend and 'Find_Details_by_Update_Date'.
30 October 2008	0.6	Include Extra SOAP Methods: 'Find_Third_Party_Details_by_Update_Date'. Also confirm the new requirements for the details in the AP E-cert XML Submission Data Standards.
15 July 2009	0.7	Removed checksum references and replaced proprietary batch file tags with XML tags
November 2009	0.8	Addition of new Get method, review of submission data standard
16 May 2011	0.9	Removal of .NET examples

Document Purpose

The purpose of this document is to provide Industry users who intend to implement Animal Product (AP) E-cert Submissions via XML and SOAP with sufficient information to:

- Establish a SOAP client
- Test Connectivity
- Confirm the format of their XML files is correct in the AP E-cert context
- Understand the interaction between the user's own systems and AP E-cert via SOAP and
- Understand the XML responses returned by AP E-cert

Intended Audience

This document is targeted at users who are familiar with the concept of SOAP as a data exchange mechanism and who are familiar with the data requirements of AP E-cert.

Document Scope

This document

- Provides a brief introduction to XML and SOAP
- Details the Business Rules included within the AP E-cert XML and SOAP Submission Interface.
- Provides detail for the development of a SOAP client for AP E-cert
- Provides detail of the search, retrieve, and submission methods (interactions) currently available within AP E-cert using XML and SOAP
- Defines the XML structure required for each interaction

Support

MAF cannot provide any development/coding support for the creation of a SOAP client but does provide a general help function with regards to WSDL files, DTDs and SOAP methods.



TABLE OF CONTENTS

VERSION HISTORY	2
DOCUMENT PURPOSE	2
INTENDED AUDIENCE.....	2
DOCUMENT SCOPE	2
SUPPORT.....	2
1. OVERVIEW	5
1.1. BACKGROUND TO XML	5
1.2. BACKGROUND TO SOAP.....	5
1.3. USE OF XML AND SOAP IN AP E-CERT.....	5
1.4. IMPACT ON AP E-CERT USERS	5
1.5. SCOPE OF XML AND SOAP IN AP E-CERT	6
1.5.1. Searching for Eligibility Documents and Export Certificate	6
1.5.2. Retrieving Eligibility Document and Export Certificate Detail	6
1.5.3. Submitting (Raising) Eligibility Document and Export Certificate Detail	6
1.5.4. Cancel an Eligibility Document or an Export Certificate	7
1.5.5. The amendment of an Eligibility Document or an Export Certificate.....	7
1.5.6. Recording Inspection outcome for Post Approval.....	7
1.6. SECURITY	8
1.6.1. Firewalls.....	8
1.6.2. Authentication.....	8
1.6.3. Data Encryption	8
1.6.4. Additional Security Measures	8
2. BUSINESS RULES	9
3. SOAP CLIENT CONNECTIVITY.....	10
3.1. INTRODUCTION	10
3.2. CONNECTIVITY	10
3.2.1. Server Locations.....	10
3.2.2. User Authentication.....	10
3.2.3. Usage	10
3.2.4. WSDL	10
3.2.5. Error Handling.....	10
3.2.6. Connectivity Testing	11
3.3. WEB SERVICE DEFINITIONS	11
3.3.1. E-cert Live – https://sancrt.maf.govt.nz/APEcertServiceLive/APEcertLive.wsdl	11
3.3.2. E-cert Training - https://ectrain.maf.govt.nz/APEcertServiceTrain/APEcertTrain.wsdl	11
4. XML AND SOAP INTERFACE METHODS	12
4.1. INTRODUCTION	12
4.2. SEARCH METHODS.....	12
4.2.1. <i>find_eligibility_documents_by_update_date_and_status</i>	12
4.2.2. <i>find_eligibility_documents_by_update_date</i>	13
4.2.3. <i>find_eligibility_documents_details_by_update_date</i>	13
4.2.4. <i>find_third_party_eligibility_documents_details_by_update_date</i>	14
4.2.5. <i>find_certificates_by_update_date_and_status</i>	15
4.2.6. <i>find_certificates_by_update_date</i>	16
4.2.7. <i>find_export_certificates_detail_by_update_date</i>	17
4.2.8. <i>find_third_party_export_certificates_detail_by_update_date</i>	17
4.3. RETRIEVE DATA METHODS	19
4.3.1. <i>get_eligibility_document_xml</i>	19
4.3.2. <i>get_multiple_eligibility_document_xml</i>	19
4.3.3. <i>get_eligibility_document_process_history_xml</i>	19
4.3.4. <i>get_export_certificate_xml</i>	20
4.3.5. <i>get_multiple_export_certificate_xml</i>	20
4.3.6. <i>get_official_certificate_xml</i>	20



4.3.7.	<i>get_multiple_official_certificate_xml</i>	20
4.4.	SUBMISSION METHODS	21
4.4.1.	<i>raise_eligibility_document</i>	21
4.4.2.	<i>resubmit_eligibility_document</i>	21
4.4.3.	<i>exhaust_eligibility_document</i>	21
4.4.4.	<i>exhaust_eligibility_document_products</i>	22
4.4.5.	<i>amend_eligibility_document</i>	22
4.4.6.	<i>raise_export_certificate</i>	22
4.4.7.	<i>resubmit_export_certificate</i>	23
4.4.8.	<i>amend_export_certificate</i>	23
4.5.	CHANGE STATUS METHODS	23
4.5.1.	<i>accept_certificate</i>	23
4.5.2.	<i>detain_certificate</i>	24
4.5.3.	<i>reject_certificate</i>	24
4.5.4.	<i>request_replacement_certificate</i>	24
4.5.5.	<i>acknowledge_certificate</i>	25
4.6.	CANCEL METHODS	25
4.6.1.	<i>cancel_eligibility_document</i>	25
4.6.2.	<i>cancel_export_certificate</i>	25
5.	AP E-CERT XML SUBMISSION DATA STANDARDS	26
5.1.	ELIGIBILITY DOCUMENT SUBMISSION DATA STANDARDS	26
5.1.1.	<i>Wrapper</i>	26
5.1.2.	<i>ED Header Submission Data</i>	26
5.1.3.	<i>ED Header Example</i>	29
5.1.4.	<i>ED Product and Process Submission Data</i>	30
5.1.5.	<i>ED Product and Process Example</i>	33
5.2.	EXPORT CERTIFICATE SUBMISSION DATA STANDARDS	34
5.2.1.	<i>Introduction</i>	34
5.2.2.	<i>Wrapper</i>	34
5.2.3.	<i>EC Header Submission Data</i>	34
5.2.4.	<i>EC Header Example</i>	38
5.2.5.	<i>EC Product and Process Submission Data</i>	39
5.2.6.	<i>EC Product and Process Example</i>	42
5.2.7.	<i>EC Declaration Submission Data</i>	43
5.2.8.	<i>EC Declaration Example</i>	44
6.	ELIGIBILITY DOCUMENT SUBMISSION DTD AND EXAMPLE	45
6.1.	ELIGIBILITY DOCUMENT SUBMISSION DTD	45
6.2.	ELIGIBILITY DOCUMENT SUBMISSION EXAMPLE	45
7.	EXPORT CERTIFICATE SUBMISSION DTD AND EXAMPLE	47
7.1.	EXPORT CERTIFICATE SUBMISSION DTD	47
7.2.	EXPORT CERTIFICATE SUBMISSION EXAMPLE	47

1. Overview

1.1. Background to XML

The Extensible Markup Language, XML, has become an international standard for exchanging data. It is independent of particular technology, e.g. specific database products and application languages.

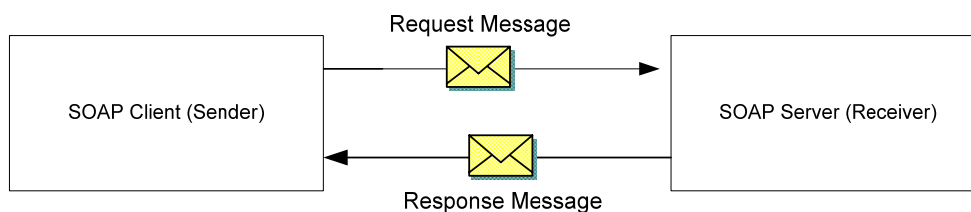
Each XML file is accompanied by a Data Type Definition (DTD), which list all data elements, their names, their relationships, and their optionality. For example, the issue date of the certificate contains a day and a time component, both of which must be stated.

The DTD can be used to enforce these rules when the data is validated, ensuring high data integrity when exporting and importing data from one system to another.

1.2. Background to SOAP

SOAP (originally Simple Object Access Protocol) is a protocol for exchanging XML-based messages over computer networks. First designed in 1998 with backing from Microsoft, the SOAP specification is currently maintained by the XML Protocol Working Group of the World Wide Web Consortium (W3C).

There are several different types of messaging patterns in SOAP, but by far the most common is the Remote Procedure Call (RPC) pattern, in which one network node (the client) sends a request message to another node (the server), and the server immediately sends a response message to the client.



1.3. Use of XML and SOAP in AP E-cert

In the context of AP E-cert, XML and SOAP over HTTPS have been adopted because they

- recognise the same AP E-cert usernames and passwords
- are as secure as the main AP E-cert system because it encrypts messages
- are an Internet standard (endorsed by the W3C)

1.4. Impact on AP E-cert Users

Users wishing to interface with AP E-cert via XML and SOAP will need to add SOAP over HTTPS functionality to their individual systems to connect to the AP E-cert SOAP servers. This process is simplified by the availability of various SOAP software development kits (for instance .net, Soap::Lite).

Once AP E-cert SOAP is accessible from within the user's environment, software can **request, retrieve, submit, amend and cancel** information in XML format from AP E-cert via SOAP.



1.5. Scope of XML and SOAP in AP E-cert

The original implementation of XML and SOAP in AP E-cert was in early 2007. New methods have been introduced November 2008.

1.5.1. Searching for Eligibility Documents and Export Certificate

Actor	Activity
SOAP Client	<ul style="list-style-type: none"> The Find requests of Eligibility Documents or Export Certificates is a Date Range
SOAP Server	<ul style="list-style-type: none"> Receives Request, i.e. E-cert Identifies qualifying records (linking with E-cert) or produces an error Produces XML file of qualifying shoulder numbers Sends the file to the SOAP client
SOAP Client	<ul style="list-style-type: none"> Receives Shoulder Numbers or error message Receives for the new Find Detail and Find Third Party Details methods, the Shoulder Number, Update Date, Status Name, Submission Date, and the Submitter.

