



Australian Government

Department of the Environment, Water, Heritage and the Arts



www.npi.gov.au

User Guide

for

NPI Report Upload Web Service

Version 1.0

June 2008

ISBN: 978-0-642-55419-2

© Commonwealth of Australia 2008

This manual may be reproduced in whole or part for study or training purposes subject to the inclusion of an acknowledgment of the source. It may be reproduced in whole or part by those involved in estimating the emissions of substances for the purpose of National Pollutant Inventory (NPI) reporting. The manual may be updated at any time. Reproduction for other purposes requires the written permission of the Department of the Environment, Water, Heritage and the Arts, GPO Box 787, Canberra, ACT 2601, e-mail: npi@environment.gov.au, web site: www.npi.gov.au, phone: 1800 657 945.

Disclaimer

The screen shots of web pages and the NPI Online Reporting System are correct at time of publication and while all care will be taken to keep this User Guide up to date, from time to time there will be updates to the web pages and/or the online system that will change the look of some screens.

While reasonable efforts have been made to ensure the contents of this manual are factually correct, the Australian Government does not accept responsibility for the accuracy or completeness of the contents and shall not be liable for any loss or damage that may be occasioned directly or indirectly through the use of, or reliance on, the contents of this manual.

Contents

1	INTRODUCTION.....	5
1.1	WHO SHOULD READ THIS DOCUMENT?	5
1.2	WHY USE THE REPORT UPLOAD WEB SERVICE?	5
1.3	HOW EASY IS IT TO INTEGRATE THE WEB SERVICE?	5
1.4	HOW DO I REGISTER FOR THE WEB SERVICE?	5
1.5	I AM A CONSULTANT AND ONLY HAVE A REPORTER ROLE, WHAT SHOULD I DO?	5
1.6	WHY CAN'T I SUBMIT A REPORT USING THE WEB SERVICE?	5
1.7	WHAT DO I DO IF I HAVE A PROBLEM?	6
2	WEB SERVICE, SCHEMA AND WSDL URLS.....	7
3	WEB SERVICE METHODS.....	8
3.1	OVERVIEW	8
3.2	METHOD: GETFACILITIES()	8
3.2.1	<i>Request</i>	8
3.2.2	<i>Response</i>	9
3.2.3	<i>Developers Beware!</i>	10
3.2.4	<i>Example</i>	11
3.3	METHOD: GETREFERENCEDATA().....	13
3.3.1	<i>Request</i>	13
3.3.2	<i>Response</i>	13
3.3.3	<i>Example</i>	15
3.4	METHOD: UPLOADREPORT()	17
3.4.1	<i>Request</i>	17
3.4.2	<i>Response</i>	26
3.4.3	<i>Example</i>	26
4	EXCEPTIONS.....	29
	APPENDIX A – REPORT UPLOAD WSDL.....	31
	APPENDIX B – REPORT UPLOAD SCHEMA.....	34

List of Tables

Table 1 – Key to Diagrams	8
Table 2 – getFacilitiesResponse details	9
Table 3 – getReferenceDataResponse details.....	13
Table 4 – reportUpload() details	17
Table 5 – reportUpload() – reportDetails	19
Table 6 – Exception List	29

List of Figures

Figure 1 – getFacilities() definition.....	8
Figure 2 – getFacilities() response	9
Figure 3 - Historical data 'snap-shotting'	10
Figure 4 – getReferenceData definition.....	13
Figure 5 – getReferenceData() Response.....	13
Figure 6 – uploadReport() definition	17
Figure 7 – reportUpload() reportDetails.....	19
Figure 8 – Report Upload Exception	29

List of Examples

Example 1 – getFacilities() behaviour	10
Example 2 – getFacilities()	11
Example 3 – getReferenceData()	15
Example 4 – reportUpload()	26
Example 5 – Invalid user name / password combination Exception	30

1 Introduction

1.1 Who should read this document?

This document assumes the reader has a solid knowledge, understanding and experience in using web service technologies. This document provides a detailed description of the operation of the NPI Report Upload Web Service and will assist with the integration between in-house systems and the NPI Online Reporting System.

1.2 Why use the Report Upload web service?

The NPI Report Upload web service is a service provided to registered users of the NPI Online Reporting System who may wish to directly integrate their in-house systems with the NPI online system. The service allows for a report to be uploaded to the NPI online system, however users must still log in to the online reporting system to validate, certify and submit reports. Whilst this service can be used by all, this service will greatly benefit users who submit large numbers of reports per year.

1.3 How easy is it to integrate the web service?

Incorporating the Report Upload web service into your application is usually quite straight forward for experienced developers. Most modern IT languages have the ability to easily integrate with web services. Some integrated development environments (IDEs) can even build most of the required code automatically!

Please note that whilst the States/Territories and Commonwealth NPI teams provide support for the NPI Online Reporting System and the web service, it is up to you to engage appropriately qualified and experienced developers to build and/or integrate your system with our web service.

1.4 How do I register for the web service?

If you are a registered user in the online system and are a Coordinator for one or more facilities, you can use the web service to upload reports to these facilities. If you are only a Reporter for a facility, you will be unable to upload a report for that facility.

1.5 I am a consultant and only have a Reporter role, what should I do?

As at the time of writing, the NPI Implementation Working Group (IWG) has agreed that consultants may only have reporter level access to the production version of the NPI online reporting system. As a compromise, the IWG has agreed that consultants who intend to use the report upload web service may be granted coordinator level access, providing they have written approval from the facility or company.

1.6 Why can't I submit a report using the web service?

Whilst the web service allows you to upload a report, you will still need to log in to the NPI online system to complete data validation and to certify that your report is true and correct prior to submitting your report to your jurisdiction.

1.7 What do I do if I have a problem?

Support for the NPI online reporting system, calculation tools and report upload web service are available from the following sources:

1. User Guide for NPI Report Upload Web Service (i.e. this document)
2. NPI Online Reporting System – Online Help

The online help feature in the system describes the purpose of each data entry field in the user interface. The majority of fields available in the online system are available via this web service.

3. User Guide for NPI Online Reporting System

This guide contains detailed instructions for every screen in the NPI online reporting system. Whilst not directly related to this web service, this document may assist in trouble shooting some issues. This document can be found at <http://www.npi.gov.au/> under the lodging NPI reports section.

4. Jurisdiction Support

Contact your jurisdiction by using the Support menu in the NPI online reporting system. Make sure you have consulted the relevant support documentation before contacting your jurisdiction. Alternatively, jurisdiction contact details can be found at <http://www.npi.gov.au/> under the About Us - > Contacts section.

5. Commonwealth Support

If your jurisdiction is unable to help you with your support request, they will contact the Commonwealth. The Commonwealth may contact you directly if required.

Please note that whilst the States/Territories and Commonwealth NPI teams provide support for the online reporting system and the web service, is it up to you to engage appropriately qualified and experienced developers to build and/or integrate your system with our web service.

2 Web Service, Schema and WSDL URLs

The NPI report Upload Web Service URL is:

https://npi.environment.gov.au/webservice/upload_service

The most up-to-date schema and definition of the web service can be found at the following URLs.

WSDL:

https://npi.environment.gov.au/webservice/upload_service?wsdl

XSD:

https://npi.environment.gov.au/webservice/upload_service?xsd=1

For quick reference, Appendix A and Appendix B contain copies of these documents as at the time of writing.

3 Web Service methods

3.1 Overview

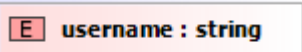
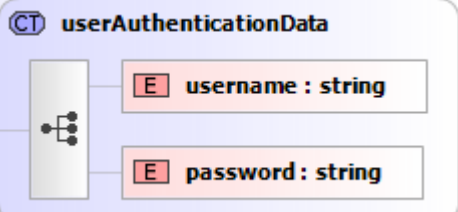
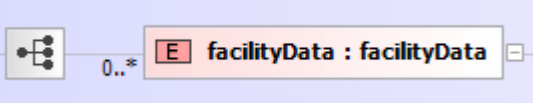
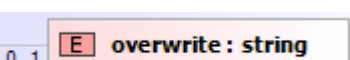
There are three methods provided by the report upload service and all must be used to successfully upload a report. The three methods are:

- getFacilities();
- getReferenceData(); and
- uploadReport().

These methods are described in detail in the following three sections. Each section contains a description of the method in terms of request, response and also provides an example.

Whilst the diagrammatic notations and definitions may be familiar to developers, Table 1 contains brief descriptions for the benefit of all.

Table 1 – Key to Diagrams

	A simple (type) Element. The diagram states the name of the element and its data type.
	A complex type. This is an 'element' that contains multiple simple elements.
 	If not specified explicitly in a diagram (as shown on left), the cardinality is 1..1. An element with cardinality 1..1 is a mandatory element that will always contain or required data.

3.2 Method: getFacilities()

The getFacilities() method returns a list of facilities and reports that the user coordinates. This is the first method that a web service client should call.

3.2.1 Request

Figure 1 shows the definition of the getFacilities method. All calls to the NPI Report Upload Web Service require authentication data in the form of your system user name and password.

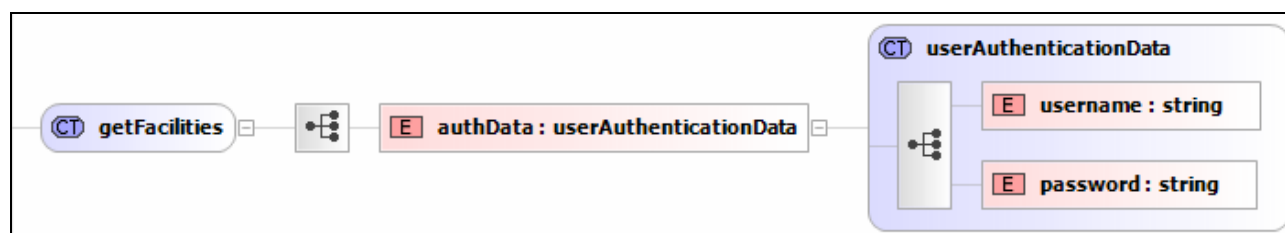


Figure 1 – getFacilities() definition

3.2.2 Response

On successful completion of the getFacilities() method, you will receive the getFacilitiesResponse message (see Figure 2). As can be seen in Figure 2, this message contains a list of facilities associated to the user credentials passed to the getFacilities() method. Each facilityData element contains a list of reports that are either Uncommenced or Draft¹. Table 2 describes the response in more detail.

