



BICON Software Developers Guide

Department of
Agriculture and
Water Resources

Version 2.2
Date 04/04/2017



Revision History			
Date	Version	Description	Author
18/09/2015	2.1	Revised to make a final	
04/04/2017	2.2	Updated technical specifications. Removed some non-technical definitions. Updated references. Minor description edits and updated casing on parameters.	

Contents

Introduction.....	7
Overview.....	7
Purpose.....	8
Audience.....	8
Scope	8
Interface Testing.....	9
Reference Material.....	9
Maintenance of Interfaces	9
Disclaimer	9
Abbreviations and Acronyms.....	10
BICON Overview	11
Services Overview and Standards	13
Process Flow	13
Software Developer Package.....	14
BICON Schemas	14
How do I extract the schemas?.....	14
What is in the Schemas?	15



What is not in the Schemas?	15
How to use the Schemas.....	15
Web Services	17
Returned XML String.....	18
Validating Data against the Schema	19
Business Rules.....	19
Why business rules are required	19
When to apply the business rules.....	19
High Level Business Rules	19
Authentication Rules.....	20
Required Reference Data.....	20
Message Exchange	21
SOAP over HTTPS.....	21
WSDL File	22
SOAP Faults.....	22
Request-Response Message Exchange.....	22
Error Handling.....	23
FaultContract	23
ErrorContract	23
Interface Capability and Logical Information Flows.....	24
Overview.....	24
Check the service is available	25
PingCLSP	25
Perform a search for the required data	25
SearchCasesByTerms	26
SearchCasesByScientificName	26



SearchTariffWizards	27
Interrogate the data	27
RetrieveCaseDetails	27
EvaluateCase	28
EvaluateTariffWizard.....	29
SaveCaseEvaluation	29
RetrieveMediaRequirement	29
Render the data	29
RenderCase	30
RetrievePermit	30
RetrieveRenderedDocument	30
Services.....	30
Ping	31
PingCLSP	31
Searching	32
SearchCasesByTerms	32
SearchTariffWizards	33
SearchCasesByScientificName	34
Interrogating.....	35
EvaluateTariffWizard.....	35
EvaluateCase	36
RetrieveMediaRequirement	38
RetrieveCaseDetails	39
SaveCaseEvaluation	40
Rendering.....	41
RenderCase	41



RetrievePermit	42
RetrieveRenderedDocument	43
Response Data Contracts	43
Common Message Fields	43
Searching	44
SearchCasesByTermsResponseDataContract	44
SearchTariffWizardsResponseDataContract	44
SearchCasesByScientificNameResponseDataContract	45
Interrogating	45
EvaluateTariffWizardResponseDataContract	45
EvaluateCaseResponseDataContract	46
RetrieveMediaRequirementResponseDataContract	47
RetrieveCaseDetailsResponseDataContract	48
SaveCaseEvaluationResponseDataContract	49
Rendering	50
RenderCaseResponseDataContract	50
RetrieveRenderedDocumentResponseDataContract	51
RetrievePermitResponseDataContract	51
Data Contracts	52
CaseSearchResultDataContract	52
NoticeDataContract	53
CaseNoticeDataContract	53
QuestionDataContract	54
AnswerDataContract	55
TariffSearchWizardDataContract	55
CaseVersionInformationDataContract	56



ErroredQuestionDataContract.....	56
EvidenceDataContract	57
EvidenceCriteriaDataContract	57
EvidenceCriteriaConditionDataContract.....	58
WarningAndInformationNoticesContract.....	58
MediaGroupDataContract	58
ElementContentDataContract	59
Reference Types	59
Searching	59
CaseSearchTermSet	59
Interrogating.....	60
NoticeType	60
ElementType	60
BICONStatus.....	61
BICONQuestionType	61
PermitStatus.....	61
QuestionModeType	61
Rendering.....	62
BICONOutputType.....	62

Figures

Figure 1 BICON Information Structure	12
Figure 2 BICON sample data flow	12
Figure 3 BICON Interface process flow.....	13
Figure 4 BICON Logical Information Flow.....	24



Introduction

Overview

This document provides details of the Biosecurity Import Conditions (BICON) external web services interfaces that enables third party software developers to create and maintain data communications with the BICON application.

The BICON web services provide a read only interface to the BICON system and do not provide the ability to register or apply for permits.

The document has been structured to allow for a logical flow of information and includes the following sections:

- **BICON Overview** – this section outlines BICON’s core capability and information structures.
- **Services Overview and Standards** – this section outlines BICON’s process flows, the available web services schemas and generic business rules. This information is provided to ensure that third parties can correctly extract the required information from the BICON system.
- **Message Exchange** – this section defines the details that apply to message exchange between BICON and third party applications.
- **Interface Capability and Logical Information Flows** – this section defines the web services interface capabilities and logical information flows.
- **Services** - this section provides detailed information on each BICON web service and its use.
- **Response Data Contracts** - this section outlines the data returned by the BICON web services interface in response to messages received.
- **Data Contracts** - this section details the various data contracts contained within the BICON web services.
- **Reference Types** - this section lists the available reference types and their values.

Please note that all XML shown within this document is provided for example purposes only and all third party software developers must download the currently released version from within the software developer’s package or directly from the web services via the WSDL.



Purpose

This document describes the BICON external web services interfaces that allow third party software developers to retrieve Department of Agriculture and Water Resources import conditions and permits for use within their applications.

Audience

This document should be read by any third party software developers intending to design, implement or maintain an interface to the BICON application.

Scope

This document relates to the BICON external web services interface and specifically addresses:

- Searching for BICON Cases using Term, Scientific Name, and Tariffs;
- Interrogating a Case to determine specific Import Conditions; and
- Rendering a case to view the Import Conditions retrieving a Permit that has been granted by the Department.



Interface Testing

To allow third party software developers to test their interfaces the Department of Agriculture and Water Resources will establish a testing environment. This environment will allow developers to communicate with the BICON web services to ensure that their applications are transmitting acceptable data.

The services within this environment will perform the same functionality as the live services. Access to the test environment is restricted. To gain access please e-mail BICON.Administrator@agriculture.gov.au.

Reference Material

In addition to this document further information regarding BICON, its functions and data will be available on the BICON web site. The Department of Agriculture and Water Resources recommends that all developers familiarise themselves and keep up to date with the data provided on the BICON web site.

Maintenance of Interfaces

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

During these changes a developer may find that their particular application requires further development to remain compliant with the BICON 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 of Agriculture and Water Resources will allow an adequate period of time for software developers to amend their software in response to interface changes. All third party software developers who wish to continue to utilise the BICON interfaces will need to make the necessary changes to their application to comply with these business rule changes and may be required to redeploy their applications.

Disclaimer

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

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



- Department of Agriculture and Water Resources 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 of Agriculture and Water Resources makes no representation, and provides no warranty, of any kind (whether express or implied) in respect to BICON, 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 of Agriculture and Water Resources or any other person.

It is mandatory that each third party software developer using the BICON web services display and enforce acknowledgement of the above disclaimer to all third party users of their application.

Abbreviations and Acronyms

The following abbreviations and acronyms are used throughout this document.

Abbreviations and Acronyms

Acronym	Description
Alert	An Alert is a type of notice which is used to highlight to BICON users critical announcements in relation to Cases and Import Conditions. For example an Alert would be created in the case of an outbreak of a particular Department of Agriculture and Water Resources concern such as the Swine Flu (H1N1 influenza virus) outbreak in 2009.
BICON	Biosecurity Import Conditions application. This system supersedes the Import Conditions (ICON) application. BICON'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.
Case	A collection of Import Conditions grouped under a common theme e.g. Wool.
CLSP	Cargo Logistics Software Providers – the external development companies who will integrate the web servers into their applications as required.
Evaluate	The process of completing Q&A's to navigate through a Wizard to retrieve a specific Case, or navigate through a Case to retrieve specific import conditions.
ICON	Import Conditions application. Superseded by the BICON application.



Acronym	Description
Import Conditions	Conditions that are or may be imposed by legislation, under which a consignment may be imported.
Import Permit	A written document granted by the Director of Biosecurity or a delegate to the person, or a person acting on behalf of the person, to bring or import certain conditionally non-prohibited goods into Australian territory.
Q&A's	Questions and Answers.
MTOM	SOAP Message Transmission Optimization Mechanism.
Notice	Notices provide Industry and Department of Agriculture and Water Resources Officers with a range of additional information including important changes or announcements. They include Alerts, Warnings, Information, Instructions, What's Changed and Authoring Notes.
Service Request	A service request allows clients to log and track problems relating to the BICON System and the authored content.
SOAP	Simple Object Access Protocol.
SSL	Secure Sockets Layer.
What's Changed	A particular type of Notice that is used to indicate whether there have been important changes to a Case. Cases can be changed for other administrative reasons such as changes to sentence structure or spelling.
Wizard	A specific set of Q&A's which result in the identification of a single Case or Notice. As a user progresses through the Wizard a decreasing set of results is returned until only one remains.
WSDL	Web Services Description Language.
XML	Extensible Markup Language.
XMLDSIG	An XML Signature used for signing and verification.
XSD	XML Schema Definition.

BICON Overview

BICON'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.

This flow is shown below in Figure 1 and further expanded in Figure 2. All third party software developers should be aware of these flows when designing the implementation of their applications to ensure the correct information is returned to their applications. Failure to understand this information could lead to additional information being returned for a specified import scenario.