1.5.2. Retrieving Eligibility Document and Export Certificate Detail

Actor	Activity
SOAP Client	<ul style="list-style-type: none"> Requests Eligibility Document or Export Certificate data
SOAP Server	<ul style="list-style-type: none"> Receives Request, i.e. E-cert Identifies qualifying records (linking with E-cert) or produces an error Produces XML file of qualifying documents or certificates (where data can be returned) Sends the file to the SOAP client
SOAP Client	<ul style="list-style-type: none"> Receives Eligibility Document or Export Certificate Data or error message

1.5.3. Submitting (Raising) Eligibility Document and Export Certificate Detail

Actor	Activity
SOAP Client	<ul style="list-style-type: none"> Supplies Eligibility Document or Export Certificate data
SOAP Server	<ul style="list-style-type: none"> Receives Request, i.e. E-cert Checks the format of the request Validates Document or Certificate Data (processes acceptance of data and generates an eligibility document or export certificate or produces an error) Produces XML file of successful submission response or error message Sends the file to the SOAP client
SOAP Client	<ul style="list-style-type: none"> Receives successfully submission response or error message

1.5.4. Cancel an Eligibility Document or an Export Certificate

Actor	Activity
SOAP Client	<ul style="list-style-type: none"> Requests the cancellation of an Eligibility Document or an Export Certificate for a Shoulder number.
SOAP Server	<ul style="list-style-type: none"> Receives request to cancel an Eligibility Document or an Export Certificate, Identifies the record to be cancelled. Validates the permissions and reason for the cancellation (processes acceptance of data and cancels eligibility document or export certificate or produces an error) Produces XML file (which includes a check sum validation) of successful submission response or error message Sends the file to the SOAP client
SOAP Client	<ul style="list-style-type: none"> Receives a response from the SOAP server indicating whether the cancellation was successful

1.5.5. The amendment of an Eligibility Document or an Export Certificate

Actor	Activity
SOAP Client	Requests the amendment of an Eligibility Document or an Export Certificate
SOAP Server	<ul style="list-style-type: none"> Receives Request to amend an Eligibility Document or an Export Certificate for a shoulder number Identifies and validates the record to be amended Validates the permissions and reason for the amendment (processes acceptance of data and updates an eligibility document or export certificate, or produces an error) Produces XML file (which includes a check sum validation) of successful submission response or error message Sends the file to the SOAP client
SOAP Client	<ul style="list-style-type: none"> Receives a response from the SOAP server indicating whether the amendment was successful

1.5.6. Recording Inspection outcome for Post Approval

This functionality is available to Border Control Agencies of Importing Countries with permissions to access E-cert to retrieve data for export consignments destined for their port or country. It is provided at this point for reference only. **Paperless Certification with NZFSA – Technical Guide** provides full detail of this functionality.

Actor	Activity
SOAP Client	<ul style="list-style-type: none"> • Identifies the appropriate inspection outcome for a particular sanitary certificate: Accepted, Rejected, Detained, Request Replacement (of sanitary certificate). • From December 2007, Acknowledge receipt of certificate data prior to inspection • Issues a SOAP update request to E-cert for that certificate
SOAP Server	<ul style="list-style-type: none"> • Receives the request • Identifies the sanitary certificate on its database • Validates the status change request against the current status of the certificate • Compiles an error message if the certificate is not found or the status change cannot be actioned • Sends the update confirmation to the SOAP client
SOAP Client	<ul style="list-style-type: none"> • Receives update confirmation as return parameters

1.6. Security

Data security is of paramount importance and the AP E-cert system was designed to provide maximum protection from unauthorised access and use.

For example: A user may be granted access to AP E-cert, but is not able to see eligibility documents or export certificates for organisations they have not been granted permissions for.

1.6.1. Firewalls

Each AP E-cert component is protected by a firewall. The firewall is set up in such a way that only traffic essential to AP E-cert operation is allowed. All other Internet traffic is blocked. Data from and to the AP E-cert servers is encrypted. The firewalls contain monitoring software to detect break-ins.

1.6.2. Authentication

Each AP E-cert user has their individual username and password for access. No public access is allowed. Usernames and passwords are never transmitted over the Internet in clear text, as all data is encrypted.

1.6.3. Data Encryption

All web access to the AP E-cert system is fully encrypted using industry-standard 128 bit SSL encryption. The same kind of encryption is employed for secure exchange of data over SOAP.

1.6.4. Additional Security Measures

For each user only data relevant to the user will be accessible and roles defined within the system will limit transactions users can carry out.



2. Business Rules

The following business rules have been incorporated into the SOAP service for XML and SOAP transaction methods:

ID	Description
BR1	The service is available continuously in the Live and Training environments, with the exception of scheduled maintenance times.
BR2	<p>A new XML format has been introduced for Submissions (including raise and amend). Each file contains those elements required for successful submission of an Eligibility Document or Export Certificate.</p> <p>Submission files are defined by the DTD and comply with the UN/CEFACT standard for E-cert.</p>
BR3	In addition to the DTD that defines the data elements, names, relationships and optionality of data required for submission, AP E-cert (itself) has a number of business rules and data validation checks that are tested.
BR4	The service only returns results (shoulder numbers, document or certificate details) for which the SOAP user holds access permission.
BR5	<p>The service will issue error messages.</p> <p>An error is returned when there are no records matching the request, or the request was not able to be performed due to poor XML format or E-cert data validation errors.</p> <p>For example,</p> <p>Processing file ==> ./NZL2007-ME800-16T.xml</p> <p><code>\$VAR1 = {'Fault' => {'faultcode' => 'SOAP-INV: Server', 'faultstring' => 'Error A certificate with this number already exists. The certificate is not in Resubmit status, so it cannot be updated via batch' } };</code></p>
BR6	Response messages are initiated in real time upon processing the request.
BR7	The sequence of tags within the XML file must follow that defined by the DTD.
BR8	Tags within XML files are case sensitive. Opening and closing tags must match.
BR9	For amending certificate documents the existing online business rules apply.
BR10	For cancelling certificate documents the existing online business rules apply.



3. SOAP Client Connectivity

3.1. Introduction

The choice of SOAP Client is up to the developer, there are a number of SOAP development tools available.

3.2. Connectivity

E-cert via SOAP can only be accessed over an HTTPS connection to the AP E-cert SOAP servers.

3.2.1. Server Locations

Environment	Location
Training	https://ectrain.maf.govt.nz/soap-ectrain
Production	https://sancrt.maf.govt.nz/soap-eclive

3.2.2. User Authentication

Standard web server authentication is used to provide the mandatory user and password details. SOAP access for a particular user requires that the user has accepted the E-cert terms and conditions on the main E-cert web site.

A SOAP envelope must be created to enable connectivity to be tested. The client must include the following credentials:

- User Name
- Password
- URL

3.2.3. Usage

Restrictions on usage apply to prevent over-usage and abuse. For details refer to the AP E-cert Terms & Conditions on the main E-cert web site.

3.2.4. WSDL

In section 3.3 two WSDL files are provided for connecting to Training and Production, respectively.

These WSDL files provide a standard description of E-cert SOAP services.

Many programming languages now understand WSDL and can use this file to automatically invoke E-cert's API.

3.2.5. Error Handling

Standard SOAP faults are returned if any errors are encountered during the processing of the SOAP request. These may be translated to exceptions by the programming environment of the SOAP client.



3.2.6. Connectivity Testing

Once the SOAP client has established connectivity with the E-cert, the SOAP server can be tested by using the following ping command.

Ping

Description	Simple connectivity test. Returns the string 'pong' if successful.
Parameters	None
Results	String
Access	All users
Example	<pre>my \$result = \$soap->ping(); \$VAR1 = { 'pingResponse' => { 's-gensym15' => 'pong' } }; Done</pre>

3.3. Web Service Definitions

The following WSDL files are found on the below URLs.

3.3.1. E-cert Live – <https://sancrt.maf.govt.nz/APEcertServiceLive/APEcertLive.wsdl>

3.3.2. E-cert Training - <https://ectrain.maf.govt.nz/APEcertServiceTrain/APEcertTrain.wsdl>



4. XML and SOAP Interface Methods

4.1. Introduction

This section documents the methods that are currently supported by XML and SOAP in the AP E-cert context.

Methods are available to particular sector groups (NZ Industry, Border Control Agencies of Importing Countries)

There are five types of methods currently defined. These are **Search, Retrieve, Submission, Status change and Update.**

Search methods provide the ability to find either eligibility documents or export certificates by update date or update date and status. The SOAP request returns the shoulder numbers of qualifying records.

New find methods have been introduced. The SOAP request returns the following extra details; Update Date, Status Name, Submission Date and the Submitter. The new find method has also been developed to work for Third Party requests.

Retrieve methods provide the ability to obtain the specific detail associated with an export certificate or eligibility document. A shoulder number is passed with a retrieve method and the data returned is either the official certificate XML associated with the record OR the submission data view (that is the view of the record as it was submitted).

Submission methods provide the means for Industry Users to create Eligibility Documents or Export Certificates via SOAP (rather than on-line).

Amend methods provide the means for industry to amend some part of the Export Certificate or some part of an Eligibility Documents, including non exhausted product data via SOAP.

Cancel - Status Change methods will allow industry to cancel and Export Certificate or an Eligibility Document and supply a reason via SOAP.

Update methods currently provide the means for Overseas Border Inspection Agencies to acknowledge the receipt of a consignment or to record the outcome of the inspection action.