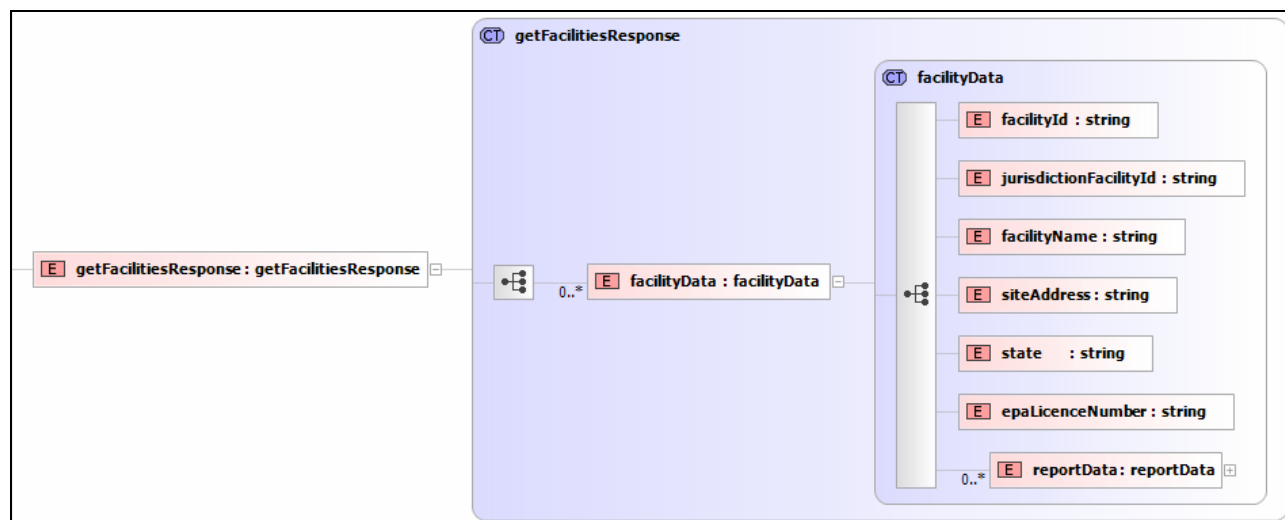


Figure 2 – getFacilities() response

Table 2 – getFacilitiesResponse details

element	description	
E facilityId : string	This is the system generated Id for your facility. Please note that due to historical data 'snap-shooting', this ID does not always remain constant. See the following section for details.	
E jurisdictionFacilityId : string	The jurisdiction assigned facility Id. This is the facility Id that is assigned to your facility by the jurisdiction.	
E facilityName : string	This is the name of your facility. If your facility does not have a name, this will be the registered company name of your facility.	
E siteAddress : string	This is the concatenated site address of your facility. Please note that it does not contain the state/territory details as this data is provided as a separate element.	
E state : string	This is the state or territory in which your facility is located.	
E epaLicenceNumber : string	This is the licence number assigned to the facility by your jurisdiction. The use of this value may vary depending on the jurisdiction in which your facility is located.	
CT reportData E reportingYearId : string E reportingYearName : string	The reportData complex type is used to provide a list of Uncommenced and Draft reports for your facility.	
	reportingYearId	This is the identifier for a reporting year.
	reportingYearName	The name of the reporting year is (for example) "07/08".

¹ Please refer to the User Guide for NPI Online Reporting System for more information on Uncommenced and Draft reports.

3.2.3 Developers Beware!

Although unlikely, a Facility's facilityId may change in the system over time and therefore you should always use the getFacilities() method to obtain the correct facilityId.

As mentioned in Table 2, due to historical data 'snap-shotting', a report's system generated facilityId changes once the report is submitted to a jurisdiction (see Figure 3).

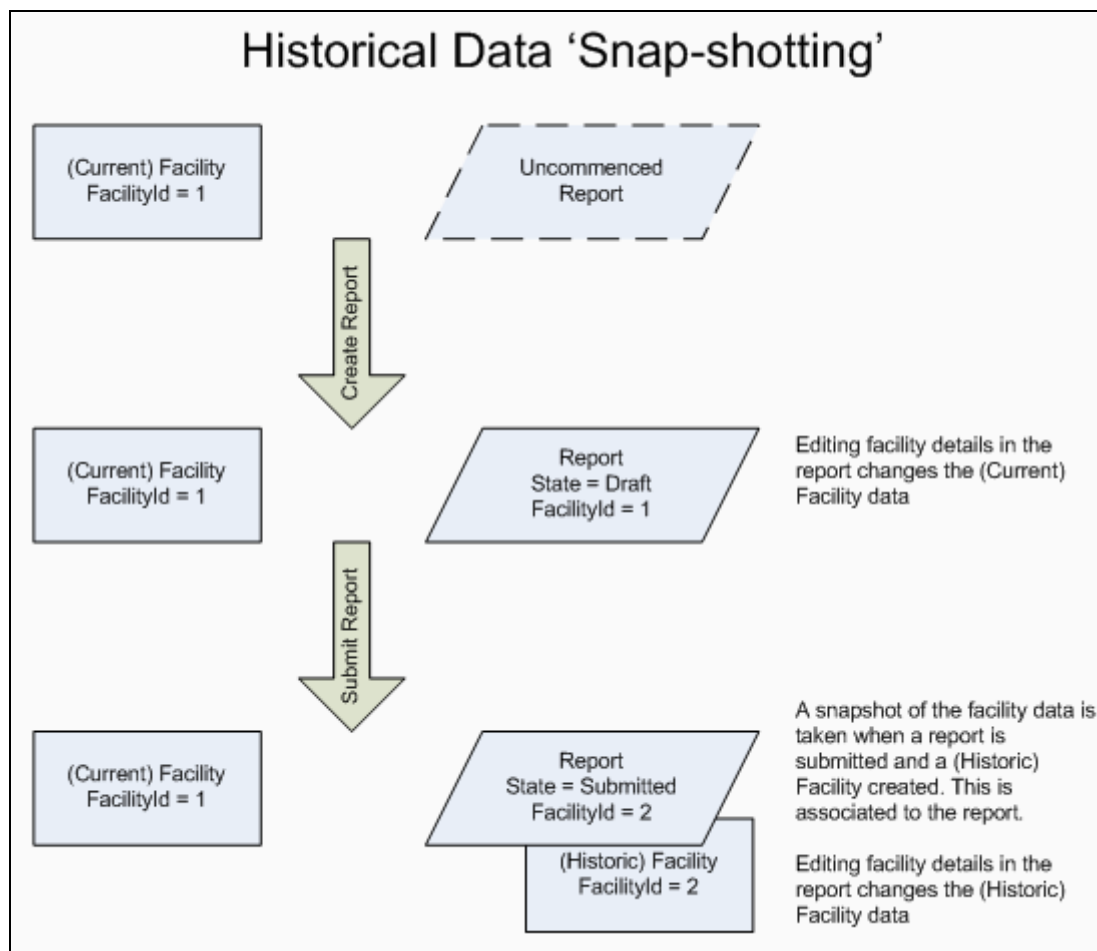


Figure 3 - Historical data 'snap-shotting'

There is one scenario where this behaviour could cause problems in an incorrectly implemented web service client. If a report is submitted and then later rejected by a jurisdiction, the report's facilityId will not match the current Facility's facilityId. In this scenario the getFacilities() method returns both the current and historic facilities as separate entities. See below:

```
<?xml version="1.0" encoding="utf-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ns1="http://services.report.ws.lucinda.apl.env.gov.au/">
  <soapenv:Body>
    <ns1:getFacilitiesResponse>
      <facilityData>
        <facilityId>1</facilityId>
        <jurisdictionFacilityId>ACT999</jurisdictionFacilityId>
        <facilityName>My Facility</facilityName>
        <siteAddress></siteAddress>
        <state>ACT</state>
```

Example 1 – getFacilities() behaviour

```

    <epaLicenceNumber></epaLicenceNumber>
  </facilityData>
  <facilityData>
    <facilityId>2</facilityId>
    <jurisdictionFacilityId>ACT999</jurisdictionFacilityId>
    <facilityName>My Facility</facilityName>
    <siteAddress>1 MyStreet MySuburb 2601</siteAddress>
    <state>ACT</state>
    <epaLicenceNumber>My Licence Number</epaLicenceNumber>
    <reportData>
      <reportingYearId>11</reportingYearId>
      <reportingYearName>07/08</reportingYearName>
    </reportData>
  </facilityData>
</ns1:getFacilitiesResponse>
</soapenv:Body>
</soapenv:Envelope>

```

Example 1 – getFacilities() behaviour contd...

It is therefore recommended that when developing web service clients, developers approach the user interface design/implementation for selecting a report with a “report has a facilityId” as opposed to “facility has reports” concept.

3.2.4 Example

The following SOAP message executes the getFacilities() web service method.

```

<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:SOAP-
  ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <tns:getFacilities
      xmlns:tns="http://services.report.ws.lucinda.apl.env.gov.au/"
      xmlns:tnsa=http://schemas.xmlsoap.org/soap/encoding/
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <authData>
        <username>UserName</username>
        <password>Password</password>
      </authData>
    </tns:getFacilities>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Example 2 – getFacilities()

The response to the above request is shown below:

```
<?xml version="1.0" encoding="utf-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ns1="http://services.report.ws.lucinda.apl.env.gov.au/">
  <soapenv:Body>
    <ns1:getFacilitiesResponse>
      <facilityData>
        <facilityId>123456</facilityId>
        <jurisdictionFacilityId>ACT999</jurisdictionFacilityId>
        <facilityName>My Facility</facilityName>
        <siteAddress>1 MyStreet MySuburb 2601</siteAddress>
        <state>ACT</state>
        <epaLicenceNumber>My Licence Number</epaLicenceNumber>
        <reportData>
          <reportingYearId>11</reportingYearId>
          <reportingYearName>07/08</reportingYearName>
        </reportData>
      </facilityData>
    </ns1:getFacilitiesResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Example 2 – getFacilities() Response

3.3 Method: *getReferenceData()*

As with many IT systems, reference data (e.g. Reduction activities) can change over time. The *getReferenceData()* method provides a listing of all current reference data in order for you to upload a report. You will need to call this method to obtain the data identifiers required for the *uploadReport()* method.

3.3.1 Request

The *getReferenceData()* method call is similar to the *getFacilities()* method in that you only need to provide authentication details.