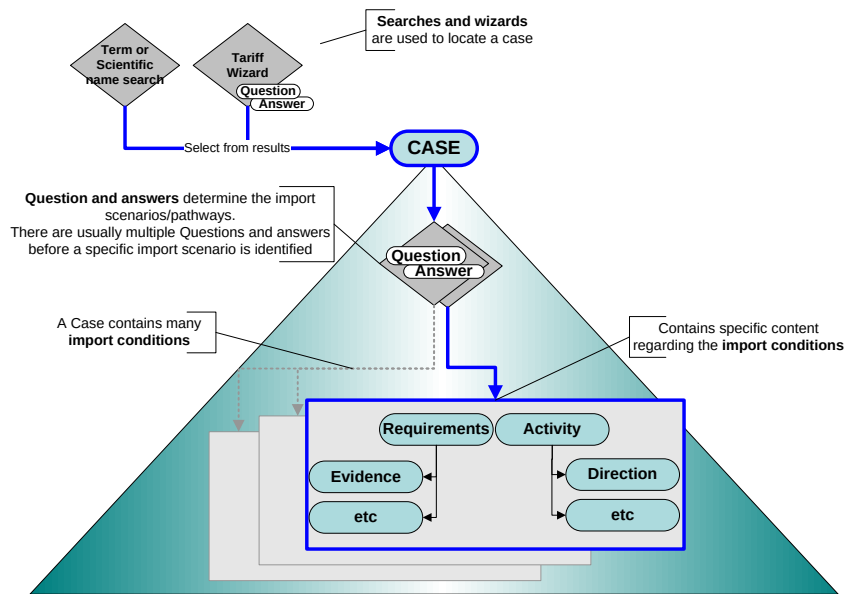


Figure 1 BICON Information Structure

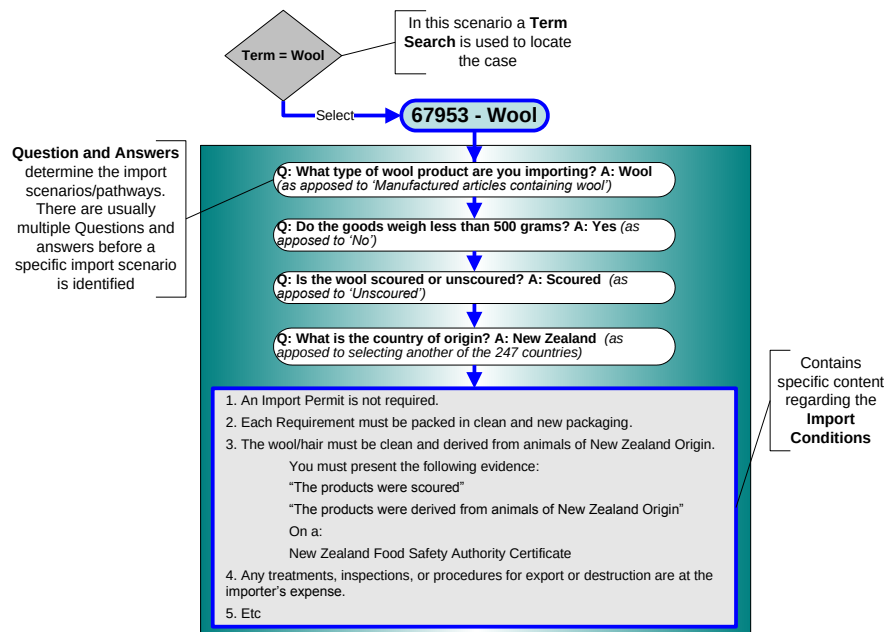


Figure 2 BICON sample data flow



Services Overview and Standards

Process Flow

The below diagram provides details, broken out into the relevant party's area of responsibility, of the possible steps to implement the BICON Interface. Each third party development would need to evaluate these to determine if they suit their needs.

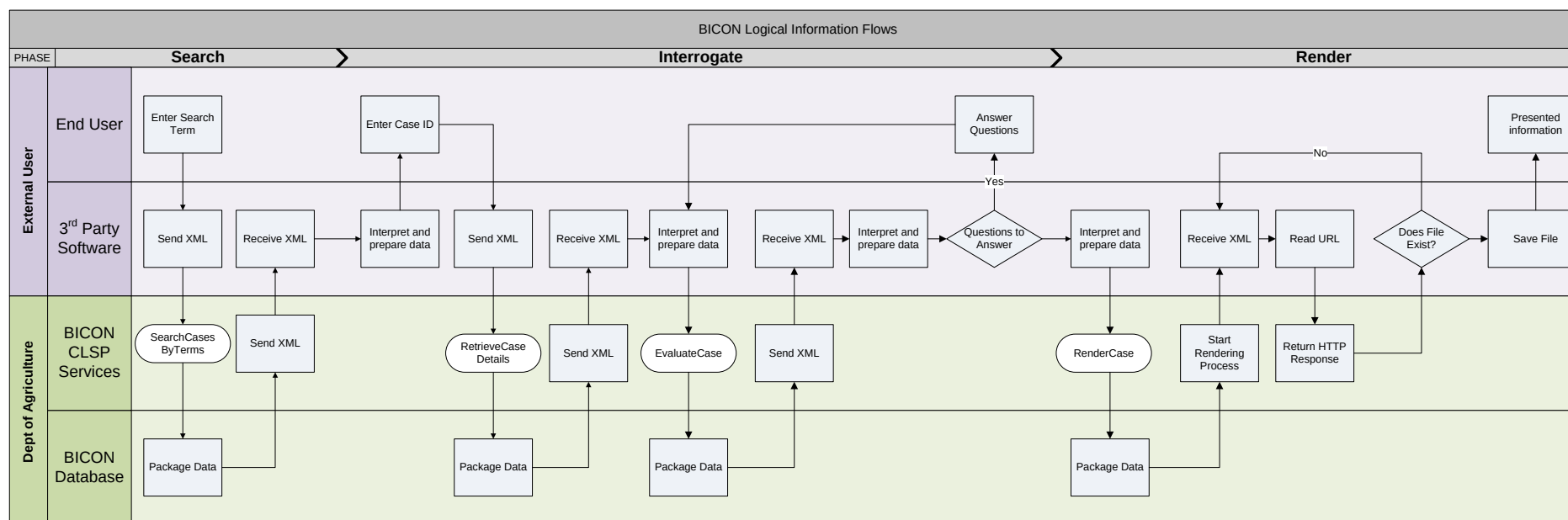


Figure 3 BICON Interface process flow



Software Developer Package

To assist third party software developers in their activities the Department of Agriculture and Water Resources will provide a software developers package. This package will combine the BICON schemas and other supporting documentation into one compressed downloadable file. Each individual file that is contained in the compressed file should then be extracted retaining the full path structure to ensure usability.

BICON Schemas

A number of BICON XML Schemas have been created to enable the electronic transmission of data requests from external users to BICON. These schemas are all generated from a common conceptual data model and form part of the service WSDL and are located within the `<wsdl:types>` section. They are known by the namespace `http://bicon.agriculture.gov.au`.

The two custom XSD's that are directly relevant to the CLSP Services are:

- AGRICULTURE.ICON.Web.ExternalServices.Contracts
- AGRICULTURE.ICON.Web.Shared.Enumerations

The additional XSD's that are imported for the services are:

- `schemas.microsoft.com.2003.10.Serialization.Arrays.xsd`
- `schemas.microsoft.com.2003.10.Serialization.xsd`

How do I extract the schemas?

Locate the `<wsdl:types>` section of the XML file by opening the WSDL (`/CargoLogisticsService.svc?wsdl`) in a web browser. Once complete the XML shown below or similar will be displayed. As seen this information details the Schema's that are used by this WSDL.

```
<wsdl:types>

  <xsd:schema targetNamespace="http://www.bicon.agriculture.gov.au/Imports">

    ...

    <xsd:import
      schemaLocation="https://bicon.agriculture.gov.au/Biconweb4.0.CLSPServices/CargoLogisticsService.
      svc?singleWsdI"
      namespace="http://schemas.datacontract.org/2004/07/AQIS.ICON.Web.ExternalServices.Contracts"
    />

    ...

  </xsd:schema>
</wsdl: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. For example based on the



information provided above pasting the below line into your browser will present you with the schema for the AQIS.ICON.Web.ExternalServices.Contracts.

<https://bicon.agriculture.gov.au/Biconweb4.0.CLSPServices/CargoLogisticsService.svc?singleWsdI>

What is in the Schemas?

Each schema contains a definition of all the data elements and their validation rules that make up a BICON 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:

```
<xs:element minOccurs="0" name="Cases" nillable="true" type="tns:ArrayOfCaseSearchResultDataContract" />
```

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 CLSP validation process and be performed prior to transmission to the BICON application to prevent rejected messages. More details on the specific business rule requirements are defined in the following sections of this document.

How to use the Schemas

All third party 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 BICON 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 business rules determined by the external interface into BICON.

Element Attributes

The following attributes will form part of the element and must be used to validate the data being captured. An example element is detailed below.

```
<xs:element name="ElementType" nillable="true" type="tns:ElementType" />
```

```
<xs:complexType name="InvalidParameterFaultContract">
```

```
<xs:sequence>
```

```
<xs:element minOccurs="0" name="Description" nillable="true" type="xs:string" />
```



```
<xs:element minOccurs="0" name="FieldName" nillable="true" type="xs:string" />
```

```
</xs:sequence>
```

```
</xs:complexType>
```

Name

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

```
name="ElementType"
```

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. An example type is detailed below.

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

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

```
<xs:simpleType name="ElementType">  
  <xs:restriction base="xs:string">  
    <xs:enumeration value="Media" />  
    <xs:enumeration value="Activity" />  
    <xs:enumeration value="Procedure" />  
    <xs:enumeration value="ExternalReference" />  
    <xs:enumeration value="TRIMReference" />  
    <xs:enumeration value="File" />  
    <xs:enumeration value="Annex" />  
  </xs:restriction>  
</xs:simpleType>
```

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="tns:MatchTypes" />
```

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, then it can occur as many times as needed. An example MaxOccurs is detailed below

```
<xs:element minOccurs="0" maxOccurs="unbounded" name="MatchTypes" type="tns:MatchTypes" />
```