4.2. Search Methods

4.2.1. find_eligibility_documents_by_update_date_and_status

Description	<p>Finds eligibility documents that have been changed on the date that is > than from_date and <= to_date and are currently in the given status.</p> <p>Results are returned as list of shoulder numbers representing a 'page' of results.</p> <p>The first page is page 0. The last result on the last page is set to the string 'END'.</p> <p>If multiple pages are available (ie the page does not show an 'END' tag) then this method can be called subsequent times increasing the page parameter until all pages have been retrieved.</p> <p>The current maximum number of records returned by this query is 1000.</p>
Parameters	<p>integer: page (required) string: from_date (required) string: to_date (required)</p>



	<p>string: status (required)</p> <p><i>status</i> can be one of the following values 'RAISED','RESUBMIT','AMENDED','APPROVED','REJECTED','CANCELLED','REQUEST_AMENDMENT','TO_BE_AMENDED'</p>
Results	array of strings: 1 page of results
Access	NZ Industry, the result set is restricted to shoulder numbers of those eligibility documents the SOAP user has permission to access.
Example	<pre>my \$result = \$soap- >find_eligibility_documents_by_update_date_and_status(0,'2007-10-01 00:00:00','2007-10-31 00:00:00','APPROVED');</pre> <p><i>Returns:</i> \$VAR1 = { 'find_eligibility_documents_by_update_date_and_statusResponse' => { 'Array' => ['NZL2007/ME800/14T', 'NZL2007/ME800/15T', 'NZL2007/ME800/17T'...</p>

4.2.2. find_eligibility_documents_by_update_date

Description	<p>Finds eligibility documents that have been changed on the date that is > than <i>from_date</i> and <= <i>to_date</i></p> <p>Results are returned as a list of shoulder numbers representing a 'page' of results.</p> <p>The first page is page 0. The last result on the last page is set to the string 'END'.</p> <p>If multiple pages are available (ie the page does not show an 'END' tag) then this method can be called subsequent times increasing the page parameter until all pages have been retrieved.</p> <p>The current maximum number of records returned by this query is 1000.</p>
Parameters	<p>integer: page (required) string: <i>from_date</i> (required) string: <i>to_date</i> (required)</p>
Results	array of strings: 1 page of results
Access	NZ Industry, the result set is restricted to shoulder numbers of those eligibility documents the SOAP user has permission to access.
Example	<pre>my \$result = \$soap->find_eligibility_documents_by_update_date(0,'2007-10-01 00:00:00','2007-10-31 00:00:00');</pre> <p><i>Returns:</i> \$VAR1 = {'find_eligibility_documents_by_update_dateResponse' => { 'Array' => ['NZL2007/ME282/354113T', 'NZL2007/ME800/14T', 'NZL2007/ME727/15T'...</p>

4.2.3. find_eligibility_documents_details_by_update_date

Description	<p>Finds eligibility documents that have been changed on the date that is > than <i>from_date</i> and <= <i>to_date</i></p> <p>Results are returned as a list, each line includes:</p>
--------------------	--



	<p>Shoulder Number, Update Date, Status Name, Submission Date and Submitter. Representing a 'page' of results.</p> <p>The first page is page 0. The last page of results is reflected by more_results="No".</p> <p>If multiple pages are available (ie the more_results= "Yes") then this method can be called subsequent times increasing the page parameter until all pages have been retrieved</p> <p>The current maximum number of records returned by this query is 1000.</p>
Parameters	<p>integer: page (required) string: from_date (required) string: to_date (required)</p>
Results	Base64Binary
Access	NZ Industry, the result set is restricted to shoulder numbers of those eligibility documents the SOAP user has permission to access.
Example	<pre>my \$result = \$soap- >find_eligibility_documents_details_by_update_date(0,'2007-10-01 00:00:00','2007-10-31 00:00:00');</pre> <p><i>Returns:</i></p> <pre>\$VAR1 = '<find_eligibility_documents_details_by_update_date more_results="No"> <eligibility_document_details> <shoulder_no>NZL2008/PHOENIX1/549356T</shoulder_no> <update_date>2008-05-06T10:39:04</update_date> <status_id>Approved</status_id> <submitted_time>2008-04-16T12:34:59</submitted_time> <submitted_name>lulu lala</submitted_name> </eligibility_document_details> <eligibility_document_details> <shoulder_no>NZL2008/PHOENIX1/859663T</shoulder_no> <update_date>2008-05-12T10:44:55</update_date> <status_id>Raised</status_id> <submitted_time>2008-05-12T10:44:55</submitted_time> <submitted_name>lulu lala</submitted_name> </eligibility_document_details> <eligibility_document_details> <shoulder_no>NZL2008/PHOENIX1/859603T</shoulder_no> <update_date>2008-05-21T08:50:55</update_date> <status_id>Raised</status_id> <submitted_time>2008-05-21T08:50:55</submitted_time> <submitted_name>lulu lala</submitted_name> </eligibility_document_details> </find_eligibility_documents_details_by_update_date></pre>

4.2.4. find_third_party_eligibility_documents_details_by_update_date

Description	<p>Finds third party eligibility documents that have been changed on the date that is > than from_date and <= to_date</p> <p>Results are returned as a list, each line includes: Shoulder Number, Update Date, Status Name, Submission Date and Submitter. Representing a 'page' of results.</p> <p>The first page is page 0. The last page of results is reflected by more_results="No".</p>
--------------------	---



	<p>If multiple pages are available (ie the more_results= "Yes") then this method can be called subsequent times increasing the page parameter until all pages have been retrieved</p> <p>The current maximum number of records returned by this query is 1000.</p>
Parameters	<p>integer: page (required) string: from_date (required) string: to_date (required)</p>
Results	Base64Binary
Access	NZ Industry, the result set is restricted to shoulder numbers of those eligibility documents the SOAP user has permission to access.
Example	<pre>my \$result = \$soap- >find_third_party_eligibility_documents_details_by_update_date(0,'2007-10-01 00:00:00','2007-10-31 00:00:00');</pre> <p><i>Returns:</i></p> <pre>soap_result = '<find_third_party_eligibility_documents_details_by_update_date more_results="No"> <eligibility_document_details> <shoulder_no>NZL2001/ME17/410T</shoulder_no> <update_date>2007-03-08T13:31:55</update_date> <status_id>Amended</status_id> <submitted_time>2001-11-28T18:14:09</submitted_time> <submitted_name>sammy smith</submitted_name> </eligibility_document_details> <eligibility_document_details> <shoulder_no>NZL2007/ME21/174301T</shoulder_no> <update_date>2007-08-31T12:43:57</update_date> <status_id>Raised</status_id> <submitted_time>2007-08-31T12:43:57</submitted_time> <submitted_name>sammy smith</submitted_name> </eligibility_document_details> <eligibility_document_details> <shoulder_no>NZL2007/EXPORTER1/676005T</shoulder_no> <update_date>2007-10-30T18:16:14</update_date> <status_id>Raised</status_id> <submitted_time>2007-10-30T18:16:14</submitted_time> <submitted_name>Xena Warrior</submitted_name> </eligibility_document_details> </find_third_party_eligibility_documents_details_by_update_date></pre>

4.2.5. find_certificates_by_update_date_and_status

Description	<p>Finds certificates that have been changed on the date that is > than from_date and <= to_date and are currently in the given status.</p> <p>Results are returned as a list of shoulder numbers representing a 'page' of results.</p> <p>The first page is page 0. The last result on the last page is set to the string 'END'.</p> <p>If multiple pages are available (ie the page does not show an 'END' tag) then this method can be called subsequent times increasing the page parameter until all pages have been retrieved.</p> <p>The current maximum number of records returned by this query is 1000.</p>
--------------------	--



Parameters	<p>integer: page (required) string: from_date (required) string: to_date (required) string: status (required)</p> <p><i>NZ Industry status</i> can be one of the following values 'RAISED','RESUBMIT','AMENDED','APPROVED','REJECTED','CANCELLED',' REQUEST_REPLACEMENT','TO_BE_REPLACED','REPLACED','REVOKED',' REPLACEMENT_AUTHORISED','ACCEPTED','DETAINED','REQUEST_AMEN DMENT','TO_BE_AMENDED','ACKNOWLEDGED'</p> <p><i>Overseas Authority status</i> can be one of the following values 'APPROVED','REJECTED','REQUEST_REPLACEMENT','TO_BE_REPLACED',' 'REPLACED','REVOKED','REPLACEMENT_AUTHORISED','ACCEPTED','DET AINED','ACKNOWLEDGED'</p>
Results	array of strings: 1 page of results
Access	Overseas Authorities and NZ Industry, the result set is restricted to shoulder numbers of those Export Certificates the SOAP user has permission to access.
Example	<pre>my \$result = \$soap->find_certificates_by_update_date_and_status(0,'2007-10-01 00:00:00','2007-10-31 00:00:00','APPROVED');</pre> <p><i>Returns:</i> \$VAR1 = {'find_certificates_by_update_date_and_statusResponse' => {'Array' => ['NZL2007/NZFSA1/12', 'NZL2007/NZFSA1/15', 'NZL2007/NZFSA1/19'...</p>

4.2.6. find_certificates_by_update_date

Description	<p>Finds certificates that have been changed on the date that is > than from_date and <= to_date.</p> <p>Results are returned a list of shoulder numbers representing a 'page' of results.</p> <p>The first page is page 0. The last result on the last page is set to the string 'END'.</p> <p>If multiple pages are available (ie the page does not show an 'END' tag) then this method can be called subsequent times increasing the page parameter until all pages have been retrieved.</p> <p>The current maximum number of records returned by this query is 1000.</p>
Parameters	<p>integer: page (required) string: from_date (required) string: to_date (required)</p>
Results	array of strings: 1 page of results
Access	Overseas Authorities and NZ Industry, the result set is restricted to shoulder numbers of those Export Certificates the SOAP user has permission to access.
Example	<pre>my \$result = \$soap->find_certificates_by_update_date(0,'2007-10-01 00:00:00','2007-10-31 00:00:00');</pre> <p><i>Returns:</i> \$VAR1 = {'find_certificates_by_update_dateResponse' => { 'Array' => ['NZL2007/NZFSA1/8',</p>