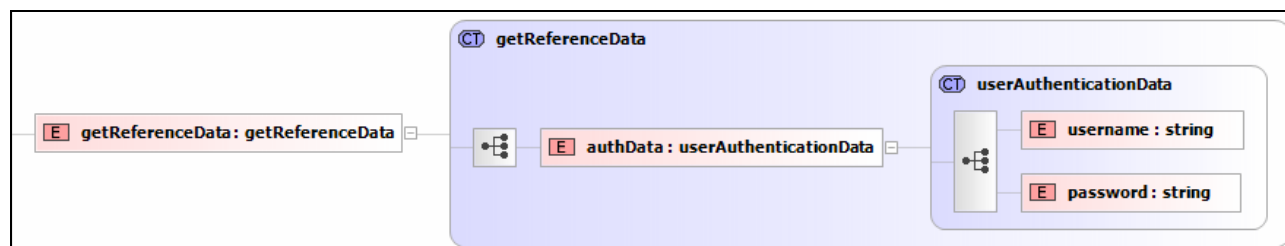


Figure 4 – *getReferenceData* definition

3.3.2 Response

Figure 5 shows the structure of the *getReferenceData()* response. Table 3 describes the individual components that make up this response.

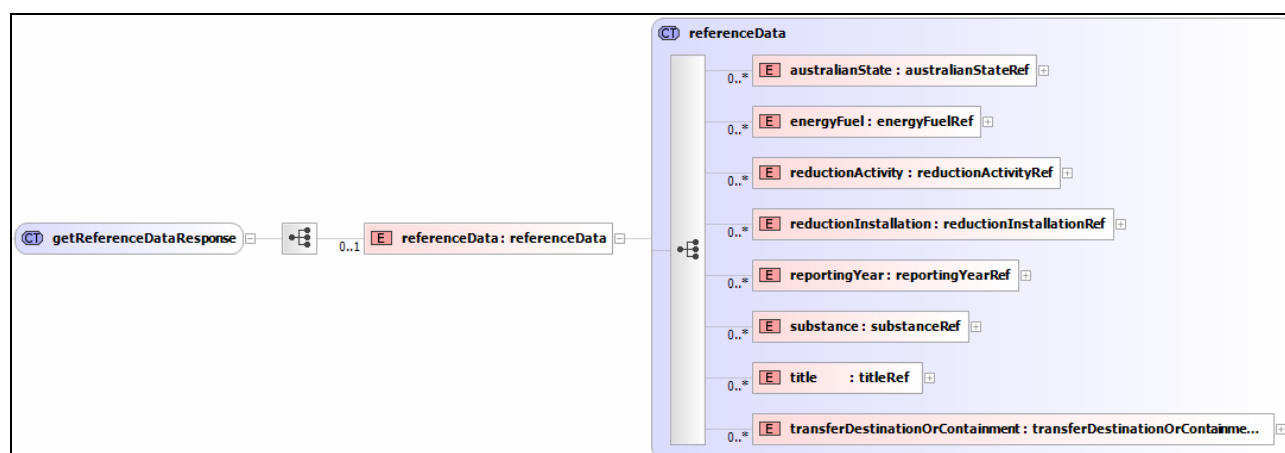
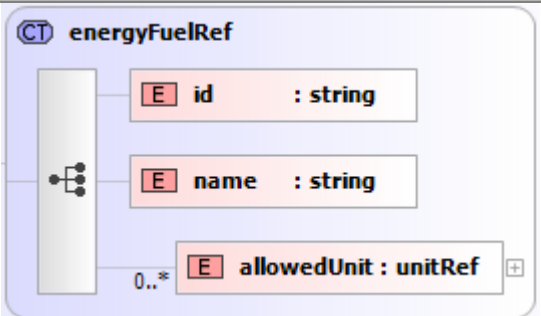
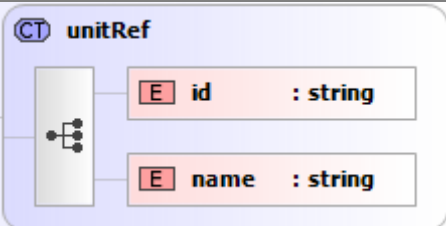
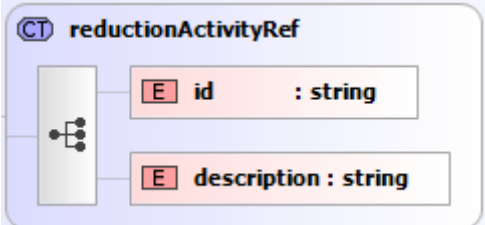
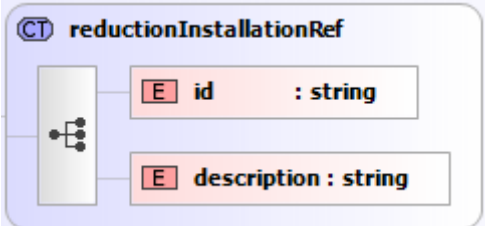
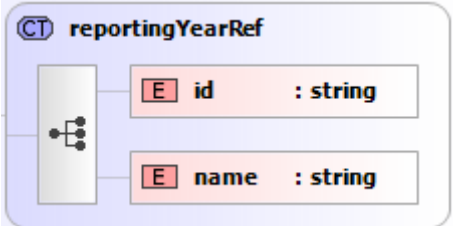
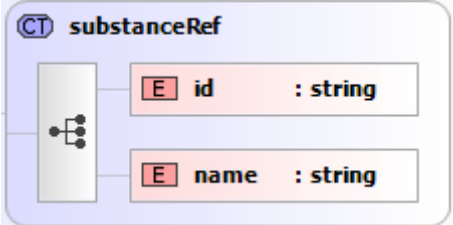
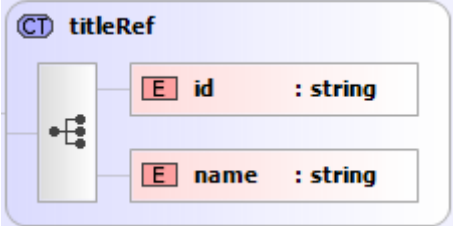
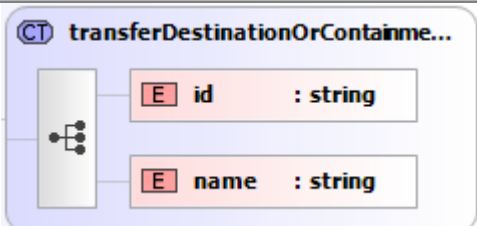


Figure 5 – *getReferenceData()* Response

Table 3 – *getReferenceDataResponse* details

element	description
	australianStateRef holds the state/territory identifier and name.

element	description
	This element holds a fuel/energy source and a list of allowed units of measure (e.g. kg, t, L, ML etc). Values not already in tonnes (e.g. Litres) will be automatically converted using standard conversions when a report is uploaded. The assumptions used in these conversions are published in the NPI Guide. If these assumptions do not accurately represent your process/fuel characteristics (e.g. density), it is best that you calculate and provide the fuel use in tonnes.
	This element defines the allowed unit(s) for a specific fuel. These are standard SI units.
	This element is used to provide definitions for Pollution reduction activities
	This element is used to provide definitions for Pollution control devices.
	This element is used to provide a list of valid reporting years. Note that getReferenceData() lists all reporting years available in the system.
	This element is used to provide a list of the 93 substances used in the NPI.
	This element is used to provide the definitions for valid titles (e.g. Mr, Mrs, Miss etc)

element	description
	This element is used provide the definitions for destination and containment types used for transfers.

3.3.3 Example

The following SOAP message can be used to execute the getReferenceData() web service method

```
<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <tns:getReferenceData
      xmlns:tns="http://services.report.ws.lucinda.apl.env.gov.au/"
      xmlns:tnsa=http://schemas.xmlsoap.org/soap/encoding/
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <authData>
        <username>UserName</username>
        <password>Password</password>
      </authData>
    </tns:getReferenceData>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Example 3 – getReferenceData()

For clarity, the below response to getReferenceData() is a partial extract. The full response to this method will contain a much larger data set.

```
<?xml version="1.0" encoding="utf-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ns1="http://services.report.ws.lucinda.apl.env.gov.au/">
  <soapenv:Body>
    <ns1:getReferenceDataResponse>
      <referenceData>
        <australianState>
          <id>ACT</id>
          <name>Australian Capital Territory</name>
        </australianState>
        <energyFuel>
          <id>1</id>
          <name>LPG</name>
          <allowedUnit>
            <id>2</id>
            <name>t</name>
          </allowedUnit>
        </energyFuel>
        <reductionActivity>
          <id>21</id>
          <description>Improved maintenance scheduling, record keeping, or
procedures</description>
        </reductionActivity>
        <reductionInstallation>
          <id>6</id>
          <description>Cyclone/multiclone</description>
        </reductionInstallation>
        <reportingYear>
          <id>11</id>
          <name>07/08</name>
        </reportingYear>
        <substance>
          <id>1</id>
          <name>Acetaldehyde</name>
        </substance>
      </referenceData>
    </ns1:getReferenceDataResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Example 3 – getReferenceData() Response

```
</substance>
<title>
  <id>1</id>
  <name>Mr</name>
</title>
<title>
  <id>4</id>
  <name>Mrs</name>
</title>
<transferDestinationOrContainment>
  <id>1</id>
  <name>Landfill</name>
</transferDestinationOrContainment>
</referenceData>
</ns1:getReferenceDataResponse>
</soapenv:Body>
</soapenv:Envelope>
```

Example 3 – getReferenceData() Response contd...

3.4 Method: uploadReport()

As its name suggests, the uploadReport() method is the method used to upload a NPI report into the NPI Online Reporting System. This will generally be the 3rd method call that web service clients execute.

3.4.1 Request

The upload report method requires you to send an entire report in one transaction and therefore needs to contain authentication, facility and report data. Figure 6 shows the required information for this method. Table 4 describes each component of the method separately.