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>
```

Web Services

All third party applications will need to communicate with the BICON Web Services developed for processing known XML schemas. There are several web services available that third party application can communicate with. The communication between the third party applications and BICON will be over a secure sockets layer (SSL) and will be encrypted using the Department of Agriculture and Water Resources digital certificate.

The testing (Test) site will allow for all of the same functionality as the Production (Prod) site and is a separate environment. All registrations on the Test site will be separate to those on the live site. It is up to the user to know which messages are being received from the Test site and which ones are being received from the Prod site.

The URLs to be accessed are defined in the table below.

Env	URL	Description
Prod	https://www.bicon.agriculture.gov.au/BICONWeb4.0	BICON Web site used for registration and additional BICON activities.
Prod	https://bicon.agriculture.gov.au/BiconWeb4.0.CLSPServices/CargoLogisticsService.svc	BICON CLSP Services used by 3 rd Party applications.
Prod	https://bicon.agriculture.gov.au/BiconWeb4.0.CLSPServices/CargoLogisticsService.svc?wsdl	BICON CLSP Services WSDL used by 3 rd Party applications.



Returned XML String

An example of the returned XML string (as an XSD) is provided below:

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

<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
XMLNs:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="record">

    <xs:complexType>

      <xs:sequence>

        <xs:element name="systemmessages">

          <xs:complexType>

            <xs:sequence>

              <xs:element maxOccurs="unbounded" name="message">

                <xs:complexType>

                  <xs:simpleContent>

                    <xs:extension base="xs:string">

                      <xs:attribute name="id" type="xs:unsignedShort" use="required" />

                      <xs:attribute name="type" type="xs:string" use="required" />

                    </xs:extension>

                  </xs:simpleContent>

                </xs:complexType>

              </xs:element>

            </xs:sequence>

          </xs:complexType>

        </xs:element>

      </xs:sequence>

    </xs:element>

    <xs:attribute name="recordidentifer" type="xs:string" use="required" />

    <xs:attribute name="clientid" type="xs:unsignedInt" use="required" />

    <xs:attribute name="status" type="xs:string" use="required" />

  </xs:element>

</xs:schema>
```



</xs:complexType>

</xs:element>

</xs:schema>

Validating Data against the Schema

Once the data has been captured in the 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 BICON.

When the data is received by BICON, it will be validated against the schema stored in BICON to ensure that the data received is valid and that the schema used by the third party application is still correct.

Business Rules

Why business rules are required

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

When to apply the business rules

It is up to the third party software developer to determine how and when the business 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 business rules are applied before the data is transmitted to BICON.

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

High Level Business Rules

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

XML File

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

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 of Agriculture and Water Resources.

Authentication Rules

All interactions with the CLSP Services will require a **UsernameToken** within the header of the SOAP message. This will be used to authenticate the credentials against the BICON registered users store and will be used to authorize the user when accessing particular data. Failure to include the **UsernameToken** within the header will result in a rejection of the message. An example of the **UsernameToken** for the SOAP Headers is:

```
<soapenv:Header>
```

```
  <wsse:Security soapenv:mustUnderstand="1" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">
```

```
    <wsse:UsernameToken wsu:Id="UsernameToken-1" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
```

```
      <wsse:Username>myusername</wsse:Username>
```

```
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0#PasswordText">mypassword</wsse:Password>
```

```
      <wsse:Nonce EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary">nGwik+EFPT2PDjPcArZp4A==</wsse:Nonce>
```

```
    <wsu:Created>2012-08-15T04:05:15.920Z</wsu:Created>
```

```
  </wsse:UsernameToken>
```

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

Required Reference Data

The allowable values for elements that are not permitted to be “free text” 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 **CaseSearchTermSet** to be defined as a set of four possible string values:

```
<xs:simpleType name="CaseSearchTermSet">
```

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

```
    <xs:enumeration value="TitlesAndSynonyms" />
```

```
    <xs:enumeration value="CommonNames" />
```



```
<xs:enumeration value="AllTermSets" />
```

```
</xs:restriction>
```

```
</xs:simpleType>
```

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

(tns:CaseSearchTermSet):

```
<xs:element name="CaseSearchTermSet" nillable="true" type="tns:CaseSearchTermSet" />
```

This method ensures that only the allowable values can be entered. If a user enters data that is not one of the three options, then the schema validation will fail and the third party application should not transmit the XML to BICON for further processing.

Other supporting data types are captured in the secondary schemas, for example, the following XML example shows how the **duration** data type is defined:

```
<xs:element name="duration" nillable="true" type="tns:duration" />
```

```
<xs:simpleType name="duration">
```

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

```
<xs:pattern value="\-?P(\d*D)?(T(\d*H)?(\d*M)?(\d*(\.\d*)?S)?)" />
```

```
<xs:minInclusive value="-P10675199DT2H48M5.4775808S" />
```

```
<xs:maxInclusive value="P10675199DT2H48M5.4775807S" />
```

```
</xs:restriction>
```

```
</xs:simpleType>
```

Message Exchange

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

SOAP over HTTPS

All BICON web services utilise the Simple Object Access Protocol (SOAP) and BICON 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.

A 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 BICON will need to comply with the WSDL.



WSDL File

All third party software developers wishing to use the BICON web services will have access to download the Web Service Description Language (WSDL) generated by the service by using the URL's provided in section 0 Web Services.

SOAP Faults

All errors in BICON web services are reported using SOAP faults. When an error occurs as a result of an internal BICON system problem the fault code provided in the SOAP fault will indicate "InternalServiceFault". In these circumstances CLSP software can resend the same request at a later time.

When business rule checks are being performed on an incoming message any non-compliance with the business rules will be reported with a descriptive fault message. The CLSP software should not attempt to resend the same message without firstly correcting the issues identified.

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

An example Fault is shown below.

```
<s:Fault>

  <faultcode>s:Client</faultcode>

  <faultstring xml:lang="en-AU">The case or specified version does not exist.</faultstring>

</s:Fault>
```

Request-Response Message Exchange

All CLSP web services must 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 CLSP web service operations occur synchronously and provide an immediate response. In many cases, all the BICON activity required for the transaction invoked by the request message is completed before the response message is returned. However the rendering of Cases and Permits requires additional activity to be performed which is performed asynchronously. The additional activity will be queued and third party software will have to intermediately poll to check for completion. The recommended period for polling is 5 seconds. Once the rendering process has completed, the message response will provide a base64 byte version of the document as well as a URL where the document can be downloaded as required.



Error Handling

Common errors encountered during the actioning of a web service request will be returned to the caller as a SOAP fault. Faults will be compliant with SOAP 1.1. Faults will be embedded in the fault element of the response data contract for the web service method.

Common SOAP faults will be returned when the requested information is invalid or if there is another underlying exception during the request. For example if a request contract requires a date field in a specific date format then any request supplied with an invalid date in that field will generate a generic fault contract for an invalid parameter in the request. Uncommon errors will not follow this standard and will be returned as SOAP errors.

Callers can expect the following as part of a common SOAP fault element in the return message:

FaultContract

Property Name	Data Type	Description
FaultCode	Long	Number representing the fault. Each fault has a unique number.
FaultDescription	String	Description of the error. For example "One or more of the parameters are invalid"
FaultMessage	String	Additional information for the fault instance.

Callers can expect the following as part of an uncommon SOAP fault element in the return message:

ErrorContract

Property Name	Data Type	Description
FaultCode	String	A short description of the fault message.
FaultString	String	Description of the error. For example "One or more of the parameters are invalid"



Interface Capability and Logical Information Flows

This section defines the interface capabilities and logical information flows between third party software applications and BICON. It provides specific details on each option available.

Overview

The below diagram provides an example of the interface capabilities and typical logical information flows.

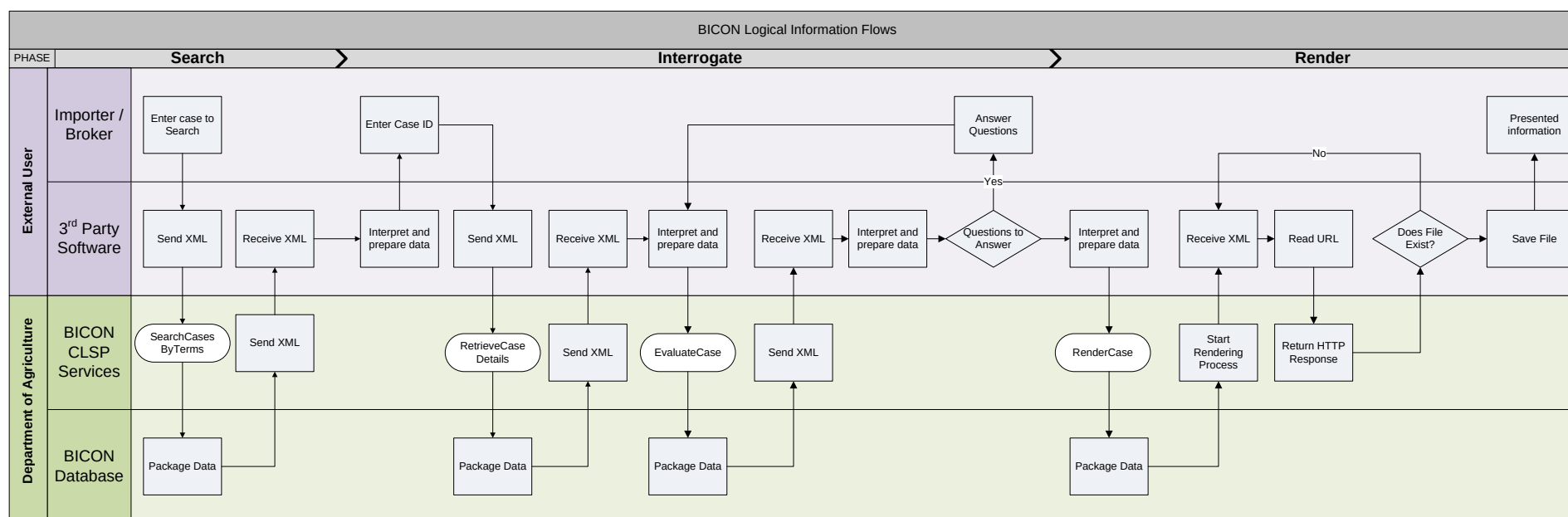


Figure 4 BICON Logical Information Flow



Figure 4: BICON Logical Information Flows shows a sample scenario of a search of a case. All scenarios begin with searching for a Case. Once this information is located you can then perform various other actions as outlined below.