	'NZL2007/NZFSA1/9'...
--	-----------------------

4.2.7. find_export_certificates_detail_by_update_date

Description	<p>Finds certificates that have been changed on the date that is > than from_date and <= to_date.</p> <p>Results are returned as a list, each line includes: Shoulder Number, Update Date, Status Name, Submission Date and Submitter. Representing a 'page' of results.</p> <p>The first page is page 0. The last page of results is reflected by more_results="No".</p> <p>If multiple pages are available (ie the more_results= "Yes") then this method can be called subsequent times increasing the page parameter until all pages have been retrieved</p> <p>The current maximum number of records returned by this query is 1000.</p>
Parameters	<p>integer: page (required) string: from_date (required) string: to_date (required)</p>
Results	Base64Binary
Access	Overseas Authorities and NZ Industry, the result set is restricted to shoulder numbers of those Export Certificates the SOAP user has permission to access.
Example	<pre>my \$result = \$soap->find_export_certificates_details_by_update_date(0,'2007-10-01 00:00:00','2007-10-31 00:00:00');</pre> <p><i>Returns:</i> /soap]\$ \$VAR1 = '<find_export_certificates_details_by_update_date more_results="No"> <export_certificate_details> <shoulder_no>NZL2008/EX1456/51401</shoulder_no> <update_date>2008-08-01T11:37:20</update_date> <status_id>Amended</status_id> <submitted_time>2008-07-10T14:59:15</submitted_time> <submitted_name>Xena Warrior</submitted_name> </export_certificate_details> <export_certificate_details> <shoulder_no>NZL2008/EX1456/516101</shoulder_no> <update_date>2008-08-01T12:35:39</update_date> <status_id>Amended</status_id> <submitted_time>2008-06-17T22:20:12</submitted_time> <submitted_name>Xena Warrior</submitted_name> </export_certificate_details> <export_certificate_details> <shoulder_no>NZL2008/PHOENIX1/494847</shoulder_no> <update_date>2008-08-04T11:30:22</update_date> <status_id>Amended</status_id> <submitted_time>2008-07-04T10:50:37</submitted_time> <submitted_name>luilu laal</submitted_name> </export_certificate_details> </find_export_certificates_details_by_update_date>' </p>

4.2.8. find_third_party_export_certificates_detail_by_update_date

Description	Finds third party certificates that have been changed on the date that is > than from_date and <= to_date.
--------------------	--



	<p>Results are returned as a list, each line includes: Shoulder Number, Update Date, Status Name, Submission Date and Submitter. Representing a 'page' of results.</p> <p>The first page is page 0. The last page of results is reflected by more_results="No".</p> <p>If multiple pages are available (ie the more_results= "Yes") then this method can be called subsequent times increasing the page parameter until all pages have been retrieved</p> <p>The current maximum number of records returned by this query is 1000.</p>
Parameters	<p>integer: page (required) string: from_date (required) string: to_date (required)</p>
Results	Base64Binary
Access	NZ Industry, the result set is restricted to shoulder numbers of those Export Certificates the SOAP user has permission to access.
Example	<pre>my \$result = \$soap- >find_third_party_export_certificates_detail_by_update_date(0,'2007-10-01 00:00:00','2007-10-31 00:00:00');</pre> <p><i>Returns:</i></p> <pre>\$soap_result = '<find_third_party_export_certificates_details_by_update_date more_results="No"> <export_certificate_details> <shoulder_no>NZL2006/EX1456/516102</shoulder_no> <update_date>2007-01-11T11:47:14</update_date> <status_id>Detained</status_id> <submitted_time>2006-11-23T11:34:39</submitted_time> <submitted_name>Xena Warrior</submitted_name> </export_certificate_details> <export_certificate_details> <shoulder_no>NZL2007/EX1456/669910</shoulder_no> <update_date>2007-08-10T14:46:32</update_date> <status_id>Raised</status_id> <submitted_time>2007-08-10T14:46:32</submitted_time> <submitted_name>Xena Warrior</submitted_name> </export_certificate_details> <export_certificate_details> <shoulder_no>NZL2007/ME85/29901</shoulder_no> <update_date>2007-09-19T12:27:38</update_date> <status_id>Approved</status_id> <submitted_time>2007-09-19T12:12:39</submitted_time> <submitted_name>sammy smith</submitted_name> </export_certificate_details> <export_certificate_details> <shoulder_no>NZL2007/ME85/29902</shoulder_no> <update_date>2007-09-19T14:05:09</update_date> <status_id>Raised</status_id> <submitted_time>2007-09-19T14:05:09</submitted_time> <submitted_name>sammy smith</submitted_name> </export_certificate_details> </find_third_party_export_certificates_details_by_update_date>'</pre>



4.3. Retrieve Data Methods

4.3.1. get_eligibility_document_xml

Description	Returns an XML string for a single eligibility document. The data returned is a view that represents the submission input for the Eligibility Document. This XML data conforms to the ECEDEX data exchange standard defined in nzfsa-edsub*.dtd.
Parameters	string: shoulder_no (required)
Results	string: submission XML of the eligibility document
Access	NZ Industry, the data will only be returned if the user has permission to the record.
Example	my \$result = \$soap->get_eligibility_document_xml('NZL2007/FPH200/14T'); Returns XML data for NZL2007/FPH200/14T

4.3.2. get_multiple_eligibility_document_xml

Description	Returns the XML of multiple eligibility documents as a single XML string. The data returned is a view that represents the submission input for the Eligibility Documents returned. This XML data conforms to the ECEDEX data exchange standard defined in nzfsa-edsub*.dtd.
Parameters	array of strings: shoulder_nos (required)
Results	string: submission XML of 1 or more eligibility documents
Access	NZ Industry, only those records the user has permission to view will be returned.
Example	my \$result = \$soap->get_multiple_eligibility_document_xml(['NZL2007/ME200/354113T', 'NZL2007/ME800/14T', 'NZL2007/ME800/15T']); Returns XML data for 'NZL2007/ME200/354113T', 'NZL2007/ME800/14T', and 'NZL2007/ME800/15T'

4.3.3. get_eligibility_document_process_history_xml

Description	Returns an XML string for a single eligibility document which includes the process history of any source ED. The data returned is a view that represents the online view for the Eligibility Document. This XML data conforms to the ECEDEX data exchange standard defined in nzfsa-edsub*.dtd.
Parameters	string: shoulder_no (required)
Results	string: submission XML of the eligibility document
Access	NZ Industry, the data will only be returned if the user has permission to the record.
Example	my \$result = \$soap->get_eligibility_document_process_history_xml('NZL2007/ME811/14T'); Returns XML data for NZL2007/ME811/14T

**4.3.4. get_export_certificate_xml**

Description	Returns an XML string for a single export certificate. The data returned is a view that represents the submission input for the Export Certificate. This XML data conforms to the UN/CEFACT standard defined in nzfsa-ecsub*.dtd.
Parameters	string: shoulder_no (required)
Results	string: submission XML of the export certificate
Access	NZ Industry, the record will only be returned if the user has permission to view it.
Example	my \$result = \$soap->get_export_certificate_xml('NZL2007/NZFSA1/11'); Returns XML data for 'NZL2007/NZFSA1/11'

4.3.5. get_multiple_export_certificate_xml

Description	Returns the XML of multiple export certificates as a single XML string. The data returned is a view that represents the submission input for the Export Certificates returned. This XML data conforms to the UN/CEFACT data exchange standard defined in nzfsa-ecsub*.dtd.
Parameters	array of strings: shoulder_nos (required)
Results	string: submission XML of 1 or more export certificates
Access	NZ Industry, only those records the user has permission to view will be returned.
Example	my \$result = \$soap->get_multiple_export_certificate_xml(['NZL2007/NZFSA1/8', 'NZL2007/NZFSA1/9']); Returns XML data for 'NZL2007/NZFSA1/8', and 'NZL2007/NZFSA1/9'

4.3.6. get_official_certificate_xml

Description	Returns an XML string for a single certificate. The data returned is a view that represents the official certificate output for the Export Certificate returned. This XML data conforms to the UN/CEFACT data exchange standard defined in ecedex-export*.dtd (2.9.1 is the current version)
Parameters	string: shoulder_no (required)
Results	string: official XML of the certificate
Access	Overseas Authorities and NZ Industry, the record will only be returned if the user has permission to view the record.
Example	my \$result = \$soap->get_official_certificate_xml('NZL2007/NZFSA1/8'); Returns XML data for 'NZL2007/NZFSA1/8'

4.3.7. get_multiple_official_certificate_xml

Description	Returns the XML of multiple certificates as a single XML string.
--------------------	--



	<p>The data returned is a view that represents the official certificate output for the Export Certificate returned.</p> <p>This XML data conforms to the UN/CEFACT data exchange standard defined in <i>ecedex-export*.dtd (2.9.1 is the current version)</i></p>
Parameters	array of strings: shoulder_nos (required)
Results	string: official XML of 1 or more certificates
Access	Overseas Authorities and NZ Industry, only those records the user has permission to view will be returned.
Example	<pre>my \$result = \$soap->get_multiple_official_certificate_xml(['NZL2007/NZFSA1/11', 'NZL2007/NZFSA1/12', 'NZL2007/NZFSA1/15', 'NZL2007/TEST1/19']);</pre> <p>Returns XML data for 'NZL2007NZFSA1/11', 'NZL2007/ NZFSA1/12', 'NZL2007/ NZFSA1/15', and 'NZL2007/TEST1/19'</p>

4.4. Submission Methods

4.4.1. raise_eligibility_document

Description	<p>Raises a single eligibility document using the XML supplied.</p> <p>XML must conform to the ECEDEX data exchange standard <i>nzfsa-edsub*.dtd (1.0 is the current version)</i>.</p>
Parameters	string: XML of the eligibility document (required)
Results	string: result
Access	NZ Industry, user must have permission to raise the eligibility document
Example	See Section 6 for detail.