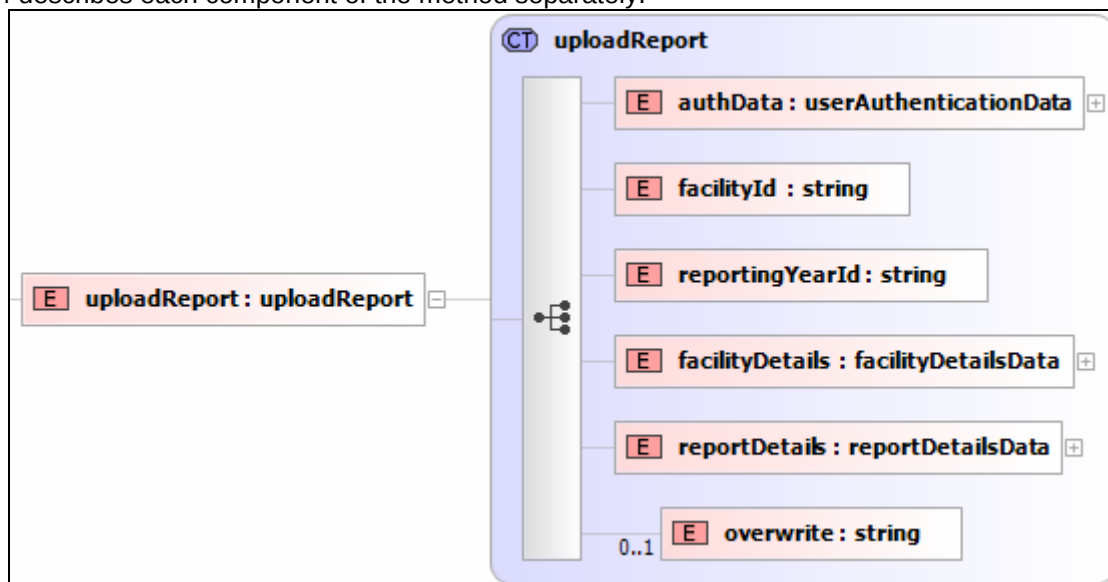
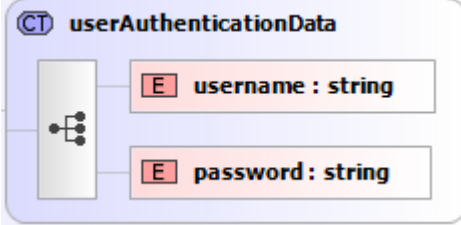
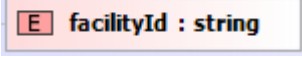
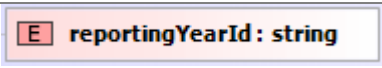
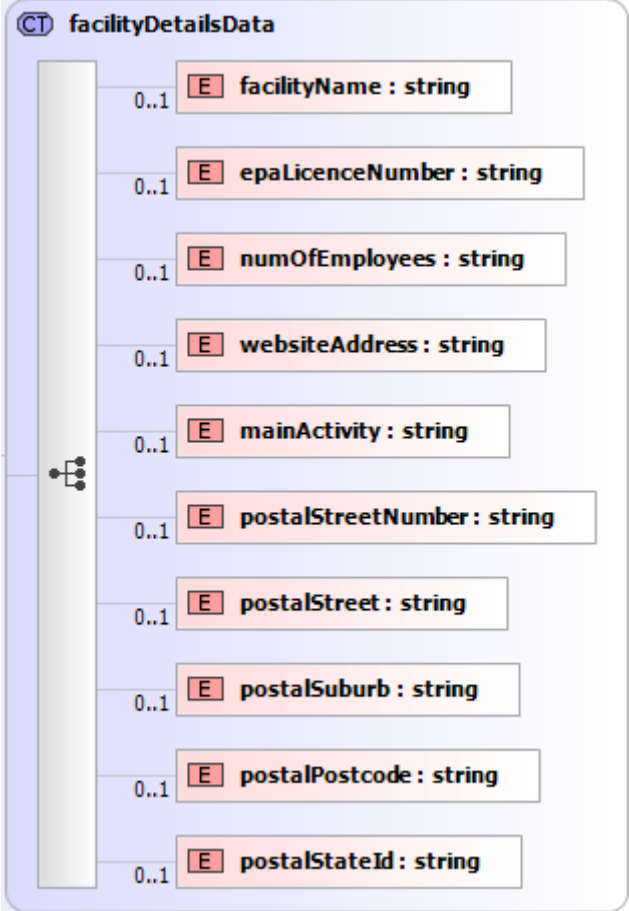


Figure 6 – uploadReport() definition

Table 4 – reportUpload() details

element	description
	The authentication data element is as per the getFacilities() and getReferenceData() methods. Please refer to sections 3.3 and 3.4 for further details.
	The facilityId element identifies the facility to which the report belongs. This ID is the system generated ID obtained using the getFacilities() method and not the identifier assigned to a facility by a Jurisdiction. Please ensure you have read and understand section 3.2.3 Developers Beware!
	The reportingYearId element identifies the reporting year to which the report belongs. This ID is the system generated ID that is returned by the getReferenceData() method, specifically from the reportingYear element.

element	description
 The diagram shows a container element CT facilityDetailsData containing several child elements, each with a cardinality of 0..1: E facilityName : string E epaLicenceNumber : string E numOfEmployees : string E websiteAddress : string E mainActivity : string E postalStreetNumber : string E postalStreet : string E postalSuburb : string E postalPostcode : string E postalStateId : string	The facilityDetails element holds facility specific data that can be updated as part of the report submission process. facilityName – The name of your facility epaLicenceNumber – The licence number assigned to the facility by the jurisdiction. <i>Note:</i> The use of this value may vary depending on the jurisdiction in which the facility is located. numOfEmployees – The number of employees/contractors at your facility websiteAddress – Your website address including the “http://” prefix mainActivity – The main activity that occurs at the facility Facility Postal Address – Street Number Facility Postal Address – Street Name Facility Postal Address – Suburb/Town Facility Postal Address – Postcode Facility Postal Address – StateId. This Id is obtained from the response of the getReferenceData() method.
E reportDetails : reportDetailsData	The reportDetails element holds report specific data. Please see Figure 7 & Table 5 below for specific details
E overwrite : string	The overwrite flag is used to inform the NPI system that an existing report should be overwritten. Valid values are ‘true’ or ‘false’. <i>Note:</i> When overwriting, <u>all</u> fields are overwritten when the ‘new version’ of the report is uploaded.

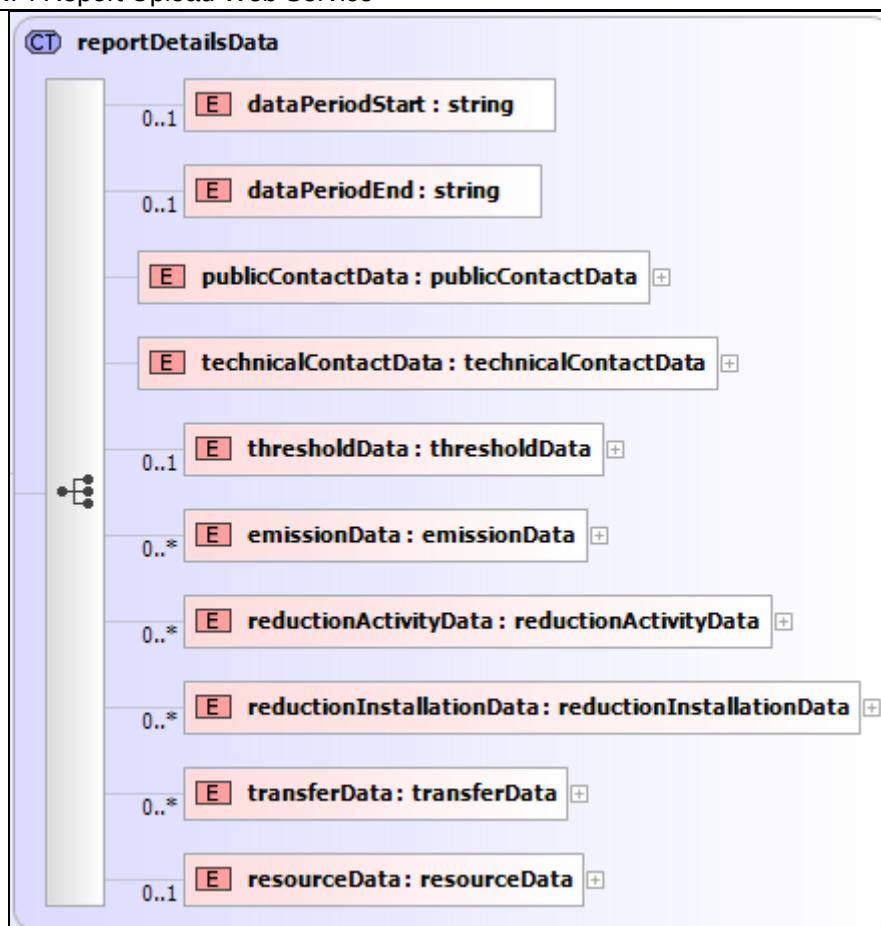
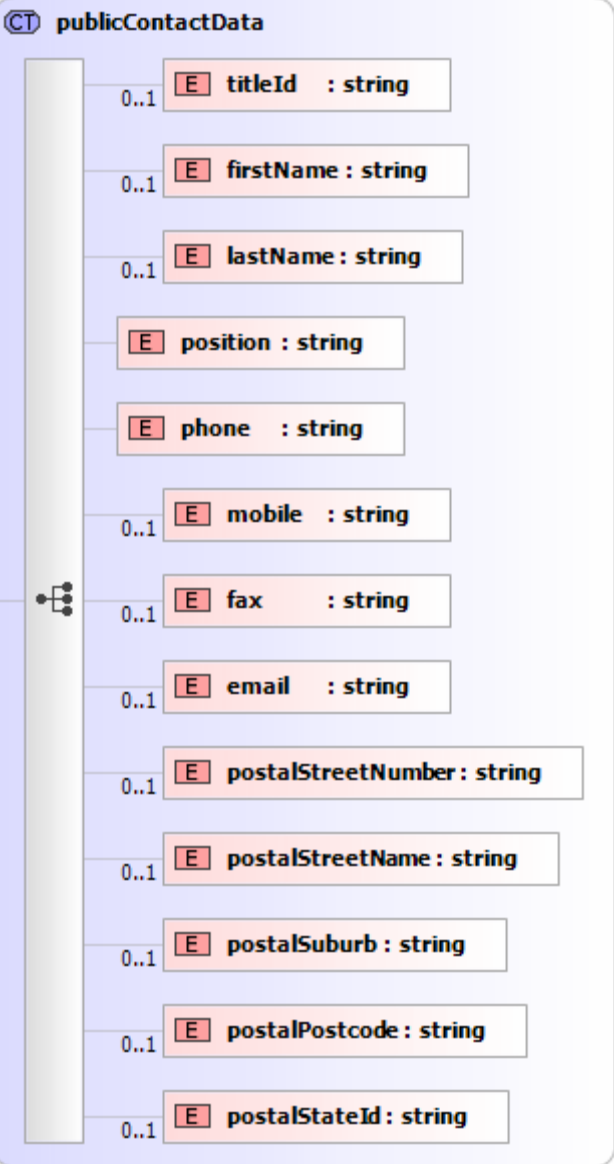
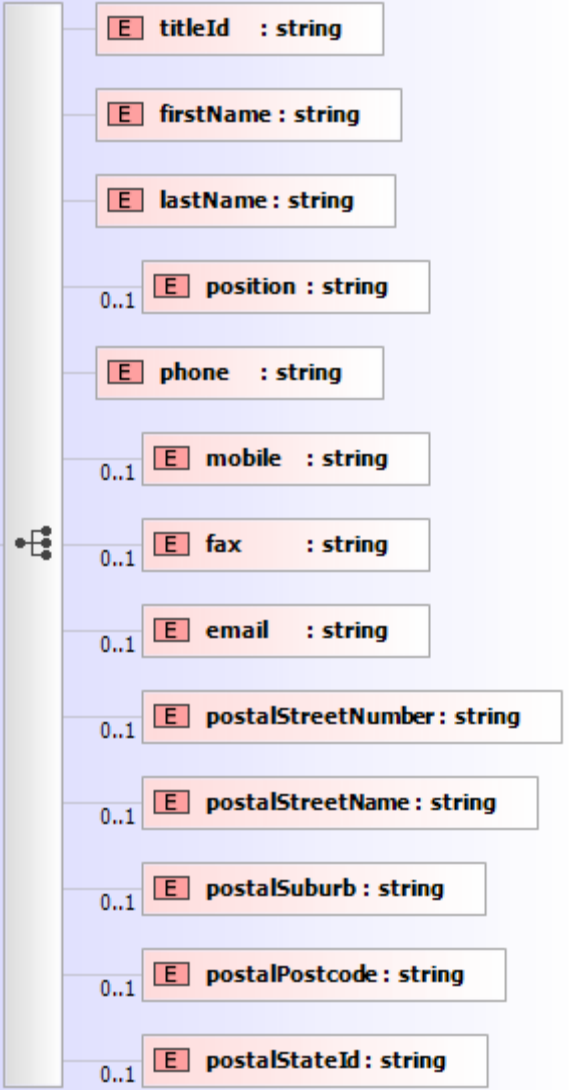
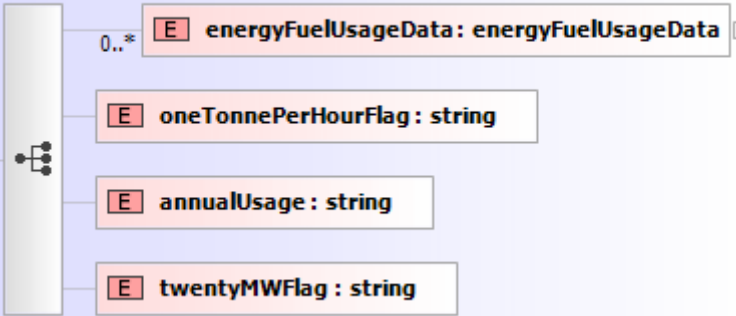