1. Check the service is available
 - Confirm you can access the services and be authenticated using your credentials.
2. Perform a Search for the required data by:
 - Searching for a Case using a:
 - i. Term Search
 - ii. Scientific Name Search
 - iii. Tariff Search
3. Interrogate the data by:
 - Choosing a Case and:
 - i. Retrieve the Case details
 - ii. Evaluate the Case
4. Render the data by:
 - Using the located Permit or Case and:
 - i. Render the Case
 - ii. Retrieve the Permit

Each of these steps is further discussed in the next section of this document and details of each method are provided in the **Services** section of this document.

Check the service is available

The CLSP services allow a user to “ping” the service to check for availability of the services.

PingCLSP

This method can be used during development as a simple service call to ensure you have connectivity and are passing through the correct credentials. It can also be used in Production mode as a connection check before passing through larger amounts of data.

Perform a search for the required data

The CSLP services allow third party systems to search for and retrieve Cases from the BICON system using a variety of methods. The method used will depend on the criteria that are available or preferred. For example, if the Tariff code is available then a specific Case for the Tariff code can be retrieved using that Tariff code. If the only information is a product, material or commodity name then the case can be searched for using these terms.



The available BICON web services provide for search capabilities are:

- Search Cases using Terms
- Search Cases using Scientific Names
- Search Cases using Tariff Wizard

Provided on the following pages are explanations of when to use each search, the matching criteria, and the expected results.

SearchCasesByTerms

SearchCasesByTerms gives the external system the ability to search for cases by entry of a string containing multiple search terms. This is the primary method of searching for Cases. Depending on the number of search results returned, it may be difficult to determine which Case is the actual Case desired. Additional information can be requested for specific Cases using the **RetrieveCaseDetails** method.

Examples of using this method are:

- Used when searching on commodity name e.g. "Horse";
- BICON matches against Case Title, Synonym or Common Name;
- BICON returns a result list which includes cases and associated notices, or it could return no matching results;
- BICON returns a ranking for each result showing how well it matched the term;
- Contains other basic details for the Case such as Case ID, Title and version number;

SearchCasesByScientificName

SearchCasesByScientificName gives the external system the ability to search for cases by entry of one or more scientific names. Depending on the number of search results returned, it may be difficult to determine which Case is the actual Case desired from the search results information. Additional information can be requested for specific Cases using the **RetrieveCaseDetails** method.

Examples of using this method are:

- Used when searching on commodity scientific name e.g. "Equus"
- BICON matches against scientific names.
- BICON returns a result list which includes cases and associated notices, or it could return no matching results.
- BICON returns a ranking for each result showing how well it matched the scientific name.
- Contains other basic details for the Case such as Case ID, Title and version number.



SearchTariffWizards

SearchTariffWizards gives the external system the ability to search for Cases and by the entry of a Tariff Code. Because Cases are not directly associated to Tariffs, the results of this search are Tariff Wizards (Tariff Driven Search Decision Trees). There will be one Tariff Wizard (directly or indirectly through the Tariff hierarchy) returned for the supplied Tariff code. The Tariff Wizard will contain all the Cases for the entered Tariff. Use the **EvaluateTariffWizard** method to process the Tariff Wizard and retrieve information on Cases for the Tariff. This method is similar to the **EvaluateCase** method. It requires a list of answered questions to be submitted to the method. It then returns a list of matching cases based on the evaluation of the Tariff Search Decision-tree.

An initial call to the **EvaluateTariffWizard** with no answered questions will return all Cases in the tree. It will also return all unanswered questions in the tree.

Examples of using this method are:

- Used when searching on commodity Tariff Classification e.g. 0101.10.00, 5101.21.00 etc. A minimum of 4 characters is required for all searches.
- BICON matches against tariff wizards. BICON attempts to find a tariff wizard at the maximum set of characters provided (e.g. up to 10 characters) and continues up the tariff hierarchy until a result is found i.e. there may be a result at the chapter level.
- BICON returns a result list of matching Tariff Wizards which contain cases &/or notices, or it could return no matching results.
- Third Party software should then select and pass a Wizard ID into **EvaluateTariffWizard**.
- BICON response to this contains Q&A's to evaluate the Wizard with the result being a diminishing list of matching cases &/or notices.

Interrogate the data

Once you have located the relevant data from your search you can further interrogate the data by choosing a Case ID and:

- Check the Status and Change History of a Case (**RetrieveCaseDetails**).
- Evaluate a Case to find specific Import Conditions (**EvaluateCase**).

Provided below is an explanation of when to use each method and the expected results.

RetrieveCaseDetails

- Used when wanting to know general information about a case such as:
 - The version history of a Case including the latest version number,
 - Change Notices for each version of a Case, and
 - Alerts for each version of a Case.



- BICON returns a response containing a history of versions with any Change notices or Alerts that were created against each version of the Case. E.g.:
 - If there are change notices for a new version of a Case its likely they are Department of Agriculture and Water Resources risk-related changes and therefore potentially changes in import conditions,
 - If there are no change notices for a new version of a Case it is likely the change is simply an administrative change i.e. fixing spelling or grammatical errors, or rewording to simplify language.
- For each Change Notice and Alert there will be a promotion date and expiry date.

EvaluateCase

- Used when filtering a case down to Import Conditions that are more specific to a particular cargo scenario.
- Pass in the Case ID, Case Version and question mode.
- If Case Version is left blank the latest version will be used.
- BICON Response contains all the Questions & Answers contained within the Case or one Questions & Answers at a time minimising the number of questions that need to be answered.
- BICON Response returns the version for each question.
- BICON Response indicates when no more questions need answering.
- BICON Response indicates whether questions could not be processed.
- Can partially evaluate a Case.
- Can pass the BICON response into **RenderCase**.
- Can pass the Media requirement response into **RetrieveMediaRequirement**.

In addition this method can be used as follows:

- Use the Q&A sets returned for a Case to map to data within the third party system.
- Store the response to 'replay' historically i.e. decisions were made based on the information returned at that time.
- Store the response to 'replay' against latest version of a Case:
- Allows evaluation against stored, previously answered questions to determine whether questions have a new version.
- Can use the previously answered questions to partially evaluate even if the Case has changed – minimises user intervention.
- Retrieve a URL for the evaluated or partially evaluated Case that can be used in the BICON External web browser.
- Determine whether there are What's Changed Notices and Alerts specific to the evaluated or partially evaluated Case.



EvaluateTariffWizard

- **SearchTariffWizards** gives the external system the ability to search for Tariff Wizards by entry of a Tariff Decision Tree ID. The results of this search are Tariff Driven Search Decision Trees.
- A Tariff Wizard is used to locate the appropriate Case for a Tariff good through decision support / questions.
- The call will return all Cases in the tree as well as a list of unanswered questions and questions submitted.
- Submitting answers will reduce the number of Cases returned in the response.
- Submitting answers to all questions should produce a response with a single Case or Notice.
- When a single Case is returned then this is the appropriate Case for your Tariff good/item. You can then evaluate or render the selected Case (**EvaluateCase**).
- If a single Case or Notice is not returned after answering all questions then there may not be an appropriate Case for your Tariff good/item. Please check the returned notice details.

SaveCaseEvaluation

- Save Case Evaluation will save an in progress or complete Case evaluation in BICON for the purpose of viewing the evaluation on the BICON website.
- Use this after evaluating a Case (see EvaluateCase).
- The response will contain a URL that can be used to open the saved Case Evaluation in BICON via the BICON Website.

RetrieveMediaRequirement

RetrieveCaseDetails gives the external system the ability to search for media requirements that are returned as part of the case evaluation process.

Examples of using this method are:

- Used when searching to determine the media requirement asked for in a question in (**EvaluateCase**) such as “Is the evidence ok?”
- BICON matches against the Media requirement identifier provide and returns the content of the Media Requirements.

Render the data

Once you have searched and further interrogated the data you can:

1. Choose a Case ID and:
 - Generate a document containing all the Import Conditions within the Case (**RenderCase**).
2. Choose a Permit ID and:



- Generate a document containing the Permit Details (RetrievePermit).

Provided below is an explanation of when to use each method and the expected results.

RenderCase

Rendering a BICON Case is the process of formatting the Case and its decision tree into document format. It is recommended that Cases be rendered with at least one question and selected answer provided. This ensures that only the relevant information is rendered. To Render a Case document use the **RenderCase** service method.

- Used when generating a view of the Import Conditions.
- Pass in a Case ID and Evaluated Questions i.e. the **EvaluateCase** response from BICON and the type of output required (PDF, Word, and XML).
- BICON response returns the generated output – either a full Case (not recommended), a partially or fully evaluated Case i.e. filtered Import Conditions.

RetrievePermit

This service method will allow the calling service to retrieve a permit document. By passing in the permit ID for a permit that the user is authorised to access this method will result in a rendered Permit being sent back to the calling service.

RetrieveRenderedDocument

This service method will allow the caller to retrieve a document rendered by a prior call to **RenderCase**.

Services

This section outlines each BICON service available to third party software developers and details the following:

- **Description** – a high level description of what the method is used for.
- **Prerequisites** – details any pre-existing conditions that the system must adhere to for this method to be used.
- **Signature** – details the method of signature including the parameters and their data types.
- **Parameters** – describes each parameter, any business rules and if the parameter is mandatory.
- **Returns** – the information returned by the method on successful processing.
- **Business Rules** – details any specific business rules associated with this method and its parameters.

These methods are grouped in the following section by their order in the overall process



1. Ping
2. Searching
3. Interrogating
4. Rendering

Ping

PingCLSP

Description

This service method will return a string of information detailing a successful connection to the service or an error generated by the connection.

Prerequisites

You must have registered the user credentials using the BICON Web site for a successful connection. If the credentials supplied are not known to BICON, or have an incorrect password, then a generic error will be returned.

An example of the message is:

```
<s:Fault>
```

```
  <faultcode xmlns:a="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">a:InvalidSecurity</faultcode>
```

```
  <faultstring xml:lang="en-AU">An error occurred when verifying security for the message.</faultstring>
```

```
</s:Fault>
```

Signature

PingCLSP()

Parameters

There are no parameters required.