4.4.2. resubmit_eligibility_document

Description	<p>Resubmits a single eligibility document using the XML supplied.</p> <p>The eligibility document must exist in the AP E-cert and be in status <i>Resubmit</i>.</p> <p>XML must conform to the ECEDEX data exchange standard <i>nzfsa-edsub*.dtd (1.0 is the current version)</i>.</p>
Parameters	string: XML of the eligibility document (required)
Results	string: result
Access	NZ Industry, user must have permission to resubmit the eligibility document
Example	The resubmission of an eligibility document is managed from the eligibility document submission dtd. See Section 6 for detail

4.4.3. exhaust_eligibility_document

Description	Exhausts all products on a single eligibility document
Parameters	string: shoulder_no (required)



Results	string: result Returns the string 'eligibility document <shoulder_no> exhausted' if successful.
Access	NZ Industry, user must have permission to exhaust the eligibility document
Example	my \$result = \$soap->exhaust_eligibility_document('NZL2007/FPH2001/15T'); <i>Returns:</i> \$VAR1 = { 'exhaust_eligibility_documentResponse' => { 's-gensym41' => 'eligibility document NZL2007/FPH200/15T exhausted' }}

4.4.4. exhaust_eligibility_document_products

Description	Exhausts one or more products on an eligibility document.
Parameters	string: shoulder_no (required) array of strings: product_nos (required)
Results	string: result Returns the string 'eligibility document <shoulder_no>, product(s) <product_nos> exhausted' if successful.
Access	NZ Industry, user must have permission to exhaust one or more products on the eligibility document.
Example	my \$result = \$soap->exhaust_eligibility_document_products('NZL2007/ME800/16T', [1,3]); Exhausts products 1 and 3 on 'NZL2007/ME800/16T'

4.4.5. amend_eligibility_document

Description	Amends a single eligibility document using the XML supplied. The eligibility document must exist in the AP E-cert and be in status of : <ul style="list-style-type: none"> • Raised • To be Amended • Amended • Resubmit XML must conform to the ECEDEX data exchange standard nzfsa-edsub*.dtd (1.0 is the current version).
Parameters	string: XML of the eligibility document (required)
Results	string: result
Access	NZ Industry, user must have permission to amend the eligibility document
Example	The amendment of an eligibility document is managed from the eligibility document submission dtd. See Section 6 for detail

4.4.6. raise_export_certificate

Description	Raises a single export certificate using the XML supplied. XML must conform to the ECEDEX data exchange standard nzfsa-ecsub*.dtd (1.0 is the current version)
Parameters	string: XML of the export certificate (required)
Results	string: result



Access	NZ Industry, user must have permission to raise the export certificate
Example	See Section 7 for Detail

4.4.7. resubmit_export_certificate

Description	Resubmits a single export certificate using the XML supplied. The export certificate must exist in AP E-cert and be in status <i>Resubmit</i> . XML must conform to the ECEDEX data exchange standard nzfsa-ecsub*.dtd (1.0 is the current version)
Parameters	string: XML of the export certificate (required)
Results	string: result
Access	NZ Industry, user must have permission to resubmit the export certificate
Example	The resubmission of an export certificate is managed from the export certificate submission dtd. See Section 7 for Detail

4.4.8. amend_export_certificate

Description	Amend a single export certificate using the XML supplied. The export certificate must exist in AP E-cert and be in status <ul style="list-style-type: none"> • Raised • To be Amended • Amended • Resubmit XML must conform to the ECEDEX data exchange standard nzfsa-ecsub*.dtd (1.0 is the current version)
Parameters	string: XML of the export certificate (required)
Results	string: result
Access	NZ Industry, user must have permission to resubmit the export certificate
Example	The resubmission of an export certificate is managed from the export certificate submission dtd. See Section 7 for Detail

4.5. Change Status Methods

4.5.1. accept_certificate

Description	Change the status a certificate to Accepted
Parameters	string: shoulder_no (required) string: reason (optional)
Results	string: result Returns the string 'certificate accepted' if successful.
Access	Overseas Authorities, user must have permission to view the certificate
Example	my \$result = \$soap->accept_certificate('NZL2007/NZFSA1/19', 'This is a reason'); <i>Returns:</i> \$VAR1 = { 'accept_certificateResponse' => { 's-gensym93' => 'certificate accepted' } };



4.5.2. detain_certificate

Description	Change the status a certificate to Detained.
Parameters	string: shoulder_no (required) string: reason (required)
Results	string: result Returns the string 'certificate detained' if successful.
Access	Overseas Authorities, user must have permission to view the certificate
Example	<pre>my \$result = \$soap->detain_certificate(NZL2007/NZFSA1/19', 'This is a reason'); Returns: \$VAR1 = { 'detain_certificateResponse' => { 's-gensym92' => 'certificate detained' } };</pre>

4.5.3. reject_certificate

Description	Change the status a certificate to Rejected.
Parameters	string: shoulder_no (required) string: reason (required)
Results	string: result Returns the string 'certificate rejected' if successful.
Access	Overseas Authorities, user must have permission to view the certificate
Example	<pre>my \$result = \$soap->reject_certificate(NZL2007/NZFSA1/19', 'This is a reason'); Returns: \$VAR1 = { 'reject_certificateResponse' => { 's-gensym94' => 'certificate rejected' } };</pre>

4.5.4. request_replacement_certificate

Description	Change the status a certificate to request replacement.
Parameters	string: shoulder_no (required) string: reason (required)
Results	string: result Returns the string 'certificate replacement requested' if successful.
Access	Overseas Authorities, user must have permission to view the certificate
Example	<pre>my \$result = \$soap->request_replacement_certificate(NZL2007/NZFSA1/19', 'This is a reason'); Returns: \$VAR1 = { 'request_replacement_certificateResponse' => { 's-gensym96' => 'certificate replacement requested' } };</pre>

**4.5.5. acknowledge_certificate**

Description	Change the status a certificate to Acknowledged.
Parameters	string: shoulder_no (required) string: reason (optional)
Results	string: result Returns the string 'certificate acknowledged' if successful.
Access	Overseas Authorities, user must have permission to view the certificate
Example	<pre>my \$result = \$soap->acknowledge_certificate(NZL2007/NZFSA1/19', 'This is a reason'); Returns: \$VAR1 = {'acknowledge_certificateResponse' => {'s-gensym45' => 'certificate acknowledged' } };</pre>

4.6. Cancel Methods**4.6.1. cancel_eligibility_document**

Description	Change the status of an Eligibility Document to Cancelled.
Parameters	string: shoulder_no (required) string: reason (Mandatory)
Results	string: result Returns the string 'Eligibility Document Cancelled' if successful.
Access	Industry users must have permission to cancel eligibility documents.
Example	<pre>my \$result = \$soap->cancel_eligibility_document(NZL2007/NZFSA1/19', 'This is a reason'); Returns: \$VAR1 = { 'cancel_eligibility_documentResponse' => { 's-gensym92' => 'eligibility document cancelled' } };</pre>

4.6.2. cancel_export_certificate

Description	Change the status of the export certificate to Cancelled
Parameters	string: shoulder_no (required) string: reason (Mandatory)
Results	string: result Returns the string 'Export Certificate Cancelled' if successful.
Access	Industry User must have permission to Cancel export certificates
Example	<pre>my \$result = \$soap->Cancel_Export)_certificate('NZL2007/NZFSA1/19', 'This is a reason'); Returns: \$VAR1 = { 'Cancel_Export_certificateResponse' => { 's-gensym93' => 'certificate Cancelled' } };</pre>

5. AP E-cert XML Submission Data Standards

The following AP E-cert XML submission Data Standards are based on the requirements for a submission of an xml to raise either an Eligibility Document or an Export Certificate.

5.1. Eligibility Document Submission Data Standards

5.1.1. Wrapper

Every Eligibility Document XML file processed by SOAP must be encased within the following format

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<EligibilityDocument>
```

Header and product/process data elements, subelements and attributes go here....

```
</EligibilityDocument>
```

A = Alphabetic, **N** = Numeric, **AN** = Alpha-Numeric, **Date** = Date field, **Y/N** = Yes/no flag
Numbers in brackets denote field length, eg AN(8) is up to 8 characters long; unl = unlimited.

Fields on blue background are mandatory, e.g. Consignee

5.1.2. ED Header Submission Data

Element (SubElement) • Attribute	Occurrence Required for XML SOAP submission	ED Definition in AP E-cert	AP E-cert Format/ Values	XML Batch File Tag
EligibilityDocument • number	Required	Unique serial number within the calendar year used as last component in shoulder number.	AN(6)	<EligibilityDocument number ="NZL2009/TEST604/123456T"> OR <EligibilityDocument number ="123456">
EligibilityDocument • exhaustion	Optional	Indicator whether the entire product quantity of each product item has been used on related outgoing documents.	AN(3) YES,NO	<EligibilityDocument exhaustion ="YES" number ="NZL2009/TEST604/123456T ">
Auto-approval (Form) • code	Optional	Indicator whether eligibility document is to be auto approved, must use code = "AA".	Alphanumeric(2) "AA"	<Form code ="AA"/>

Element (SubElement) • Attribute	Occurrence Required for XML SOAP submission	ED Definition in AP E-cert	AP E-cert Format/ Values	XML Batch File Tag
Departure(Date)	Required	Date of departure from the consigning or final premises. If desirable Departure Date can be provided using attributes <ul style="list-style-type: none"> ▪ Day ▪ Month ▪ Year 	Date YYYY-MM-DD	<Departure> <Date>2009-12-01</Date> </Departure>
Unofficial Commercial Information (Comment)	Optional	Free text field for commercial information	AN (2000)	<Comment>Unofficial commercial information</Comment>
Consignee • identifier	Required	Identifier of party to which goods are consigned In AP E-cert Consignee Identifier is mandatory and therefore must be included in the XML file. Must be a valid premises or Exporter ID.	AN(10) Valid Premise	<Consignee identifier="NZFSA1"/>
FinalDestination	Optional	Location to which the consignment is sent. Free text field which you can put the UN/LOCODE code into. UN/LOCODE will get converted to City, Country format on submission	AN(254) OR A(5) UNLOCODE	<FinalDestination>Punta Gorda, Belize</FinalDestination> OR <FinalDestination>BZPND</FinalDestination>
Consignor • identifier	Required	License number for consignor (party legally liable for goods) issued by competent authority. In AP E-cert Consignor Identifier is mandatory and therefore must be included in the XML file. Must be a valid premises or Exporter ID.	AN(10) Valid Premise	<Consignor identifier="TEST604"/>
ThirdParty • identifier	Optional Repeatable	Read-only access to the certificate is granted to all users who can read certificates for the IDs specified.	AN(254) Multiple Valid Premise allowed.	<ThirdParty identifier="ME86"/>

Element (SubElement) • Attribute	Occurrence Required for XML SOAP submission	ED Definition in AP E-cert	AP E-cert Format/ Values	XML Batch File Tag
LoadingPort OR LoadingPort • code	Optional	Location where the consignment is loaded Free text field or Code. If code of port, airport or other location to which a consignment was loaded from then must be a UN/LOCODE as per ISO 3166 compliant. Whilst Defined as Repeatable on the DTD. Only one loading port is accepted by AP E-cert.	AN(254) OR A(5) As per UNLOCODE	<LoadingPort>Napier</LoadingPort> OR <LoadingPort code="NZNPE"/>
Transport (Mode) • code	Optional	Required on all EDs other than internal transfers where the consignor and consignee are the same ID. EDs to exporters must choose either 1 or 4. Air may be used for road consignments sent directly to an airport coldstore intended for airfreight shipment.	N(1) 1 Maritime 2 Rail 3 Road 4 Air	<Transport> <Mode code="2"/> <ConveyanceRef>Conveyance Ref</ConveyanceRef> <Carrier> <Name>Carrier Name</Name> </Carrier> </Transport>
Transport (ConveyanceRef)	Optional	Transport means journey identifier. To identify a journey of a means of transport, for example voyage number, flight number, trip number.	AN(254)	
Transport Carrier (Name)	Optional	Required for edible meat and seafood products, optional for others. Transfers to exporters may additionally have the vessel or airline name. The full name or listed ID of the transport company or "Operator Vehicle"	AN(254)	

5.1.3. ED Header Example

XML File