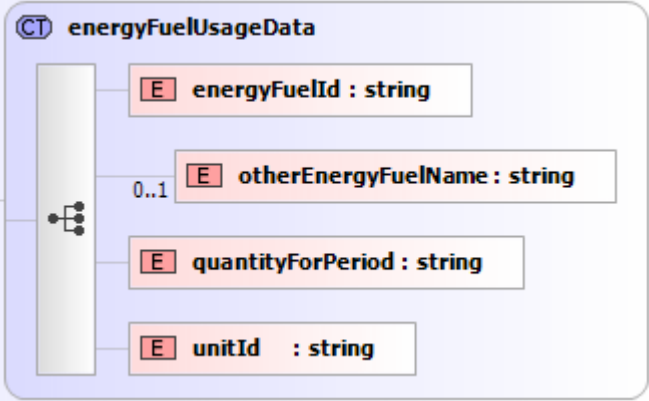
Figure 7 – reportUpload() reportDetails

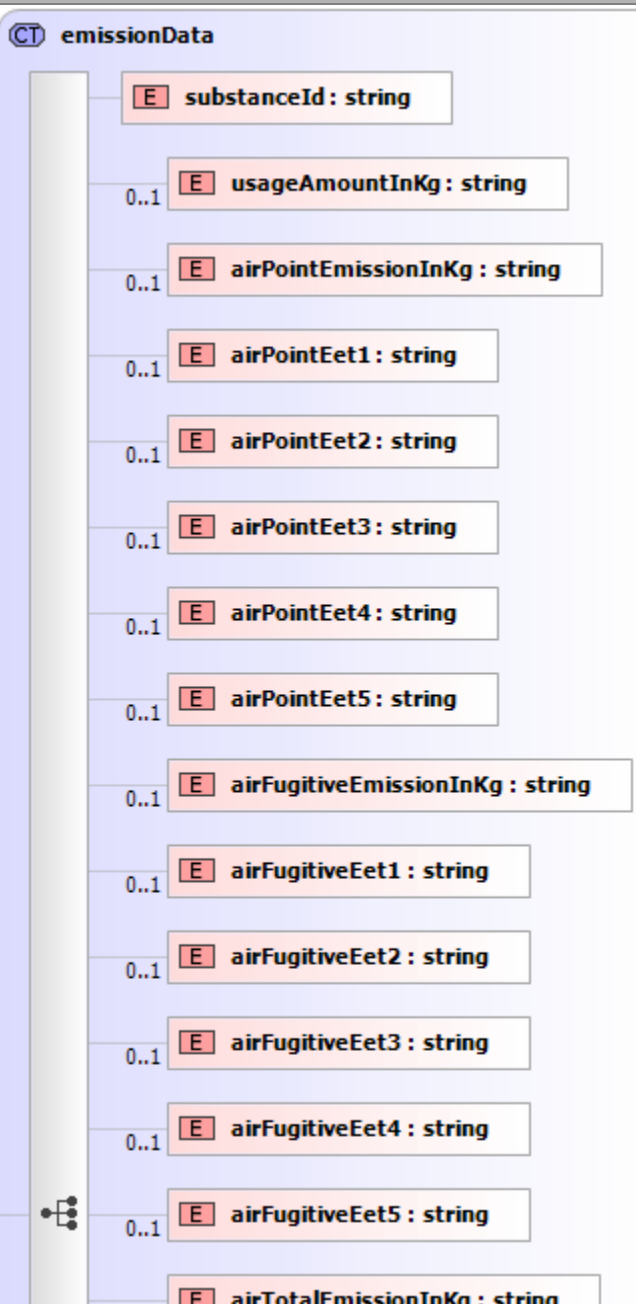
Table 5 – reportUpload() – reportDetails

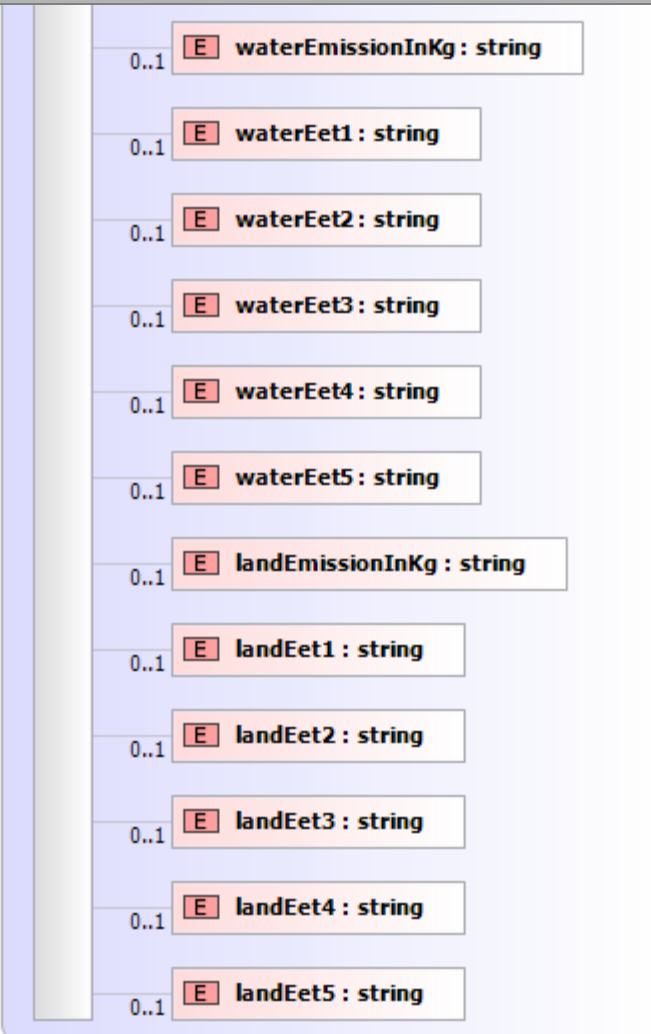
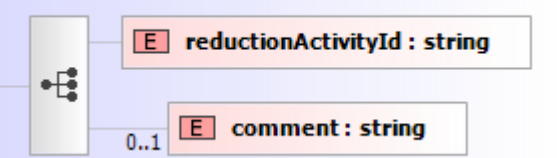
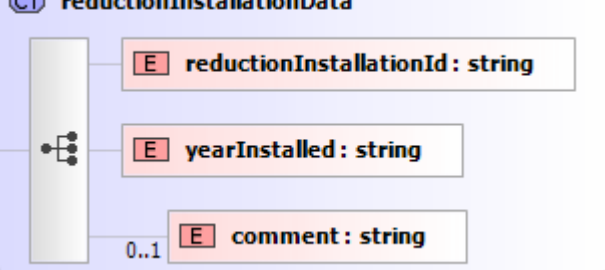
element	description
E dataPeriodStart : string	The start date of your data collection period. For the majority of reporters, this will be the start date of your reporting period.
E dataPeriodEnd : string	The end date of your data collection period. For the majority of reporters, this will be the end date of your reporting period.