Returns

A String object which contains the success or failure message for the connection.

```
<PingCLSPResponse xmlns="http://bicon.agriculture.gov.au/">
```

```
  <PingCLSPResult>PingCLSP successfully called. The service is up and running. We identified you as  
  XXX.XXX@email.com.au. </PingCLSPResult>
```

```
</PingCLSPResponse>
```

Business Rules

User Credentials must be supplied and match those registered with BICON.



Searching

SearchCasesByTerms

Description

This service method will return a list of all the Cases that match the search terms in the “QueryString” parameter. The “QueryString” should contain a string of terms separated by spaces. The search terms are matched against the Cases in the BICON database and are returned in an order based on quality of the match. The search can be further restricted by using the “AsAtDate” parameter to return only Cases that are in production on the specific date. The “QueryString” accepts wildcards. A simple query with a single wildcard (*) can be used to return all Cases in the system.

Prerequisites

Before using the SearchCasesByTerms method please ensure that reference data types are up to date.

Signature

SearchCasesByTerms (

String queryString,

DateTime asAtDate,

CaseSearchTermSet searchTermSet)

Parameters

Parameter Name	Description	Mandatory
queryString	String containing the search terms to search for. Wildcards can be used within the supplied search terms. Noise words will be filtered out of the supplied search terms. All QueryString terms use “OR” searching.	Yes
asAtDate	This parameter is used to restrict the search results to only versions of items that were/are current on the supplied date. This field will default to today's date if it is not supplied.	No
searchTermSet	‘Case Titles & Synonyms’ or ‘Common Names’ or ‘Tariff Descriptions’ or ‘All Term Sets’	No

Returns

A SearchCasesByTermsResponseDataContract object which contains the data for all the cases matching the search criteria.

Business Rules

- The search parameters are AND searching.



- QueryString terms use OR searching.
- Noise words will be removed from the QueryString
- Supplied QueryString terms will be matched against a phonetic dictionary. Any phonetic matches will be added to the search as additional terms.
- For the date entered in the As at date field, find all matching cases that are effective on the entered date (the date is between the versions production and superseded date (inclusive))
- Cases will be matched on the following fields:
 - Title Match: matching one or more search terms against the Case Title
 - Case Summary: matching one or more search terms against the Case summary. The Case summary will contain the first 255 characters of the Case's content field.
 - Synonym Match: matching one or more search terms against one or more Synonyms of the case.
 - Scientific Name Match: matching one or more search terms against one or more scientific names linked to the case.
 - Common Name Match: matching one or more search terms against one or more common names linked to a scientific name linked to the case.
- Search results are ordered by Rank which is calculated based on the number of word matches (partial and full) on each search field.
- Matched Details counts the total number of matches for terms (both partial and full) for each field. This information is concatenated into a string with respective match counts displayed next to the matched field name in brackets e.g. Title (2)(0), Summary (5)(1)

SearchTariffWizards

Description

This service method will return a list of all the Tariff Search Wizards that match the supplied Tariff Code. The Cases and Notices for the Tariff Wizard can then be retrieved using the EvaluateTariffWizard method.

Prerequisites

There are no prerequisites for this service.

Signature

SearchTariffWizards (

String tariffCode,

DateTime asAtDate)



Parameters

Parameter Name	Description	Mandatory
tariffCode	String containing the Tariff code to locate the Tariff Wizards for.	Yes
asAtDate	This parameter is used to restrict the search results to only versions of items that were/are current on the supplied date. This field will default to today's date if it is not supplied.	No

Returns

A SearchTariffWizardsResponseDataContract object which contains the Tariff Driven Search Tree for the Tariff Code.

Business Rules

The search will attempt to find an exact match of the specified Tariff Code against the Tariffs in the BICON system. The matched Tariff will be the version of the Tariff that is current at the supplied AsAtDate for each related Tariff in the matched Tariff's hierarchy.

If the related Tariff has an associated Tariff Driven Search Decision Tree then the Tariff Driven Search Decision Tree will be added to the results of the search to be returned.

SearchCasesByScientificName

Description

This service method will return a list of all the Cases that match the Scientific Names in the "ScientificNames" parameter. The search can be further restricted by using the "AsAtDate" parameter to return only Cases that are in the environment on the specified date.

Signature

```
SearchCasesByScientificName (  
    String[] scientificNames,  
    DateTime asAtDate)
```

Parameters

Parameter Name	Description	Mandatory
scientificNames	List of ScientificNames to submit to the search. These can be used to match with Species, Genus, Virus Name Common Names and Synonyms.	Yes
asAtDate	This parameter is used to restrict the search results to only versions of items that were/are current on the supplied date. This field will default to today's date if it is not supplied.	No



Returns

A SearchCasesByScientificNameResponseDataContract object which contains the data for all the cases matching the search criteria.

Business Rules

There are no specific business rules for this method.

Interrogating

EvaluateTariffWizard

Description

This service method will return all the Cases and Notices for a Tariff Search Wizard. This method accepts a list of answered questions for the Tariff Driven Search Tree. The Tariff Driven Search Tree is evaluated using the supplied questions and the Case and notice elements that exist in the evaluation path are returned in the response. Calling this method with an empty “AnsweredQuestion” set will return all Case and Notice elements in the Case Search Tree as well as a list of “UnansweredQuestions”. It can then be called in an iterative fashion with the “AnsweredQuestions” set incremented each subsequent call. Note that both Tariff and Case Wizards work in a process of elimination. An Evaluate method call with no supplied answered questions will return all Cases/Notices in the tree. Subsequent method calls with answered questions submitted will reduce the number of Cases/Notices returned.

Prerequisites

There are no prerequisites for this service.

Signature

EvaluateTariffWizard (

String tariffSearchDecisionTreeId,

Long version,

QuestionDataContract[] answeredQuestions,

QuestionModeType questionMode)



Parameters

Parameter Name	Description	Mandatory
tariffSearchDecisionTreeId	String containing the unique identifier for the Tariff driven search to use of a minimum 6 characters.	Yes
version	The version of the Tariff Wizard to use. If no version is supplied the latest version of the Tariff Wizard will be used.	No
answeredQuestions	The list of answered questions to be submitted to this request. Note that the version for questions is not required when passing in AnsweredQuestions to an Evaluate or Render method as the system will always use the versions specific to the Case or Wizard version.	No
questionMode	The mode to return questions with. "All Questions" mode will return all questions in the Response regardless of the evaluation state of the Tariff Driven Search Tree. "Current Condition" mode will only return the questions for the current condition the evaluation is on. If no more Unanswered questions are returned in the response then the evaluation is complete.	No

Returns

An EvaluateTariffWizardResponseDataContract object which contains the data for all the cases matching the search criteria.

Business Rules

There are no specific business rules for this method.

EvaluateCase

Description

This service method will allow the caller to evaluate a Case without submitting a render request. An initial request with no AnsweredQuestions will trigger the start of a new evaluation. The QuestionMode controls how the evaluation is performed.

- **All Questions** – All questions for the Case will be returned. This allows the Case to be evaluated with a bulk question answer set. An initial call to this mode will return all Questions in the Case. It is then up to the caller to submit all or a subset of the questions to either a subsequent EvaluateCase request or to a RenderCase request. The All Questions mode does not differentiate between import scenarios so the caller may end up answering questions that do not relate to their import scenario.



- **Current Condition** – Current Condition mode will only return the questions up to and including the current condition that the evaluation is on. The caller can iteratively call this method to step through the evaluation of the tree. The Current Condition mode does differentiate between import scenarios so the caller will not end up answering questions that do not relate to their import scenario.

The use of the EvaluateCase method should be iterative with each subsequent service call adding to the list of answered questions. The Case evaluation is complete when the method returns no more unanswered questions.

Prerequisites

There are no prerequisites for this service.

Signature

EvaluateCase (

String caseId,

Long version,

QuestionModeType questionMode

QuestionDataContract[] answeredQuestions,

Boolean returnNotices)



Parameters

Parameter Name	Description	Mandatory
caseId	The 10 digit Case identifier provided by BICON e.g. 0000365781.	Yes
version	The numeric Case version provided by BICON. If no version is supplied the latest version of the Case will be used.	No
questionMode	The mode to return questions with. "All Questions" mode will return all questions in the Response regardless of the evaluation state of the Case tree. "Current Condition" mode will only return the questions for the current condition the evaluation is on. If no more Unanswered questions are returned in the response then the Case evaluation is complete. A RenderCase request using the AnsweredQuestions will render the Case.	Yes
answeredQuestions	The list of answered questions to be submitted to this request. Note that the version for questions is not required when passing in AnsweredQuestions to an Evaluate or Render method as the system will always use the versions specific to the Case or Wizard version.	No
returnNotices	This parameter is used to return Alerts and What's Changed notices for the Case scenario. This means that only Alerts and What's Changed notices that are relevant to the Case based on the submitted Answered questions are returned. The default for this parameter is False so that no notices are returned. It is recommended to not use this parameter unless the notices are specifically required as it will impact on performance and may cause EvaluateCase method calls to take longer.	No

Returns

An EvaluateCaseResponseDataContract object which contains the data for the evaluation of the Case. This contains the list of answered questions as well as what questions are unanswered. The questions returned are also dependant on the QuestionMode.

Business Rules

There are no specific business rules for this method.

RetrieveMediaRequirement

Description

This service method will allow the caller to retrieve a Media Requirement for the purpose of Evaluating Evidence Questions during a Bicon Case Evaluation. Evidence Questions will make no sense on their own as the Question's text may be as simple as "Is the evidence provided Ok? The Media Requirement describes the actual Requirement being fulfilled by



the evidence as well as on what physical Media (Documentation) the evidence is to be provided.

Prerequisites

There are no prerequisites for this service.

Signature

RetrieveRenderedDocument (String mediaRequirementId

Long version)

Parameters

Parameter Name	Description	Mandatory
mediaRequirementId	String containing the unique identifier of the Media Requirement as returned by QuestionDataContract	Yes
version	The version number of the Media Requirement as returned by QuestionDataContract.	Yes

Returns

A RetrieveMediaRequirementResponseDataContract object.