```
<EligibilityDocument exhaustion="NO" number="NZL2009/TEST604/123456T">
<Form code="AA">Auto-Approval</Form>
  <Departure>
    <Date>2009-12-01</Date>
  </Departure>
  <Comment>Unofficial Commercial Information</Comment>
  <Consignee identifier="NZFSA1"/>
  <FinalDestination>Punta Gorda, Belize</FinalDestination>
  <Consignor identifier="TEST604"/>
  <ThirdParty identifier="ME86"/>
  <LoadingPort code="NZNPE"/>
  <Transport>
    <Mode code="2"/>
    <ConveyanceRef>Conveyance Ref</ConveyanceRef>
    <Carrier>
      <Name>Carrier Name</Name>
    </Carrier>
  </Transport>
```

5.1.4. ED Product and Process Submission Data

Element (SubElement) • Attribute	Occurrence Required for XML SOAP submission	ED Definition in AP E-cert	Format/ Values	Batch File Tag
Product • item	Required Repeatable	Goods item sequence identifier. Sequence number differentiating a specific goods item entry of a consignment. Must precede each set of product item fields in the batch file.	N(2) 1...99	<Product item="1">
Product • exhaustion	Optional	Indicator whether entire product quantity is used on related outgoing eligibility documents or export certificates	AN(3) YES,NO	<Product exhaustion="YES" item="1">
Description	Required	Plain text description of the nature of the goods sufficient to identify them at the level required for banking, Customs, statistical or transport purposes, avoiding unnecessary detail (Generic term). In AP E-cert Product Description is mandatory and it is a single free text field. The XML file must contain one product description attribute per product	AN(254)	<Description>Description</Description>
Species	Required	Description of the species of the product Free Text In AP E-cert Species is mandatory and therefore must be supplied in the XML File Multiple species can be entered separated by a space.	AN(254)	<Species>Species</Species>
Marks	Optional	Goods item shipping marks text. Free form description of the marks and numbers on a transport unit or package.	AN(254)	<Marks>Item Mark</Marks>
Product (Comment)	Optional	Free form description of information details. Verifiable product item related data and transfer control declarations. Can use repeatable tags for comment, but it does get concatenated in the output,	AN(4000)	<Comment>Product Comments</Comment>

Element (SubElement) • Attribute	Occurrence Required for XML SOAP submission	ED Definition in AP E-cert	Format/ Values	Batch File Tag
Classification (Class) • code	Required	Code specifying a type of goods for Customs, transport or statistical purposes (generic term). One or more class codes are required for each (Class) type code. Multiple codes can be entered separated by a space.	AN(254)	<Classification> <Class code="CH"/> <Type code="DC"/> </Classification>
Classification (Type) • code	Required, Repeatable	Code identifying a user or association maintained code list. Identifies the type of the classification. For example: CC - Country codes which can be repeatable DC - Declaration codes which can be repeatable NZHSC – only one code here is required. etc...	AN(254)	
Package • count	Required	Describes the packaging of the goods, including the number of packages and unit of measure. Quantity/count of the total number of items.	N(10)	<Package count="1" level="1"> <Type code="BG"/> </Package>
Package (Type) • code	Required	Package type code. Code specifying the type of packaging of an item.	AN(2)	
Weight • value	Required, Repeatable	Numeric value of the weight of a product item	D(10,4)	<Weight value="4.0000"> <Type code="AAA"/> <Unit code="KGM"/> </Weight>
Weight (Type) • code	Required	Coded description of the weight type (eg. AAA for Net weight and AAB for Gross Weight- which is optional)	AN(3)	
Weight (Unit) • code	Required	Measurement unit code. Indication of a unit of measurement in which weight (mass), capacity, length, area, volume or other quantity or measurement is expressed.	AN(3)	
Document • number	Optional, Repeatable	Reference identifier. Identifies a reference.	AN(24)	<Document item="1" number="NZL2009/TEST604/6T" outQuantity="1"/>
Document • item	Optional	Goods item sequence identifier. Sequence number differentiating a specific goods item entry of a consignment.	N(2)	
Document • outQuantity	Optional	outQuantity is the number of outgoing packages that are attributable to the named source ED product item.	N(10)	

Element (SubElement) • Attribute	Occurrence Required for XML SOAP submission	ED Definition in AP E-cert	Format/ Values	Batch File Tag
Container • number	Optional	Transport equipment identifier. To identify a piece of transport equipment, e.g. A container or unit load device	AN(254)	<Container number ="Container No" seal ="Seal No"></Container>
Container • seal	Optional	Transport equipment seal identifier. The identification number of a seal affixed to a piece of transport equipment.	AN(254)	
(Processing)Type • code	Required	Processing and storage plants of the product This is mandatory if you don't have a source ED Process type code Code specifying a type of process	AN(3)	<Processing dateOverride="NO"> <Type code ="SLT"/> <StartDate>2009-01-01</StartDate> <EndDate>2009-01-01</EndDate> <Premises identifier ="TEST604"/> </Processing> <Durability> <Date>2010-01-05</Date> </Durability>
Processing • dateOverride	Optional	Indicator set if the process was applied by premises other than the consignor If the dateOverride ="YES" then a processing date has to be entered for submission	AN(3) YES/NO	
Processing (StartDate)	Optional	Event effective start date time Date and or time on which specified processing event becomes effective. In AP E-cert it is mandatory to have either a Start or an End date for some processes (example, slaughter). Expiry date can be entered in as a process and it is mandatory to have a Start or an End Date, Please note that this will then appear as a durability date under the Product, in the Data view.	YYYY-MM-DD (as per ISO8601)	
Processing (EndDate)	Optional	Event effective end date time Date and or time on which a specified processing event becomes no longer effective.	YYYY-MM-DD (as per ISO8601)	
Premises • identifier	Optional	Code specifying the identity of a party. Processing premises are defined in 3035 Party Function Code as code MP (Manufacturing) Unit or GN (Processor)	AN(20)	

5.1.5. ED Product and Process Example

XML File

```
<Product item="1">
  <Description>Description</Description>
  <Species>Species</Species>
  <Marks>Item Mark</Marks>
  <Comment>Product Comments</Comment>
  <Classification>
    <Class code="CH"/>
    <Type code="DC"/></Classification>
  <Classification>
    <Class code="CA"/>
    <Type code="CC"/></Classification>
  <Classification>
    <Class code="020110"/>
    <Type code="NZHSC"/></Classification>
  <Package count="1" level="1">
    <Type code="BG"/>
  </Package>
  <Weight value="2.0000">
    <Type code="AAA"/>
    <Unit code="KGM"/>
  </Weight>
  <Weight value="4.0000">
    <Type code="AAB"/>
    <Unit code="KGM"/>
  </Weight>
  <Document item="1" number="NZL2009/TEST604/6T" outQuantity="1"/>
  <Container number="Container No" seal="Seal No"></Container>
  <Processing dateOverride="NO">
    <Type code="SLT"/>
    <StartDate>2009-01-01</StartDate>
    <EndDate>2009-01-01</EndDate>
    <Premises identifier="TEST604"/>
  </Processing>
</Product>
```

5.2. Export Certificate Submission Data Standards

5.2.1. Introduction

The use of **optional** data elements described below for Export Certificate Submission will depend on the Market Access requirement that is defined on the Certificate Template. When defining the Template MAF has the opportunity to define when data items are not required, remain optional, or become mandatory. Within E-cert itself it is possible to view the XML data requirements of a Template by selecting the **data view** from within the Available Templates page.

5.2.2. Wrapper

Every Export Certificate XML file processed by SOAP must be encased within the following format

```
<?xml version="1.0" encoding="utf-8"?>
```

```
<Certificate>
```

Header and product/process data elements, subelements and attributes go here....