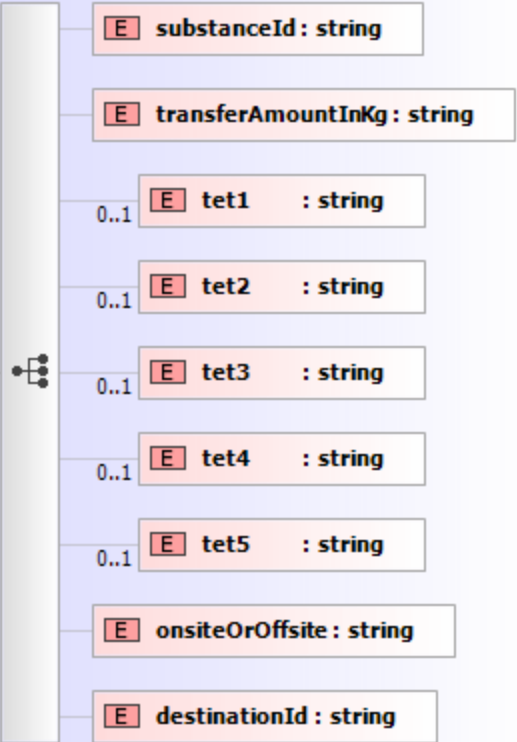
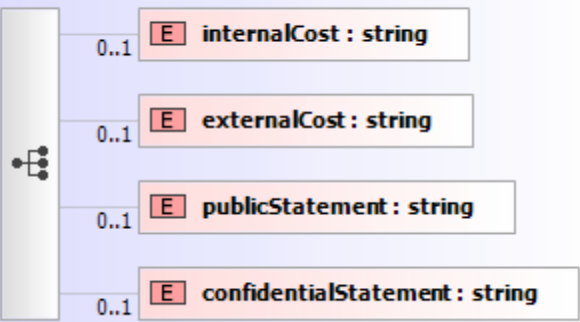
element	description
CT publicContactData  The diagram shows a vertical container for the publicContactData element. It contains a list of fields, each with a red 'E' icon, a cardinality of '0..1', and a data type of 'string'. The fields are: titleId, firstName, lastName, position, phone, mobile, fax, email, postalStreetNumber, postalStreetName, postalSuburb, postalPostcode, and postalStateId. A small tree icon is visible on the left side of the container.	Public Contact Details <i>Note: position and phone are both mandatory fields.</i>

element	description
CT technicalContactData  The diagram shows a vertical bar on the left with a connector icon. To its right are several fields, each in a box labeled 'E' (Element): titleId : string firstName : string lastName : string position : string (with cardinality 0..1) phone : string mobile : string (with cardinality 0..1) fax : string (with cardinality 0..1) email : string (with cardinality 0..1) postalStreetNumber : string (with cardinality 0..1) postalStreetName : string (with cardinality 0..1) postalSuburb : string (with cardinality 0..1) postalPostcode : string (with cardinality 0..1) postalStateId : string (with cardinality 0..1)	Technical Contact Details <i>Note:</i> titleId, firstName, lastName, phone are mandatory fields.
CT thresholdData  The diagram shows a vertical bar on the left with a connector icon. To its right are several fields, each in a box labeled 'E' (Element): energyFuelUsageData : energyFuelUsageData (with cardinality 0..* and an expand icon) oneTonnePerHourFlag : string annualUsage : string twentyMWFlag : string	energyFuelUsageData – see below for details. oneTonnePerHourFlag – Has your facility burned one tonne of fuel or waste in any hour during the reporting period? Valid values are “true” or “false” annualUsage – Annual electricity usage in MW-hr twentyMWFlag – Is your facility rated at 20MW (megawatts) or higher? Valid values are “true” or “false”

element	description
 The diagram shows the structure of the energyFuelUsageData element. It is a container element (CT) that contains four child elements, all of which are required (indicated by a red 'E' in a box): energyFuelId : string otherEnergyFuelName : string (with a cardinality of 0..1) quantityForPeriod : string unitId : string	energyFuelId – This Id is obtained from the getReferenceData() response. otherEnergyFuelName – enter fuel name here when the FuelId is set to “Other” quantityForPeriod – amount of fuel used in the reporting period unitId – This Id is obtained from the getReferenceData() response. And corresponds to the unit of measurement (e.g. Litres, cubic metres etc) for the above mentioned quantity of fuel used.

element	description
CT emissionData  The diagram shows the structure of the emissionData element. It is a container for several sub-elements, each represented by a box with a red 'E' icon and a text label. The sub-elements are: substanceId : string (required, 1..1) usageAmountInKg : string (optional, 0..1) airPointEmissionInKg : string (optional, 0..1) airPointEet1 : string (optional, 0..1) airPointEet2 : string (optional, 0..1) airPointEet3 : string (optional, 0..1) airPointEet4 : string (optional, 0..1) airPointEet5 : string (optional, 0..1) airFugitiveEmissionInKg : string (optional, 0..1) airFugitiveEet1 : string (optional, 0..1) airFugitiveEet2 : string (optional, 0..1) airFugitiveEet3 : string (optional, 0..1) airFugitiveEet4 : string (optional, 0..1) airFugitiveEet5 : string (optional, 0..1) airTotalEmissionInKg : string (optional, 0..1) A vertical bar on the left side of the diagram indicates the sequence of elements. A small icon at the bottom left of the diagram indicates that the elements are optional.	EmissionData is where the emission of a substance to all destinations is added to the report. substanceId – This Id is obtained from the getReferenceData() response All “EET” elements have valid values of “true” or “false” For quick reference, the EETs (for all destinations) are as follows: - EET1 – Mass Balance - EET2 – Engineering Calculations - EET3 – Direct Measurement - EET4 – Emission Factors - EET5 – Approved Alternative

element	description
	contd...
CT reductionActivityData 	Emission reduction Activity(s) reductionActivityId: This Id is obtained from the getReferenceData() response. Comment: optional comment about the activity
CT reductionInstallationData 	Emission Control Device(s) reductionInstallationId: This Id is obtained from the getReferenceData() response. yearInstalled: the four digit year of installation comment: optional comment about the equipment

element	description
CT transferData  The diagram shows a vertical bar on the left with a connector icon. To its right are several elements, each in a box: substanceId : string, transferAmountInKg : string, tet1 : string (with 0..1 to the left), tet2 : string (with 0..1 to the left), tet3 : string (with 0..1 to the left), tet4 : string (with 0..1 to the left), tet5 : string (with 0..1 to the left), onsiteOrOffsite : string, and destinationId : string.	transferData is where the transfer of a substance is added to the report. substanceId: The Id is obtained from the getReferenceData() response. transferAmountInkg: the amount of substance (kg) that is transferred. tet1: transfer estimation technique 1 (mass balance) Valid values are "true" or "false" tet2: transfer estimation technique 2 (engineering calculations). Valid values are "true" or "false" tet3: transfer estimation technique 3 (direct measurement) Valid values are "true" or "false" tet4: transfer estimation technique 4 (transfer factors) Valid values are "true" or "false" tet5: transfer estimation technique 5 (approved alternative method) Valid values are "true" or "false" onsiteOrOffsite: valid values are "onsite" or "offsite" destinationId: the Id corresponding to the destination of the transfer. The Id is obtained from the getReferenceData() response.
CT resourceData  The diagram shows a vertical bar on the left with a connector icon. To its right are four elements, each in a box: internalCost : string (with 0..1 to the left), externalCost : string (with 0..1 to the left), publicStatement : string (with 0..1 to the left), and confidentialStatement : string (with 0..1 to the left).	internalCost: Internal staff costs incurred by the facility in preparing the report. Please do not include a "\$" sign, just the monetary value. For example if the value was \$300, enter 300 into this field. externalCost: External contractors/consultant costs incurred in preparing the report. Please do not include a "\$" sign, just the monetary value. publicStatement: A statement that is visible to the public once the report is published. confidentialStatement: A statement that is visible only to NPI staff.

3.4.2 Response

The response from a successful execution of the uploadReport() method is a simple Boolean value where 'true' indicates the success. Please refer to Section 4 for details regarding exceptions associated with an unsuccessful execution of uploadReport().



3.4.3 Example

Example 4 below shows a SOAP message that uploads a basic report.

```

<?xml version="1.0" encoding="utf-8"?>
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
  <tns:uploadReport
    xmlns:tns="http://services.report.ws.lucinda.apl.env.gov.au/"
    xmlns:tnsa="http://schemas.xmlsoap.org/soap/encoding/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <authData>
      <username>UserName</username>
      <password>Password</password>
    </authData>
    <facilityId>123456</facilityId>
    <reportingYearId>11</reportingYearId>
    <facilityDetails>
      <facilityName>My Facility</facilityName>
      <epaLicenceNumber>My Licence Number</epaLicenceNumber>
      <numOfEmployees>1</numOfEmployees>
      <websiteAddress>http://www.npi.gov.au</websiteAddress>
      <mainActivity>Uploading Reports</mainActivity>
      <postalStreetNumber>1</postalStreetNumber>
      <postalStreet>MyStreet</postalStreet>
      <postalSuburb>MySuburb</postalSuburb>
      <postalPostcode>2601</postalPostcode>
      <postalStateId>ACT</postalStateId>
    </facilityDetails>
    <reportDetails>
      <dataPeriodStart>1/07/2007</dataPeriodStart>
      <dataPeriodEnd>31/06/2008</dataPeriodEnd>
      <publicContactData>
        <titleId></titleId>
        <firstName></firstName>
        <lastName></lastName>
        <position>Manager</position>
        <phone>(02) 1234 5678</phone>
        <mobile></mobile>
        <fax></fax>
        <email></email>
      </publicContactData>
    </reportDetails>
  </tns:uploadReport>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Example 4 – reportUpload()

```

        <postalStreetNumber></postalStreetNumber>
        <postalStreetName></postalStreetName>
        <postalSuburb></postalSuburb>
        <postalPostcode></postalPostcode>
        <postalStateId></postalStateId>
    </publicContactData>
    <technicalContactData>
        <titleId>1</titleId>
        <firstName>FirstName</firstName>
        <lastName>LastName</lastName>
        <position></position>
        <phone>(02) 1234 5678</phone>
        <mobile>1234 456 890</mobile>
        <fax>(02) 1234 5678</fax>
        <email>my_email@email.com</email>
        <postalStreetNumber>1</postalStreetNumber>
        <postalStreetName>My Street</postalStreetName>
        <postalSuburb>My Town</postalSuburb>
        <postalPostcode>2601</postalPostcode>
        <postalStateId>ACT</postalStateId>
    </technicalContactData>
    <thresholdData>
        <energyFuelUsageData>
            <energyFuelId>1</energyFuelId>
            <otherEnergyFuelName></otherEnergyFuelName>
            <quantityForPeriod>2</quantityForPeriod>
            <unitId>2</unitId>
        </energyFuelUsageData>
        <oneTonnePerHourFlag>False</oneTonnePerHourFlag>
        <annualUsage>123</annualUsage>
        <twentyMWFlag>False</twentyMWFlag>
    </thresholdData>
    <emissionData>
        <substanceId>1</substanceId>
        <usageAmountInKg>4560</usageAmountInKg>
        <airPointEmissionInKg>100</airPointEmissionInKg>
        <airPointEet1>False</airPointEet1>
        <airPointEet2>False</airPointEet2>
        <airPointEet3>False</airPointEet3>
        <airPointEet4>True</airPointEet4>
        <airPointEet5>False</airPointEet5>
        <airFugitiveEmissionInKg>100</airFugitiveEmissionInKg>
        <airFugitiveEet1>False</airFugitiveEet1>
        <airFugitiveEet2>True</airFugitiveEet2>
        <airFugitiveEet3>True</airFugitiveEet3>
        <airFugitiveEet4>False</airFugitiveEet4>
        <airFugitiveEet5>True</airFugitiveEet5>
        <airTotalEmissionInKg>200</airTotalEmissionInKg>
        <waterEmissionInKg>222</waterEmissionInKg>
        <waterEet1>False</waterEet1>
        <waterEet2>True</waterEet2>
        <waterEet3>False</waterEet3>
        <waterEet4>False</waterEet4>
        <waterEet5>False</waterEet5>
        <landEmissionInKg>444</landEmissionInKg>
        <landEet1>False</landEet1>
        <landEet2>True</landEet2>
        <landEet3>False</landEet3>
    </emissionData>

```

Example 4 – reportUpload() Request contd...