Business Rules

There are no specific business rules for this method.

RetrieveCaseDetails

Description

This service method will return additional details for a specific BICON Case. This includes information such as version, and whether there are any alerts or change notifications against the Case. It can also be used to determine if a Case version is current through the Case's status attribute returned in the response.

Prerequisites

There are no prerequisites for this service.

Signature

RetrieveCaseDetails (

String caseID,

Long version)



Parameters

Parameter Name	Description	Mandatory
caseID	The 10 digit Case identifier provided by BICON e.g. 0000365781.	Yes
version	The numeric Case version provided by BICON. If this is not supplied the system will find the latest version of the Case.	No

Returns

A `RetrieveCaseDetailsResponseDataContract` containing the information for the specific case. This information contains data such as version information and whether there are any alerts or change notifications against the Case.

Business Rules

There are no specific business rules for this method.

SaveCaseEvaluation

Description

This service method will allow the caller to save an evaluation in BICON. It will return a URL which can be used to open the saved evaluation on the BICON website.

Prerequisites:

There are no prerequisites for this service.

Signature

SaveCaseEvaluation (

String caseId,

Long version,

QuestionModeType questionMode

QuestionDataContract[] answeredQuestions)



Parameters

Parameter Name	Description	Mandatory
caseId	The 10 digit Case identifier provided by BICON e.g. 0000365781.	Yes
version	The numeric Case version provided by BICON. If no version is supplied the latest version of the Case will be used.	No
questionMode	The mode to return questions with. "All Questions" mode will return all questions in the Response regardless of the evaluation state of the Case tree. "Current Condition" mode will only return the questions for the current condition the evaluation is on. If no more Unanswered questions are returned in the response then the Case evaluation is complete. A RenderCase request using the AnsweredQuestions will render the Case.	Yes
answeredQuestions	The list of answered questions to be submitted to this request. Note that the version for questions is not required when passing in AnsweredQuestions to an Evaluate or Render method as the system will always use the versions specific to the Case or Wizard version.	No

Returns

A SaveCaseEvaluationResponseDataContract object which contains the URL to Open the Evaluation on the BICON Website.

Rendering

RenderCase

Description

This service method can be used to render a specified Case. The rendered output is not returned in the response. Instead, the response contains the URL of the rendered document and a case document key. The document can then be downloaded using the URL or retrieved using the document key. Note that this method is asynchronous and that the rendered output may not be available for download immediately. The URL or key can be polled to determine the availability of the Rendered document.

Prerequisites

There are no prerequisites for this service.

Signature

RenderCase (

 String CaseID,



Long Version,

BICONCaseOutputType BICONOutputType,

QuestionsDataContract[] AnsweredQuestions)

Parameters

Parameter Name	Description	Mandatory
caseID	The 10 digit Case identifier provided by BICON e.g. 000036578	Yes
version	The numeric Case version provided by BICON. If no version is supplied the latest version of the Case will be used.	No
BICONCaseOutputType	The output Type to render the Case in. PDF, Word or XML.	Yes
answeredQuestions	The questions and selected answers to submit for the rendering. Note that the version for questions is not required when passing in AnsweredQuestions to an Evaluate or Render method as the system will always use the versions specific to the Case or Wizard version.	No

Returns

A RenderCaseResponseDataContract containing the unique URL identifier to download the Rendered output file.

Business Rules

There are no specific business rules for this method.

RetrievePermit

Signature

RetrievePermit (
 String permitID)

Description

This service method will allow the caller to retrieve a permit document.

Parameters

Parameter Name	Description	Mandatory
permitID	String containing the unique permit identifier.	Yes



Returns

A RetrievePermitResponseDataContract object.

Business Rules

Registered User calling this method must have access to the permit id requested.

RetrieveRenderedDocument

Description

This service method will allow the caller to retrieve a document rendered by a prior call to RenderCase.

Signature

RetrieveRenderedDocument (

String documentKey)

Parameters

Parameter Name	Description	Mandatory
documentKey	String containing the unique document identifier, as returned by RenderCase.	Yes

Returns

A RetrieveRenderedDocumentResponseDataContract object.

Business Rules

Registered User calling this method must have access to the rendered document.

Response Data Contracts

This section outlines the data which is returned by the BICON Web Services interface. This section only includes information on the data that can be expected in Response to a web services call. All request information is covered in the Services section of this document.

Common Message Fields

Every web service operation will include a standard set of fields in the request message and a standard set of fields in the response message. These are common to every request and response message and are described below. They will not be described in the details for each web service operation.



Property Name	Data Type	Description
RequestProcessedDateTime	DateTime	The date and time when the request message was processed.

Searching

SearchCasesByTermsResponseDataContract

Description

This data contract contains the results of a Case search as returned by the SearchCasesByTerms Web service method. It contains all the returned Case and Notice elements for the search.

Elements

Property Name	Data Type	Description
QueryString	String	String containing the search terms to search for. Wildcards can be used within the supplied search terms. Noise words will be filtered out of the supplied search terms.
AsAtDate	DateTime	This parameter is used to restrict the search results to only versions of items that were/are current on the supplied date. This field will default to today's date if it is not supplied.
CaseSearchTermSet	CaseSearchTermSet	Reference Type deterring which fields to run the search on: 'Case Titles & Synonyms' or 'Common Names' or 'All Term Sets'
Cases	CaseSearchResultData Contract []	List of all the Cases that matched the supplied search criteria. This will be empty if no Cases matched the search.

SearchTariffWizardsResponseDataContract

Description

This data contract encapsulates the response to a SearchTariffWizards service method. It contains a list of all the Tariff Search Wizards that match the supplied Tariff Code.

Elements

Property Name	Data Type	Description
TariffCode	String	String containing the Tariff code to locate the Tariff Driven Search Trees for.
AsAtDate	DateTime	This parameter is used to restrict



Property Name	Data Type	Description
		the search results to only versions of items that were/are current on the supplied date. This field will default to today's date if it is not supplied.
TariffWizards	TariffSearchWizardData Contract []	This collection will contain all the Tariff Driven Search Decision Trees that matched the search parameters.

SearchCasesByScientificNameResponseDataContract

Description

This data contract contains data from the results of a Case search as returned by the SearchCasesByScientificName Web service method. It contains all the returned Case and Notice elements for the search.

Elements

Property Name	Data Type	Description
ScientificNames	String[]	List of ScientificNames that were submitted in the search request.
AsAtDate	DateTime	This parameter is used to restrict the search results to only versions of items that were/are current on the supplied date. This field will default to today's date if it is not supplied.
Cases	CaseSearchResultData Contract []	List of all the Cases that matched the supplied search criteria. This will be empty if no Cases matched the search.

Interrogating

EvaluateTariffWizardResponseDataContract

Description

This data contract encapsulates the response to a Tariff driven search request. It contains the details of the request and all matching Case and Notice elements.

Elements

Property Name	Data Type	Description
TariffSearchDecisionTreeld	String	String containing the unique identifier for the Tariff driven search used.
Version	Long	The version of the Tariff driven search to use.
Cases	CaseSearchResult DataContract []	List of all the Cases that matched the supplied search criteria. This will be



Property Name	Data Type	Description
		empty if no Cases matched the search.
QuestionMode	QuestionModeType	The mode to return questions with. "All Questions" mode will return all questions in the Response regardless of the evaluation state of the tree. "Current Condition" mode will only return the questions for the current condition the evaluation is on. If no more Unanswered questions are returned in the response then the evaluation is complete.
Notices	NoticeDataContract []	List of any matching notices for the search. This will be empty if no Notices matched the search.
AnsweredQuestions	QuestionDataContract []	The list of all questions that have been submitted to this request.
UnansweredQuestions	QuestionDataContract []	The list of questions that are yet to be answered. If this list is empty then there are no more questions to be answered.
QuestionsInError	QuestionDataContract []	If submitted questions and answers are out of date, incorrect or do not exist then they will be returned in this list.

EvaluateCaseResponseDataContract

Description

This data contract encapsulates the response to the EvaluateCase service method. It contains the data for the questions that have been submitted to the evaluation and any questions that have not been answered.

Elements

Property Name	Data Type	Description
CaseId	String	The 10 digit Case Identifier provided by BICON.
Version	Long	The version of the Case to use.
QuestionMode	Integer	The mode to return questions with. "All Questions" mode will return all questions in the Response regardless of the evaluation state of the Case tree. "Current Condition" mode will only return the questions for the current condition the evaluation is on. If no more Unanswered questions are returned in the response then the Case evaluation is complete. A



Property Name	Data Type	Description
		RenderCase request using the AnsweredQuestions will render the Case.
AnsweredQuestions	QuestionDataContract[]	The list of all questions that have been submitted to this request.
ReturnNotices	Boolean	This parameter is used to return Alerts and What's Changed notices for the Case scenario. This means that only Alerts and What's Changed notices that are relevant to the Case based on the submitted Answered questions are returned. The default for this parameter is False so that no notices are returned.
UnansweredQuestions	QuestionDataContract[]	The list of questions that are yet to be answered. If this list is empty then there are no more questions to be answered.
QuestionsInError	QuestionDataContract[]	If submitted questions and answers are out of date, incorrect or do not exist then they will be returned in this list.
Notices	NoticeDataContract[]	List of any notices that are associated to this particular Case version. Notices returned in this collection will only contain "Alert" and "What's Changed" notices. Only Alerts and What's Changed notices that are relevant to the Case based on the submitted Answered questions are returned. This list will be empty if there are no notices or the ReturnNotices parameter is not set to True.

RetrieveMediaRequirementResponseDataContract

Description

This data contract encapsulates the response to the RetrieveMediaRequirement

service method. This data contains all the Required information to display and assess a Media Requirement relating to the Import Conditions of a Bicon Case. A Media Requirement is a Requirement that must be satisfied by formal documentation known as Documentary Evidence.