```
</Certificate>
```

A = Alphabetic, **N** = Numeric, **AN** = Alpha-Numeric, **D** = Decimal, **Date** = Date field, **Y/N** = Yes/no flag
Numbers in brackets denote field length, eg AN(8) is up to 8 characters long; unl = unlimited.

Fields on blue background are mandatory, e.g. Departure Date

5.2.3. EC Header Submission Data

Element (Subelement) • Attribute	Occurrence Required for XML SOAP submission	EC Definition in AP E-cert	Format/ Values	Batch File Tag
Certificate • number	Required	Unique reference number assigned to a document by the issuing country.	AN(6)	<Certificate number ="NZL2009/TEST604/123456"> OR <Certificate number ="123456">
Certificate (Form) • code	Required	Template Identifier	AN(254) Valid template	<Form code ="AP1407"/>

Element (Subelement) • Attribute	Occurrence Required for XML SOAP submission	EC Definition in AP E-cert	Format/ Values	Batch File Tag
Departure(Date)	Required	Estimated/scheduled date that the consignment leaves the consignor's premises Departure date can also be displayed via the attributes <ul style="list-style-type: none"> • Day • Month • Year 	YYYY-MM-DD	<Departure> <Date>2009-09-01</Date> </Departure>
Certificate (Comment)	Optional Repeatable	Free text field for commercial information	AN (2000)	<Comment>Unofficial Commercial Information</Comment>
Certificate (OfficialInformation)	Optional Repeatable	Free text field for official information	AN (2000)	<OfficialInformation>Official Information</OfficialInformation>
Consignee • identifier	Optional	Identifier of party to which goods are consigned In AP E-cert Consignee Identifier is optional.	AN(20) Not validated	<Consignee identifier="Importer ID"> <Name>Importer Name</Name> <Address>Importer Address</Address> <Representative>Importer Representative</Representative> </Consignee>
Consignee (Name)	Required	Name of party to which goods are consigned	AN(254)	
Consignee (Address)	Required	Business address of party to which goods are consigned	AN(254)	
Consignee (Representative)	Optional	Name of a person or party representing the importing company for the purpose of the consignment	AN(254)	
ImportingCountry (Country) • code	Required	Country code 2 character country code as per ISO	A(2)	
Certificate (FinalDestination)	Optional	Location to which the consignment is sent Free text field which you can put the UN/LOCODE code into. UN/LOCODE will get converted to City, Country format on submission	AN(254) A(5)	<FinalDestination>Final Destination</FinalDestination> Or <FinalDestination>BZPND</FinalDestination>

Element (Subelement) • Attribute	Occurrence Required for XML SOAP submission	EC Definition in AP E-cert	Format/ Values	Batch File Tag
Consignor • identifier	Mandatory	Details of the consignor of the consignment. Must be a valid registered exporter ID. In AP E-cert Consignor Identifier is mandatory and therefore must be included in the XML File	AN(20) Validated Premise	<Consignor identifier="TEST604"/>
ThirdParty • identifier	Optional repeatable	License number for third party issued by competent authority.	AN(20) Validated Premise	<ThirdParty identifier="ME86"/>
Certificate (LoadingPort) LoadingPort • code	Optional	Location where the consignment is loaded Must be a valid UN/ECE code of a NZ location In AP E-cert loading port is a single field. The XML file must contain only loading port attribute	A(5) Validated	<LoadingPort code="NZCHC"/>
Certificate (DischargePort) DischargePort • code	Required	Location where the consignment is released for delivery Text description of the inspection port city code There is an AP E-cert business rule that restricts loading port to one value only	A(5)	<DischargePort code="NLRTM"/>
Certificate (InspectionPort) InspectionPort • code	Required	Location where the consignment is inspected upon import for border clearance purposes Text description of the inspection port city code Code must be a valid UN/ECE code	A(5)	<InspectionPort code="USTPA"/>
EU commodity code Subtitle • code	Optional	Code that identifies the subtitle uniquely	AN	<Subtitle code="4.A-2" />
Transport (Mode) Mode • Code	Optional	Transport mode as per UN/CEFACT/2002/CD104 Code specifying the name of a mode of transport. UNECE Recommendation No 19.	N(1) 1 Maritime 4 Air	<Transport> <Mode code="4"/> <ConveyanceRef>Conveyance Reference</ConveyanceRef> <Carrier> <Name>Carrier Name</Name>

Element (Subelement) • Attribute	Occurrence Required for XML SOAP submission	EC Definition in AP E-cert	Format/ Values	Batch File Tag
Transport (ConveyanceRef)	Optional	Transport means journey identifier. To identify a journey of a means of transport, for example voyage number, flight number, trip number.	AN(254)	</Carrier> </Transport>
Transport (Carrier)	Optional	Name of party providing the transport of goods between named points.	Tag only	
Carrier (Name)	Optional	Name of party providing the transport of goods between named points.	AN(254)	

5.2.4. EC Header Example

XML File

```
<Certificate number="NZL2009/TEST604/123456">
  <Form code="AP1407"/>
  <Departure>
    <Date>2009-09-01</Date>
  </Departure>
  <Comment>Unofficial Commercial Information</Comment>
  <OfficialInformation>Official Information</OfficialInformation>
  <Consignee identifier="Importer ID">
    <Name>Importer Name</Name>
    <Address>Importer Address</Address>
    <Representative>Importer Representative</Representative>
  </Consignee>
  <ImportingCountry>
    <Country code="AR"/>
  </ImportingCountry>
  <FinalDestination>Fainal Destination</FinalDestination>
  <Consignor identifier="TEST604"/>
  <ThirdParty identifier="ME86"/>
  <LoadingPort code="NZCHC"/>
  <DischargePort code="NLRTM"/>
  <InspectionPort code="USTPA"/>
  <Transport>
    <Mode code="4"/>
    <ConveyanceRef>Conveyance Reference</ConveyanceRef>
    <Carrier>
      <Name>Carrier Name</Name>
    </Carrier>
  </Transport>
```

5.2.5. EC Product and Process Submission Data

Element (Subelement) • Attribute	Occurrence Required for XML SOAP submission	EC Definition in AP E-cert	Format/ Values	Batch File Tag
Certificate (Product)	Required, Repeatable	Details about the product in this consignment	Tag only	
Product • item	Required	Goods item sequence identifier. Sequence number differentiating a specific goods item entry of a consignment.	N(2) 1...99	<Product item="1">
Product (Description)	Required	Plain text description of the nature of the goods sufficient to identify them at the level required for banking, Customs, statistical or transport purposes, avoiding unnecessary detail (Generic term). In AP E-cert Product Description is mandatory and it is a single free text field. The XML file must contain one product description attribute per product	AN(254)	<Description>Item Description</Description>
Product (Species)	Optional	Description of the species of the product In AP E-cert Species is mandatory and therefore must be supplied in the XML File Multiple species can be entered separated by a space	AN (254)	<Species>Species</Species>
Product (Marks)	Optional,	Goods item shipping marks text. Free form description of the marks and numbers on a transport unit or package.	AN(254)	<Marks>Item Mark</Marks>
Product (Comment)	Optional, Repeatable	Free form description of information details. Any statement relevant to the certification of the product.	AN(4000)	<Comment>Product Comment</Comment>

Element (Subelement) • Attribute	Occurrence Required for XML SOAP submission	EC Definition in AP E-cert	Format/ Values	Batch File Tag
(Classification) Class • code	Required and Optional	Code specifying a type of goods for Customs, transport or statistical purposes (generic term). AP Ecert stipulates that Class Code is required for NZHSC type code. Class code is optional for Classifications with type code = "CL" (which include : <ul style="list-style-type: none"> ▪ Intended Use ▪ Species Code ▪ Temperature ▪ Treatment 	AN(254)	<Classification> <Class code="020110"/> <Type code="NZHSC"/> </Classification>
(Classification) Type • code	Required	Code identifying a user or association maintained code list. Identifies the type of the classification. For example, NZHSC, CL, etc...	AN(254)	
Document • number	Optional	This is required if the product has any incoming EDs supporting outward certification. <certificate number>, <product item>/ <number of packages used>	AN(24)	<Document item="1" number="NZL2009/TEST604/6T" outQuantity="1"/>
Document • item	Optional	Goods item sequence identifier. Sequence number differentiating a specific goods item entry of a consignment.	N(2) 1...99	
Document • outQuantity	Optional	Quantity/count of the number of packages on the original document affected.	N(10)	
Package • count	Required	Quantity/count of the total number of items.	N(10)	<Package count="10" level="1"> <Type code="CR"/> </Package>
(Package) Type • code	Required	Package type code. Code specifying the type of packaging of an item.	AN (2)	

Element (Subelement) • Attribute	Occurrence Required for XML SOAP submission	EC Definition in AP E-cert	Format/ Values	Batch File Tag
Weight • value	Required	Numeric value of the weight of a product item	D (10,4)	<Weight value="1.0000"> <Type code="AAA"/> <Unit code="KGM"/> </Weight>
(Weight) Type • code	Required	Coded description of the weight type (eg. AAA for Net weight and AAB for Gross Weight- which is optional)	AN(3)	
(Weight) Unit • code	Required	Measurement unit code. Indication of a unit of measurement in which weight (mass), capacity, length, area, volume or other quantity or measurement is expressed.	AN(3)	
Container • number	Optional	Transport equipment identifier. To identify a piece of transport equipment, e.g. A container or unit load device	AN(254)	<Container number="Container No" seal="Seal No"></Container>
Container • seal	Optional	Transport equipment seal identifier. The identification number of a seal affixed to a piece of transport equipment.	AN(254)	
Processing • dateOverride	Optional	Indicator set if the process was applied by premises other than the consignor	AN(3) YES/NO	<Processing dateOverride="NO"> <Type code="SLT"/> <StartDate>2009-05-01</StartDate> <EndDate>2009-05-01</EndDate> <Premises identifier="TEST604"/> </Processing>
(Processing)Type • Code	Required	Process type code Code specifying a type of process	AN(3)	
Processing (StartDate)	Optional	Event effective start date time Date and or time on which specified processing event becomes effective. Expiry date can be entered in as a process and it is mandatory to have a Start or an End Date, <ul style="list-style-type: none"> Please note that this will then appear as a durability date under the Product, in the Data view. 	YYYY-MM-DD	
Processing (EndDate)	Optional	Event effective end date time Date and or time on which a specified processing event becomes no longer effective.	YYYY-MM-DD	
Premises • identifier	Optional	Code specifying the identity of a party.	AN(20) Valid premise	

5.2.6. EC Product and Process Example

XML File