```

        <landEet4>False</landEet4>
        <landEet5>False</landEet5>
    </emissionData>
    <reductionActivityData>
        <reductionActivityId>21</reductionActivityId>
        <comment>This reduces emissions by a 80%</comment>
    </reductionActivityData>
    <reductionInstallationData>
        <reductionInstallationId>6</reductionInstallationId>
        <yearInstalled>1980</yearInstalled>
        <comment>Gets services regularly</comment>
    </reductionInstallationData>
    <transferData>
        <substanceId>1</substanceId>
        <transferAmountInKg>100</transferAmountInKg>
        <tet1>False</tet1>
        <tet2>False</tet2>
        <tet3>False</tet3>
        <tet4>True</tet4>
        <tet5>False</tet5>
        <onsiteOrOffsite>Offsite</onsiteOrOffsite>
        <desitinationId>1</desitinationId>
    </transferData>
    <resourceData>
        <internalCost>123</internalCost>
        <externalCost>321</externalCost>
        <publicStatement>Working with community on what we
        do</publicStatement>
        <confidentialStatement></confidentialStatement>
    </resourceData>
</reportDetails>
<overwrite>True</overwrite>
</tns:uploadReport>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Example 4 – reportUpload() Request contd...

The following response is returned on success upload of a report.

```

<?xml version="1.0" encoding="utf-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ns1="http://services.report.ws.lucinda.apl.env.gov.au/">
    <soapenv:Body>
        <ns1:uploadReportResponse>
            <boolean>true</boolean>
        </ns1:uploadReportResponse>
    </soapenv:Body>
</soapenv:Envelope>

```

Example 4 – reportUpload() Response

4 Exceptions

If the NPI Report Upload Web Service encounters any exceptions, it returns a standard SOAP fault message. In addition to the standard faultcode/faultstring, this message also contains a ReportUploadException element which provides an identifier and specific message. Figure 8 shows the structure of this element.

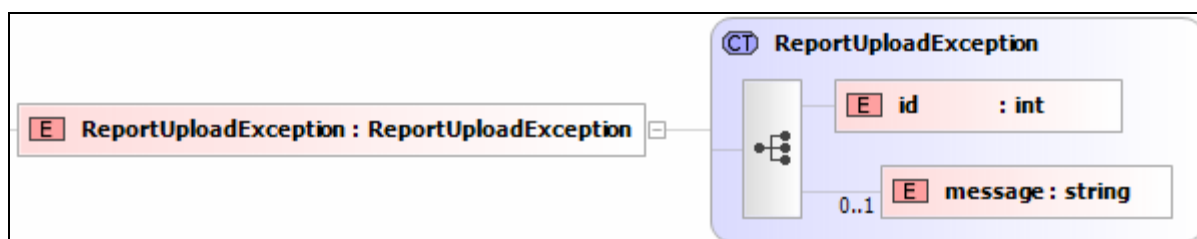


Figure 8 – Report Upload Exception

The table below describes the exceptions that are returned by the service. If you require further assistance in determining the cause of an exception, please contact your jurisdiction.

Table 6 – Exception List

Exception Name	ID	Description
User Not Authenticated	300000001	The cause of this exception is usually triggered by an invalid user name/password combination.
User Not Authorised	300000002	Occurs when trying to upload against a report that is not UNCOMMENCED or in the DRAFT state.
Unknown Report	300000005	A report with the provided identifier cannot be found.
Overwrite Required	300000006	A report for the given facility/year already exists and the overwrite flag was set to false.
Invalid Emission Data	300000007	If you encounter this error, please contact your jurisdiction.
No Facilities	300000008	The user is not associated to any facilities as a Coordinator.
Delegator Error	300000009	If you encounter this error, please contact your jurisdiction.
Datalayer Error	300000010	An unexpected error has occurred and the error message will provide specific details. This is likely to occur when bad/non-existent reference data is provided.
Report Validation Error	300000011	This exception occurs when the report contains invalid data. The content of the error message will identify the area of concern (e.g. Facility, Threshold/Fuel, Contacts Emissions, Transfers, Reduction Activities, Pollution Control Devices etc)
Initial Context	300000012	If you encounter this error, please contact your jurisdiction.
Unknown Substance	300000013	An identifier that does not correspond to a current substance has been provided.
Invalid Emission Amount	300000014	An invalid value has been provided for an emission. The error message will provide specific details on the particular field.
Initial State Not Found	300000015	If you encounter this error, please contact your jurisdiction.
Invalid Date Format	300000016	An invalid date has been provided. The error message will provide specific details on the particular field.

Exception Name	ID	Description
Invalid Number Format	300000017	An invalid number has been provided. The error message will provide specific details on the particular field.
Invalid Boolean	300000018	An invalid value has provided where the system expects a true/false value only. The error message will provide specific details on the particular field.
Invalid Address	300000019	An invalid address has been provided. The error message will provide specific details on which address is incorrect (e.g. Registered/Postal etc)
Empty Object	300000020	An unexpected error has occurred and the error message will provide specific details.

Below is an example of a response when an exception has occurred. This particular response was generated by sending an invalid password for a user's account.

```
<?xml version="1.0" encoding="utf-8"?>
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ns1="http://services.report.ws.lucinda.apl.env.gov.au/">
  <soapenv:Body>
    <soapenv:Fault
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
      <faultcode>soapenv:Server</faultcode>
      <faultstring>Authentication failed for user: UserName.
        Possible reasons:
        Invalid Username / Password.
        The user doesn't have the Coordinator role.
        The account has been locked after three incorrect login
        attempts.
      </faultstring>
      <detail>
        <ns1:ReportUploadException>
          <id>3000000001</id>
          <message>Authentication failed for user: UserName
            Possible reasons:
            Invalid Username / Password.
            The user doesn't have the Coordinator role.
            The account has been locked after three incorrect login
            attempts.
          </message>
        </ns1:ReportUploadException>
      </detail>
    </soapenv:Fault>
  </soapenv:Body>
</soapenv:Envelope>
```

Example 5 – Invalid user name / password combination Exception

Appendix A – Report Upload WSDL

The most up to date version of this WSDL can be found at https://npi.environment.gov.au/web/service/upload_service?wsdl

```
<?xml version="1.0" encoding="UTF-8"?><definitions xmlns:tns="http://services.report.ws.lucinda.apl.env.gov.au/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://services.report.ws.lucinda.apl.env.gov.au/"
name="ReportUploadService">
  <types>
    <xsd:schema>
      <xsd:import namespace="http://services.report.ws.lucinda.apl.env.gov.au/"
schemaLocation="https://npi.environment.gov.au/web/service/upload_service?xsd=1"></xsd:import>
    </xsd:schema>
  </types>
  <message name="getFacilities">
    <part element="tns:getFacilities" name="parameters"></part>
  </message>

  <message name="getFacilitiesResponse">
    <part element="tns:getFacilitiesResponse" name="parameters"></part>
  </message>
  <message name="ReportUploadException">
    <part element="tns:ReportUploadException" name="fault"></part>
  </message>
  <message name="getReferenceData">
    <part element="tns:getReferenceData" name="parameters"></part>
  </message>

  <message name="getReferenceDataResponse">
    <part element="tns:getReferenceDataResponse" name="parameters"></part>
  </message>
  <message name="uploadReport">
    <part element="tns:uploadReport" name="parameters"></part>
  </message>
  <message name="uploadReportResponse">
    <part element="tns:uploadReportResponse" name="parameters"></part>
  </message>
```

```

<portType name="ReportUpload">
  <operation name="getFacilities">
    <input message="tns:getFacilities"></input>
    <output message="tns:getFacilitiesResponse"></output>
    <fault message="tns:ReportUploadException" name="ReportUploadException"></fault>
  </operation>
  <operation name="getReferenceData">
    <input message="tns:getReferenceData"></input>
    <output message="tns:getReferenceDataResponse"></output>

    <fault message="tns:ReportUploadException" name="ReportUploadException"></fault>
  </operation>
  <operation name="uploadReport">
    <input message="tns:uploadReport"></input>
    <output message="tns:uploadReportResponse"></output>
    <fault message="tns:ReportUploadException" name="ReportUploadException"></fault>
  </operation>
</portType>
<binding type="tns:ReportUpload" name="ReportUploadPortBinding">

  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"></soap:binding>
  <operation name="getFacilities">
    <soap:operation soapAction=""></soap:operation>
    <input>
      <soap:body use="literal"></soap:body>
    </input>
    <output>
      <soap:body use="literal"></soap:body>
    </output>

    <fault name="ReportUploadException">
      <soap:fault use="literal" name="ReportUploadException"></soap:fault>
    </fault>
  </operation>
  <operation name="getReferenceData">
    <soap:operation soapAction=""></soap:operation>
    <input>

```

```

        <soap:body use="literal"></soap:body>
    </input>

    <output>
        <soap:body use="literal"></soap:body>
    </output>
    <fault name="ReportUploadException">
        <soap:fault use="literal" name="ReportUploadException"></soap:fault>
    </fault>
</operation>
<operation name="uploadReport">
    <soap:operation soapAction=""></soap:operation>

    <input>
        <soap:body use="literal"></soap:body>
    </input>
    <output>
        <soap:body use="literal"></soap:body>
    </output>
    <fault name="ReportUploadException">
        <soap:fault use="literal" name="ReportUploadException"></soap:fault>
    </fault>

</operation>
</binding>
<service name="ReportUploadService">
    <port binding="tns:ReportUploadPortBinding" name="ReportUploadPort">
        <soap:address location="https://npi.environment.gov.au/webservice/upload_service"></soap:address>
    </port>
</service>
</definitions>