Elements

Property Name	Data Type	Description
MediaRequirementId	String	The Identifier of the Media Requirement



Property Name	Data Type	Description
		returned.
Version	Long	The version number of the Media Requirement returned.
Content	ElementContentDataContract	The textual content for the Media Requirement
PreAmble	String	Preamble text used when presenting the Media Requirement on a user screen.
IsAndEvidenceRelationship	Boolean	Indicates whether the evidence required to satisfy the Requirement is cumulative of exclusive. If there are more than 1 evidence and the Media Requirement is flagged as IsAndEvidenceRelationship then all evidence must be supplied and be deemed satisfactory to fulfill the Requirement. . If there are more than 1 evidence and the Media Requirement is not flagged as IsAndEvidenceRelationship then only 1 evidence must be supplied and be deemed satisfactory to fulfill the Requirement.
MediaGroupings	MediaGroupDataContract[]	This is the list of physical Media/Documentation that all Requirement evidence is to be supplied on. An example of a Media document would be a "Veterinary Certificate".
WarningsAndInformationNotices	WarningAndInformationNoticeDataContract[]	Warning and Information notices contain additional information relating to the Requirement.

RetrieveCaseDetailsResponseDataContract

Description

This data contract encapsulates the response to the RetrieveCaseDetails service method. It contains information and metadata for the requested Case. It also contains a version history for the Case that will allow the consumer to see if there are any newer or older versions of the Case.

Elements

Property Name	Data Type	Description
CaseID	String	The 10 digit Case Identifier provided by BICON.
Version	Long	The numeric Case version provided by BICON.
CaseStatus	BICONStatus	The status of this Case. Indicates if Case is Current or Superseded or Draft.
Title	String	The Title of the Case.
ContentHtml	String	The summary content of the Case.
WarningAndInformationNotices	WarningAndInformationNoticeDataContract[]	Additional information attached to theCase.



Property Name	Data Type	Description
CaseDate	DateTime	The Date/Time that this Case was promoted.
CaseSynonyms	String[]	List of the Case Synonyms associated to this Case.
ImportDestination	string	The destination port for unloading of the goods – Australia, Cocos Island or Christmas Island. This will be a comma separated string depicting which import destinations the Case applies to. For example a Case for Cocos island and Australia would return an import destination “Cocos Island, Australia”.
Notices	CaseNoticeDataContract[]	Any notices associated with this case and version. Notices returned in this collection will only contain “Alert” and “What’s Changed” notices.
VersionHistory	CaseVersionInformationDataContract[]	List of version information of all other versions of this Case. This contains information on any previous and newer versions of the Case if they exist.

SaveCaseEvaluationResponseDataContract

Description

This data contract encapsulates the response to the SaveCaseEvaluation service method. It contains the data for the questions that have been submitted to the evaluation and any questions that have not been answered. In addition, it contains the URL for the saved evaluation which can be used to open the evaluation on the BICON website.

Elements

Property Name	Data Type	Description
CaseId	String	The 10 digit Case Identifier provided by BICON.
Version	Long	The version of the Case to use.
QuestionMode	QuestionModeType	The mode to return questions with. “All Questions” mode will return all questions in the Response regardless of the evaluation state of the Case tree. “Current Condition” mode will only return the questions for the current condition the evaluation is on. If no more Unanswered questions are returned in the response then the Case evaluation is complete. A RenderCase request using the AnsweredQuestions will render the Case.
AnsweredQuestions	QuestionDataContract[]	The list of all questions that have been submitted to this request.
UnansweredQuestions	QuestionDataContract[]	The list of questions that are yet to be answered. If this list is empty then there



Property Name	Data Type	Description
		are no more questions to be answered.
QuestionsInError	QuestionDataContract[]	If submitted questions and answers are out of date, incorrect or do not exist then they will be returned in this list.
EvaluationURL	String	The Evaluation URL uses a GUID (Globally Unique Identifier) format to provide a unique reference number for the Caseld, Version and a list of all answered questions. This URL can be used to continue an evaluation on the External Web site using the Q/A facility provided on the BICON External Web Site or to view the import conditions of a fully evaluated case.

Rendering

RenderCaseResponseDataContract

Description

This data contract encapsulates the details of a RenderCase response. This response contains the Question and answer set that the Case was rendered with as well as the URL and document key that allow the rendered Case document to be retrieved.

Elements

Property Name	Data Type	Description
Caseld	String	The 10 digit Case Identifier provided by BICON.
Version	long	The numeric Case version provided by BICON.
CaseTitle	String	The title of the BICON Case.
CaseQuestions	QuestionsDataContract[]	The question and supplied answer set that was submitted to the Render request.
CaseDocumentURL	String	The URL to download the rendered Case document from. Please note that the URL generated is unique to this request and is also temporary.
CaseDocumentKey	String	The unique key used to identify the document which can be passed into RetrieveRenderedDocument.
BICONCaseOutputType	BICONOutputType	The output Type to render in. PDF, Word or XML.



RetrieveRenderedDocumentResponseDataContract

Description

This data contract encapsulates the response to a RenderCase request. This contains the 64 bit encoded rendered document.

Elements

Property Name	Data Type	Description
Key	String	The key used to retrieve a document
DocumentRenderStatus	DocumentRenderStatus	The status of the requested rendered document. This may be Rendering, Failed or Complete. Failed rendered should be resubmitted after validating the message returned.
RenderedDocumentOutput Type	BICONOutputType	The document file type – only populated if DocumentRenderStatus is Complete
RenderedDocumentFileName	String	The (suggested) document file name – only populated if DocumentRenderStatus is Complete
RenderedDocument	Byte[]	The document file data as a 64 bit base encoded value. This is only populated if DocumentRenderStatus is Complete.

RetrievePermitResponseDataContract

Description

This data contract encapsulates the response to the RetrievePermit service method.

Elements

Property Name	Data Type	Description
PermitID	String	String containing the unique identifier for the permit retrieved.
PermitStatus	PermitStatus	Status of the retrieved permit
VariedPermitID	String	Only populated when PermitStatus is 'Varied'. When permit status is 'Varied', this is the permit id of the new permit.
PermitDocumentType	BICONOutputType	The output Type of the permit (Note: will always be PDF).
PermitDocumentFileName	String	The original filename of the permit document
PermitDocument	byte[]	The permit document in base64 encoding.



Data Contracts

This section contains details on the data contracts contained within the BICON services. For each contract it contains a description and a list of its elements.

CaseSearchResultDataContract

Description

This data contract encapsulates the details of a BICON Case as returned from a Case Search.

Elements

Property Name	Data Type	Description
CaseID	String	The 10 digit Case Identifier provided by BICON.
Version	Long	The numeric Case version provided by BICON.
CaseTitle	String	The Title of the Case.
EffectiveStartDate	DateTime	The Date/Time that this Case was made available on the web site.
EffectiveEndDate	DateTime	The Date/Time this case was superseded by a new version. This field will be null for current version results.
CaseSearchMatchRanking	Long	The search ranking is an indicator of how well the search result matches the supplied search criteria. A higher ranking indicates a better match.
MatchDetails	String	Indicates which fields were matched and whether the match is partial or full. This will be empty for scientific name search results. Matched Details counts the total number of matches for terms (both partial and full) for each field. This information is concatenated into a string with respective match counts displayed next to the matched field name in brackets full match counts first then partial match counts e.g. Title (2)(0), Description (5)(1), Synonyms (3)(0).
Notices	CaseNoticeDataContract[]	List of Alerts and What's Changed notices attached to the case. This will be empty if there are no Notices.



NoticeDataContract

Description

This data contract encapsulates the details of a BICON Notice. These refer to the BICON element group called "Notice". This includes Alert, Warning and Information ElementTypes only.

Elements

Property Name	Data Type	Description
NoticeId	String	The BICON unique identifier for this notice.
NoticeVersion	Long	The version number of this notice.
NoticeTitle	String	The title of this notice.
ContentHtml	String	The formatted html version this elements textual content.
ContentPlainText	String	The plain text version of this elements textual content.
PromotionDate	DateTime	The date this notice was promoted.
ExpiryDate	DateTime	The date on which the notice has (or will) expire and no longer apply. Only applies to Alerts and What's Changed Notices.
Type	ElementType	Enumeration value indicating the type of this notice. I.e. Alert or Information.

CaseNoticeDataContract

Description

This data contract encapsulates the details of a BICON Alert or What's Changed Notice. These special notice types are used to highlight important changes that are made to different case versions.

Elements

Property Name	Data Type	Description
AlertId	String	The BICON unique identifier for this Alert. (if this notice is of type Alert otherwise it will be null or empty)
WhatsChangedId	String	The BICON unique identifier for this Whats Changed Notice. (If this notice is of type Whats Changed Notice otherwise it will be null or empty)
Version	Long	If this Notice is an Alert then this will be the Version of the Alert element. If this Notice is a Whats Changed Notice then this will be the Version of the element this Whats Changed Notice is Attached to.



Property Name	Data Type	Description
Title	String	The title of this notice.
PromotionDate	DateTime	The date this notice was promoted.
ExpiryDate	DateTime	The date on which the notice has (or will) expire and no longer apply. Only applies to Alerts and What's Changed Notices.
Type	NoticeType	Enumeration value indicating the type of this notice. i.e. Alert or Whats Changed Notice
Content HTML	String	The content of the Alert or Whats Changed notice.

QuestionDataContract

Description

This data contract encapsulates the details of a Question. The question contains a list of all Available answers and a list of Selected answers that the question has been answered with. If the question has not been answered then the selected answers list will be empty. Note that input questions will have an empty list of AvailableAnswers but still uses this same data contract. To answer an input question, create an AnswerDataContract of the correct type and set the answers AnswerValue to the desired value. Add this question to the SelectedAnswers list. An input question can only have one SelectedAnswer.