```
<Product item="1">
  <Description>Item Description</Description>
  <Species>Species</Species>
  <Marks>Item Mark</Marks>
  <Comment>Product Comment</Comment>
  <Classification>
    <Class code="020110"/>
    <Type code="NZHSC"/>
  </Classification>
  <Package count="10" level="1">
    <Type code="CR"/>
  </Package>
  <Weight value="1.0000">
    <Type code="AAA"/>
    <Unit code="KGM"/>
  </Weight>
  <Weight value="10.0000">
    <Type code="AAB"/>
    <Unit code="KGM"/>
  </Weight>
  <Document item="1" number="NZL2009/TEST604/6T" outQuantity="1"/>
  <Container number="Container No" seal="Seal No"></Container>
  <Processing dateOverride="NO">
    <Type code="SLT"/>
    <StartDate>2009-05-01</StartDate>
    <EndDate>2009-05-01</EndDate>
    <Premises identifier="TEST604"/>
  </Processing>
</Product>
```

5.2.7. EC Declaration Submission Data

Element (Subelement) • Attribute	Occurrence Required for XML SOAP submission	EC Definition in AP E-cert	Format/ Values	Batch File Tag
Approval • form	Optional	Identifier of the document form This identifier appears in the official XML output regardless of what is submitted, so there is no need to submit this. In the training environment the version number is required. But is not required in the live product system.	AN(70)	<Approval form="AP1407"> <Declaration code="AP1407/1"></Declaration> <Declaration code="AP1407/2"></Declaration> </Approval>
Approval (Declaration)	Optional, Repeatable	Declarations relating to the entire certificate, as distinct from product related declarations.	Tag only	
Declaration • code	Optional	Code number for the declaration statement	N(5)	
Declaration (Statement)	Optional, Repeatable	Single statement certifying compliance with a particular requirement of the importing country, or its translation This is not required for submission but is returned in the official XML depending on the declaration codes entered.	AN (32739) Multilingual UTF-8 encrypted	<statement language="en">Statement </statement>
Declaration (Insert)	Optional, Repeatable	A phrase or value that is placed into the statement at a pre-defined position	AN (256) UTF-8 encrypted	<Insert order="1">Germany</Insert>
Insert • order	Required	The position of the space holder within the statement for this insert; counting from 1.	Integer	Not Applicable

5.2.8. EC Declaration Example

XML File

<pre><Approval form="AP1407"> <Declaration code="AP1407/1"></Declaration> <Declaration code="AP1407/2"></Declaration> <Declaration code="AP1407/6"> <Insert order="1">Germany</Insert> </Declaration> </Approval></pre>

6. Eligibility Document Submission DTD and Example

6.1. Eligibility Document Submission DTD

The attached Eligibility Document DTD file (nzfsa-edsub1.0.dtd) defines the XML submission format required

<http://www.foodsafety.govt.nz/industry/exporting/ecert/documents/xml-dtd/>

6.2. Eligibility Document Submission Example

```
<?xml version="1.0" encoding="utf-8"?>
<EligibilityDocument exhaustion="NO" number="987564">
  <Form code="AA"></Form>
  <Departure>
    <Date>2008-12-15</Date>
  </Departure>
  <Comment>just testing the unofficial commercial information
  box testing testing testing testing testing</Comment>
  <Consignee identifier="ME85"/>
  <FinalDestination>NZOTA</FinalDestination>
  <Consignor identifier="PHOENIX1"/>
  <ThirdParty identifier="EX1456"/>
  <ThirdParty identifier="ZABEEL1"/>
  <Transport>
    <Mode code="4"/>
    <ConveyanceRef>Ref ZXHY</ConveyanceRef>
    <Carrier>
      <Name>Fine Carriers</Name>
    </Carrier>
  </Transport>
  <Product exhaustion="NO" item="1">
    <Durability>
      <Date>2008-12-05</Date>
    </Durability>
    <Description>first product</Description>
    <Species>Nototodarus gouldi Jasus edwardsii</Species>
    <Marks>THIS IS THE ITEM MARK - MARK URJHN</Marks>
    <Comment>THIS IS THE COMMENT FOR PRODUCT NUMBER ONE.
    THE SECOND LINE OF COMMENT FOR PRODUCT NUMBER ONE.</Comment>
    <Classification>
      <Class code="CH"/>
      <Type code="DC"/>
    </Classification>
    <Classification>
      <Class code="LI"/>
      <Type code="DC"/>
    </Classification>
    <Classification>
      <Class code="BM"/>
      <Type code="CC"/>
    </Classification>
    <Classification>
      <Class code="020210"/>
      <Type code="NZHSC"/>
    </Classification>
    <Package count="345" level="1">
      <Type code="TK"/>
    </Package>
    <Weight value="345.0000">

```

```
<Type code="AAA"/>
<Unit code="KGM"/>
</Weight>
<Weight value="987.0000">
  <Type code="AAB"/>
  <Unit code="KGM"/>
</Weight>
<Document item="1" number="NZL2008/PHOENIX1/932487T" outQuantity="45"/>
<Document item="2" number="NZL2008/PHOENIX1/998754T" outQuantity="86"/>
<Container number="CONTAINER 789" seal="S90UFJVD8 SEAL"></Container>
<Processing dateOverride="NO">
  <Type code="SLT"/>
  <StartDate>2008-09-05</StartDate>
  <EndDate>2008-09-05</EndDate>
  <Premises identifier="ME85"/>
</Processing>
</Product>
<Product exhaustion="NO" item="2">
  <Description >second product with no1</Description>
  <Species >Nototodarus gouldi</Species>
  <Classification>
    <Class code="HC"/>
    <Type code="DC"/>
  </Classification>
  <Classification>
    <Class code="IN"/>
    <Type code="DC"/>
  </Classification>
  <Classification>
    <Class code="ZE"/>
    <Type code="CC"/>
  </Classification>
  <Classification>
    <Class code="020130"/>
    <Type code="NZHSC"/>
  </Classification>
  <Package count="375" level="1">
    <Type code="PL"/>
  </Package>
  <Weight value="345.0000">
    <Type code="AAA"/>
    <Unit code="KGM"/>
  </Weight>
  <Processing dateOverride="NO">
    <Type code="LLF"/>
    <StartDate>2008-08-02</StartDate>
    <EndDate>2008-08-03</EndDate>
    <Premises identifier="PHOENIX1"/>
  </Processing>
</Product>
</EligibilityDocument>
```

7. Export Certificate Submission DTD and Example

7.1. Export Certificate Submission DTD

The attached Export Certificate DTD file (nzfsa-ecsub1.0.dtd) defines the XML submission format required

<http://www.foodsafety.govt.nz/industry/exporting/ecert/documents/xml-dtd/>

7.2. Export Certificate Submission Example

```
<?xml version="1.0" encoding="utf-8"?>
<Certificate number="NZL2008/PHOENIX1/786446">
  <Form code="DEBSR22"/>
  <Departure>
    <Date>2008-12-12</Date>
  </Departure>
  <Comment >the unofficial commercial information is that this is good</Comment>
  <OfficialInformation >this is the official information on this certificate</OfficialInformation>
  <Consignee identifier="any importer or a valid premis">
    <Name>TEST</Name>
    <Address>TEST</Address>
    <Representative>Mr Importer</Representative>
  </Consignee>
  <ImportingCountry>
    <Country code="FR"/>
  </ImportingCountry>
  <FinalDestination>The final destination could be in the moon</FinalDestination>
  <Consignor identifier="PHOENIX1"/>
  <ThirdParty identifier="ME85"/>
  <ThirdParty identifier="EX1456"/>
  <LoadingPort code="NZWLG"/>
  <DischargePort code="CAVAN"/>
  <InspectionPort code="CAYVR"/>
  <Subtitle code=""/>
  <Transport>
    <Mode code="1"/>
    <ConveyanceRef>Reference 989998</ConveyanceRef>
    <Carrier>
      <Name>Hot Carriers</Name>
    </Carrier>
  </Transport>
  <Product item="1">
    <Description >SDFSDF</Description>
    <Species >Bos taurus</Species>
    <Marks>item tx657</Marks>
    <Comment>this is the product comment</Comment>
    <Classification>
      <Class code="020130"/>
      <Type code="NZHSC"/>
    </Classification>
    <Classification>
      <Class code="Intended use">slaughter</Class>
      <Type code="CL"/>
    </Classification>
    <Classification>
      <Class code="Species code">C10221S10752R2514</Class>
      <Type code="CL"/>
    </Classification>
  </Product item>
</Certificate>
```

```

<Classification>
  <Class code="Temperature">frozen</Class>
  <Type code="CL"/>
</Classification>
<Classification>
  <Class code="Treatment">carcasses-side</Class>
  <Type code="CL"/>
</Classification>
<Document item="1" number="NZL2008/PHOENIX1/932487T" outQuantity="45"/>
<Document item="1" number="NZL2008/PHOENIX1/998754T" outQuantity="34"/>
<Package count="465435" level="1">
  <Type code="TU"/>
</Package>
<Weight value="34536.0000">
  <Type code="AAA"/>
  <Unit code="KGM"/>
</Weight>
<Weight value="44895.0000">
  <Type code="AAB"/>
  <Unit code="KGM"/>
</Weight>
<Container number="Container No 78656" seal="Seal number 87568"></Container>
<Processing dateOverride="NO">
  <Type code="LLF"/>
  <Premises identifier="PHOENIX1"/>
</Processing>
<Processing dateOverride="NO">
  <Type code="HAN"/>
  <StartDate>2008-08-08</StartDate>
  <EndDate>2008-08-10</EndDate>
  <Premises identifier="ME85"/>
</Processing>
</Product>
<Product item="2">
  <Description language="en">Product number two</Description>
  <Species language="en">Capra hircus</Species>
  <Marks>ITEM MARK - 87JH</Marks>
  <Comment></Comment>
  <Classification>
    <Class code="020110"/>
    <Type code="NZHSC"/>
  </Classification>
  <Classification>
    <Class code="Intended use">feedingstuff</Class>
    <Type code="CL"/>
  </Classification>
  <Classification>
    <Class code="Species code">C10238S10827R2514</Class>
    <Type code="CL"/>
  </Classification>
  <Classification>
    <Class code="Temperature">frozen</Class>
    <Type code="CL"/>
  </Classification>
  <Classification>
    <Class code="Treatment">combinedheattre</Class>
    <Type code="CL"/>
  </Classification>
  <Package count="987" level="1">
    <Type code="PL"/>
  </Package>
  <Weight value="987.0000">
    <Type code="AAA"/>

```

```
<Unit code="KGM"/>
</Weight>
<Container number="Con.YYXU" seal="S.987JO"></Container>
<Processing dateOverride="NO">
  <Type code="FRD"/>
  <Premises identifier="PHOENIX1"/>
</Processing>
</Product>
<Approval form="DEBSR22">
  <Declaration code="DEBSR22/1"></Declaration>
  <Declaration code="DEBSR22/2"></Declaration>
  <Declaration code="DEBSR22/4">
    <Insert order="1">19/08/08</Insert>
  </Declaration>
</Approval>
</Certificate>
```