Elements

Property Name	Data Type	Description
QuestionId	String	The 10 digit Identifier for this Question.
Version	Long	The version number for the question item. The version number is returned and populated by BICON but is not required when passing in AnsweredQuestions to an Evaluate or Render method.
QuestionText	String	The question text for this question. For example "What is the country of origin for the goods".
QuestionType	BICONQuestionType	Indicates the type of the question. This determines if the question is input or has a pre-determined set of available answers.
QuestionHelp	String	Help text for the question. Question help is provide in HTML format so that HREF links for downloadable resources can be included.
AvailableAnswers	AnswerDataContract[]	If the answer has a pre-determined set of allowable answers (Standard



Property Name	Data Type	Description
		and scientific name questions) then this property will contain all available answers for the question.
SelectedAnswers	AnswerDataContract[]	This collection contains any selected answers for the question.

AnswerDataContract

Description

This data contract encapsulates the information for a BICON question answer. If this answer is for a pre-determined answer from BICON (Standard and ScientificName questions) then no AnswerId is required and the AnswerValue must be set manually.

Elements

Property Name	Data Type	Description
AnswerId	String	The 10 digit Identifier for this answer. This is not mandatory and only applies to answers of type Scientific Name and Standard.
AnswerValue	String	The value of the answer. In the Case of Input Question answers this will be the string representation of the answer. For Standard and Scientific Name question answers this will be the Name of the answer element.
AnswerQuestionType	BICONQuestionType	The type of answer. This will refer to the type of question that this answer is for. For example Input Date, Input Number, Free Text, Standard, or Scientific Name. In the case of Standard and Scientific Name the answer will have an ID that corresponds to that answer in the system.

TariffSearchWizardDataContract

Description

This data contract encapsulates the data of a BICON Tariff Wizard (Tariff Search Decision-tree).

Elements

Property Name	Data Type	Description
TariffWizardId	String	The 10 digit Identifier provided by BICON.
Version	Long	The numeric version provided by BICON.



Property Name	Data Type	Description
TariffWizardTitle	String	The title of the Tariff Wizard.
EffectiveStartDate	DateTime	The Date/Time that this Tariff Wizard was promoted.
EffectiveEndDate	DateTime	The Date/Time this Tariff Wizard was superseded by a new version. This field will be null for current version results.

CaseVersionInformationDataContract

Description

This data contract encapsulates the details of a Case version. This includes any notices such as Change Notices and Alerts that were associated to this Case version.

Elements

Property Name	Data Type	Description
CaseID	String	The 10 digit Case Identifier provided by BICON.
Version	Long	The numeric Case version provided by BICON.
CaseStatus	BICONStatus	The status of this Case. Indicates if Case is Current or Superseded.
PromotionDate	DateTime	The Date/Time that this Case was promoted.
Notices	CaseNoticeDataContract[]	List of any notices that are associated to this particular Case version. Notices returned in this collection will only contain "Alert" and "What's Changed" notices.

ErroredQuestionDataContract

Description

This data contract encapsulates the details of a Question that has been submitted as part of a Case evaluation but has Errored. This will happen if the submitted question does not exist in the decision tree of the Case/Wizard/Procedure being evaluated. It can also happen if the submitted question answers are not compatible with the question answers of the version of the question in the Case/Wizard/Procedure being evaluated.

Elements

Elements 1

Property Name	Data Type	Description
QuestionId	String	The 10 digit Identifier for this Question.
Version	Long	The version number for the question item.



Property Name	Data Type	Description
		The version number is returned and populated by BICON but is not required when passing in AnsweredQuestions to an Evaluate or Render method.
QuestionType	BICONQuestionType	Indicates the type of the question. This determines if the question is input or has a pre-determined set of available answers.
ErrorMessage	String	The description of why this question has errored. There are two messages that can be contained in this field. 1. The Answered Question does not exist in the tree for the item being evaluated. 2. The Answers specified for the Question cannot be applied to this item. This could mean the question version is incompatible with the version of the question used in the tree for the item being evaluated.

EvidenceDataContract

Description

This data contract encapsulates the details of an Evidence. An Evidence is used in satisfying special Requirements for Bicon Case Import Conditions known as Media Requirements. An Evidence will be associated with a Question.

Elements

Property Name	Data Type	Description
Content	ElementContentDataContract	Textual information that describes the Evidence.
EvidenceQuestion	QuestionDataContract	The Bicon Case Question associated with this Evidence. Evidence Questions are flagged as IsEvidenceQuestion when Evaluating a Case.
EvidenceCriteria	EvidenceCriteriaDataContract[]	Evidence Criteria contain additional textual information for the evidence.
CriteriaConditions	EvidenceCriteriaConditionDataContract[]	Evidence Criteria Conditions contain additional textual information for the evidence.

EvidenceCriteriaDataContract

Description

An Evidence Criteria contains additional textual information relating to an Evidence in a Media Requirement.



Elements

Property Name	Data Type	Description
Content	ElementContentDataContract	The text content of the Evidence Criteria.

EvidenceCriteriaConditionDataContract

Description

An Evidence Criteria Condition contains additional textual information relating to an Evidence in a Media Requirement that is subject to some condition or circumstance.

Elements

Property Name	Data Type	Description
ConditionTitle	String	The condition or circumstance under which the Criteria applies.
CriteriaContents	ElementContentDataContract[]	The text content of the Evidence Criteria.

WarningAndInformationNoticesContract

Description

This data contract encapsulates the data of a BICON Warning or information notice

Elements

Property Name	Data Type	Description
Title	String	The title of this Warning or Information notice.
NoticeElementType	String	The version number for the reference item.
ContentHTML	String	The content of the Alert or Information notice as HTML.
ContentPlainText	String	The content of the Alert or Information notice as plain text is applicable. If not applicable this will be returned as Null.
NoticeElementType	.NoticeType	Enumeration value indicating the type of this notice. I.e. Alert or Information

MediaGroupDataContract

Description

This data contract encapsulates the details of a Media Group. A Media Group is a set of 1 or more Media (Documentation) that contain Evidence to satisfy a Media Requirement. For example the Media Requirement could be for animal vaccination, the Evidence that the animal is vaccinated is supplied on a Veterinary Certificate (Media).



Elements

Property Name	Data Type	Description
MediaGroupEvidence	EvidenceDataContract[]	The list of Evidence that may be supplied on this Media groups Media.
MediaGroupPreambleForEvidenceQuestions	String	The textual statement that describes this group's media. This is used when presenting the Media Group as part of its parent Media Requirement. The statement will be in the form of "The following Evidence must be presented on a (name of Media)."

ElementContentDataContract

Description

This data contract describes textual information for Bicon Case constructs known as elements.

Elements

Property Name	Data Type	Description
Title	String	The title of this Bicon construct or element.
ContentHtml	String	The formatted html version this elements textual content.
ContentPlainText	String	The plain text version of this elements textual content.

Reference Types

This section lists the available reference types and their values.

Searching

CaseSearchTermSet

The Case Search term set is used to specify which fields in the BICON database are used for a Case Term search.

Value	Description
TitlesAndSynonyms	Indicates that search should only match terms against the Title of Cases and against Case Synonyms metadata.
CommonNames	Indicates that search should only match Terms against Common Names associated to the Cases.
AllTermSets	Indicates that the search should match terms against all searchable fields.



Interrogating

NoticeType

Used for Case Notices. Contains Alerts and What's Changed Notices. These special notice types are used to highlight important changes that are made to different case versions.

Value	Description
Alert	Alerts contain critical information on Import Condition changes in BICON.
WhatsChangedNotice	Whats Changed Notices are used to highlight important changes made to Import Conditions.

ElementType

The BICON Element Type of an item in the BICON system.

Value	Description
Media	Media defines the acceptability criteria for documents and product labels submitted as evidence that importers have demonstrated compliance with media (assessable) requirements. The media criteria that must be satisfied during an assessment is defined in the linked media decision tree.
Activity	An activity is a task undertaken by the Biosecurity Services Group such as an inspection, treatment, test, processing, isolation, movement or destruction to manage a biosecurity risk.
Procedure	Procedure.
ExternalReference	An External Reference is a link to information contained in documents (as defined in Reference) stored externally to BICON.
File	Previously called 'Document' and referenced as such in the user requirements. A File is a link to information contained in documents (as defined in File_Resource) stored internally within BICON.
Annex	An Annex is ancillary information internal to BICON that may be referenced from elements to provide additional information integral to Cases. Annexes can also be formatted information that is required to be displayed on screen or printed (usually with permits).
Alert	Alerts contain critical information on Import Condition changes in BICON.
Warning	Warnings are similar to Alerts and contain important information in regards to BICON Import Conditions.



Information	Information elements are used to convey basic information in regards to BICON Import Conditions.
-------------	--

BICONStatus

Indicates the status of a BICON element.

Value	Description
Superseded	The BICON element was once Current but a newer version exists and has been approved. The Item is still usable/relevant but the newer version should be used instead.
Production	The BICON element is approved and is the latest approved version.
No Longer Used	The BICON Case is no longer used. This could mean that the Case has been merged into a larger Case or is no longer required.

BICONQuestionType

Indicates the type of BICON question.

Value	Description
Standard	The question has a set of available answers determined in the BICON system.
ScientificName	The question has a set of available answers however these answers correspond to scientific names.
InputDate	The question accepts 1 answer that must be a DateTime value.
InputNumber	The question accepts 1 answer that must be a Number value.

PermitStatus

Status of a BICON element.

Value	Description
Active	Permit is currently active.
Revoked	Permit has been revoked.
Suspended	Permit is suspended.
Expired	Permit has expired – either the <i>Valid To</i> date has passed, or for a single use permit, the permit has been presented.
Varied	Permit has been varied – a replacement permit has been issued.

QuestionModeType

The mode to return questions with when evaluating a tree such as a Case, Case Search Tree or Tariff Driven Search Tree



Value	Description
AllQuestions	The mode to return questions with. “All Questions” mode will return all questions in the Response regardless of the evaluation state of the Case tree.
CurrentCondition	“Current Condition” mode will only return the questions for the current condition the evaluation is on. If no more Unanswered questions are returned in the response then the Case evaluation is complete. A RenderCase request using the AnsweredQuestions will render the Case.

Rendering

BICONOutputType

Indicates what type of output the generated document should be.

Value	Description
Word	Microsoft word format
PDF	Adobe Acrobat PDF
XML	XML document of content before mark up.