```

Appendix B – Report Upload Schema

The most up to date version of this XSD can be found at https://npi.environment.gov.au/webservice/upload_service?xsd=1

```
<?xml version="1.0" encoding="utf-8" ?>
<xs:schema targetNamespace="http://services.report.ws.lucinda.apl.env.gov.au/" version="1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element xmlns:ns1="http://services.report.ws.lucinda.apl.env.gov.au/" name="uploadReport"
type="ns1:uploadReport" />
  <xs:complexType name="uploadReport">
    <xs:sequence>
      <xs:element xmlns:ns2="http://services.report.ws.lucinda.apl.env.gov.au/" name="authData"
type="ns2:userAuthenticationData" />
      <xs:element name="facilityId" type="xs:string" />
      <xs:element name="reportingYearId" type="xs:string" />
      <xs:element xmlns:ns3="http://services.report.ws.lucinda.apl.env.gov.au/" name="facilityDetails"
type="ns3:facilityDetailsData" />
      <xs:element xmlns:ns4="http://services.report.ws.lucinda.apl.env.gov.au/" name="reportDetails"
type="ns4:reportDetailsData" />
      <xs:element minOccurs="0" name="overwrite" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="userAuthenticationData">
    <xs:sequence>
      <xs:element name="username" type="xs:string" />
      <xs:element name="password" type="xs:string" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="facilityDetailsData">
    <xs:sequence>
      <xs:element minOccurs="0" name="facilityName" type="xs:string" />
      <xs:element minOccurs="0" name="epaLicenceNumber" type="xs:string" />
      <xs:element minOccurs="0" name="numOfEmployees" type="xs:string" />
      <xs:element minOccurs="0" name="websiteAddress" type="xs:string" />
      <xs:element minOccurs="0" name="mainActivity" type="xs:string" />
      <xs:element minOccurs="0" name="postalStreetNumber" type="xs:string" />
      <xs:element minOccurs="0" name="postalStreet" type="xs:string" />
      <xs:element minOccurs="0" name="postalSuburb" type="xs:string" />
    </xs:sequence>
  </xs:complexType>

```

```

        <xs:element minOccurs="0" name="postalPostcode" type="xs:string" />
        <xs:element minOccurs="0" name="postalStateId" type="xs:string" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="reportDetailsData">
    <xs:sequence>
        <xs:element minOccurs="0" name="dataPeriodStart" type="xs:string" />
        <xs:element minOccurs="0" name="dataPeriodEnd" type="xs:string" />
        <xs:element xmlns:ns5="http://services.report.ws.lucinda.apl.env.gov.au/" name="publicContactData"
type="ns5:publicContactData" />
        <xs:element xmlns:ns6="http://services.report.ws.lucinda.apl.env.gov.au/" name="technicalContactData"
type="ns6:technicalContactData" />
        <xs:element xmlns:ns7="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
name="thresholdData" type="ns7:thresholdData" />
        <xs:element xmlns:ns8="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="emissionData" type="ns8:emissionData" />
        <xs:element xmlns:ns9="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="reductionActivityData" type="ns9:reductionActivityData" />
        <xs:element xmlns:ns10="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="reductionInstallationData" type="ns10:reductionInstallationData" />
        <xs:element xmlns:ns11="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="transferData" type="ns11:transferData" />
        <xs:element xmlns:ns12="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
name="resourceData" type="ns12:resourceData" />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="publicContactData">
    <xs:sequence>
        <xs:element minOccurs="0" name="titleId" type="xs:string" />
        <xs:element minOccurs="0" name="firstName" type="xs:string" />
        <xs:element minOccurs="0" name="lastName" type="xs:string" />
        <xs:element name="position" type="xs:string" />
        <xs:element name="phone" type="xs:string" />
        <xs:element minOccurs="0" name="mobile" type="xs:string" />
        <xs:element minOccurs="0" name="fax" type="xs:string" />
        <xs:element minOccurs="0" name="email" type="xs:string" />
        <xs:element minOccurs="0" name="postalStreetNumber" type="xs:string" />
        <xs:element minOccurs="0" name="postalStreetName" type="xs:string" />
    </xs:sequence>
</xs:complexType>

```

```

    <xs:element minOccurs="0" name="postalSuburb" type="xs:string" />
    <xs:element minOccurs="0" name="postalPostcode" type="xs:string" />
    <xs:element minOccurs="0" name="postalStateId" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="technicalContactData">
  <xs:sequence>
    <xs:element name="titleId" type="xs:string" />
    <xs:element name="firstName" type="xs:string" />
    <xs:element name="lastName" type="xs:string" />
    <xs:element minOccurs="0" name="position" type="xs:string" />
    <xs:element name="phone" type="xs:string" />
    <xs:element minOccurs="0" name="mobile" type="xs:string" />
    <xs:element minOccurs="0" name="fax" type="xs:string" />
    <xs:element minOccurs="0" name="email" type="xs:string" />
    <xs:element minOccurs="0" name="postalStreetNumber" type="xs:string" />
    <xs:element minOccurs="0" name="postalStreetName" type="xs:string" />
    <xs:element minOccurs="0" name="postalSuburb" type="xs:string" />
    <xs:element minOccurs="0" name="postalPostcode" type="xs:string" />
    <xs:element minOccurs="0" name="postalStateId" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="thresholdData">
  <xs:sequence>
    <xs:element xmlns:ns13="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="energyFuelUsageData" type="ns13:energyFuelUsageData" />
    <xs:element name="oneTonnePerHourFlag" type="xs:string" />
    <xs:element name="annualUsage" type="xs:string" />
    <xs:element name="twentyMWFlag" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="energyFuelUsageData">
  <xs:sequence>
    <xs:element name="energyFuelId" type="xs:string" />
    <xs:element minOccurs="0" name="otherEnergyFuelName" type="xs:string" />
    <xs:element name="quantityForPeriod" type="xs:string" />
    <xs:element name="unitId" type="xs:string" />
  </xs:sequence>

```

```

</xs:complexType>
<xs:complexType name="emissionData">
  <xs:sequence>
    <xs:element name="substanceId" type="xs:string" />
    <xs:element minOccurs="0" name="usageAmountInKg" type="xs:string" />
    <xs:element minOccurs="0" name="airPointEmissionInKg" type="xs:string" />
    <xs:element minOccurs="0" name="airPointEet1" type="xs:string" />
    <xs:element minOccurs="0" name="airPointEet2" type="xs:string" />
    <xs:element minOccurs="0" name="airPointEet3" type="xs:string" />
    <xs:element minOccurs="0" name="airPointEet4" type="xs:string" />
    <xs:element minOccurs="0" name="airPointEet5" type="xs:string" />
    <xs:element minOccurs="0" name="airFugitiveEmissionInKg" type="xs:string" />
    <xs:element minOccurs="0" name="airFugitiveEet1" type="xs:string" />
    <xs:element minOccurs="0" name="airFugitiveEet2" type="xs:string" />
    <xs:element minOccurs="0" name="airFugitiveEet3" type="xs:string" />
    <xs:element minOccurs="0" name="airFugitiveEet4" type="xs:string" />
    <xs:element minOccurs="0" name="airFugitiveEet5" type="xs:string" />
    <xs:element minOccurs="0" name="airTotalEmissionInKg" type="xs:string" />
    <xs:element minOccurs="0" name="waterEmissionInKg" type="xs:string" />
    <xs:element minOccurs="0" name="waterEet1" type="xs:string" />
    <xs:element minOccurs="0" name="waterEet2" type="xs:string" />
    <xs:element minOccurs="0" name="waterEet3" type="xs:string" />
    <xs:element minOccurs="0" name="waterEet4" type="xs:string" />
    <xs:element minOccurs="0" name="waterEet5" type="xs:string" />
    <xs:element minOccurs="0" name="landEmissionInKg" type="xs:string" />
    <xs:element minOccurs="0" name="landEet1" type="xs:string" />
    <xs:element minOccurs="0" name="landEet2" type="xs:string" />
    <xs:element minOccurs="0" name="landEet3" type="xs:string" />
    <xs:element minOccurs="0" name="landEet4" type="xs:string" />
    <xs:element minOccurs="0" name="landEet5" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="reductionActivityData">
  <xs:sequence>
    <xs:element name="reductionActivityId" type="xs:string" />
    <xs:element minOccurs="0" name="comment" type="xs:string" />
  </xs:sequence>
</xs:complexType>

```

```

<xs:complexType name="reductionInstallationData">
  <xs:sequence>
    <xs:element name="reductionInstallationId" type="xs:string" />
    <xs:element name="yearInstalled" type="xs:string" />
    <xs:element minOccurs="0" name="comment" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="transferData">
  <xs:sequence>
    <xs:element name="substanceId" type="xs:string" />
    <xs:element name="transferAmountInKg" type="xs:string" />
    <xs:element minOccurs="0" name="tet1" type="xs:string" />
    <xs:element minOccurs="0" name="tet2" type="xs:string" />
    <xs:element minOccurs="0" name="tet3" type="xs:string" />
    <xs:element minOccurs="0" name="tet4" type="xs:string" />
    <xs:element minOccurs="0" name="tet5" type="xs:string" />
    <xs:element name="onsiteOrOffsite" type="xs:string" />
    <xs:element name="desitinationId" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="resourceData">
  <xs:sequence>
    <xs:element minOccurs="0" name="internalCost" type="xs:string" />
    <xs:element minOccurs="0" name="externalCost" type="xs:string" />
    <xs:element minOccurs="0" name="publicStatement" type="xs:string" />
    <xs:element minOccurs="0" name="confidentialStatement" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:element xmlns:ns14="http://services.report.ws.lucinda.apl.env.gov.au/" name="uploadReportResponse"
type="ns14:uploadReportResponse" />
<xs:complexType name="uploadReportResponse">
  <xs:sequence>
    <xs:element name="boolean" type="xs:boolean" />
  </xs:sequence>
</xs:complexType>
<xs:element xmlns:ns15="http://services.report.ws.lucinda.apl.env.gov.au/" name="ReportUploadException"
type="ns15:ReportUploadException" />
<xs:complexType name="ReportUploadException">

```

```

<xs:sequence>
  <xs:element name="id" type="xs:int" />
  <xs:element minOccurs="0" name="message" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:element xmlns:ns16="http://services.report.ws.lucinda.apl.env.gov.au/" name="getFacilities"
type="ns16:getFacilities" />
<xs:complexType name="getFacilities">
  <xs:sequence>
    <xs:element xmlns:ns17="http://services.report.ws.lucinda.apl.env.gov.au/" name="authData"
type="ns17:userAuthenticationData" />
  </xs:sequence>
</xs:complexType>
<xs:element xmlns:ns18="http://services.report.ws.lucinda.apl.env.gov.au/" name="getFacilitiesResponse"
type="ns18:getFacilitiesResponse" />
<xs:complexType name="getFacilitiesResponse">
  <xs:sequence>
    <xs:element xmlns:ns19="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="facilityData" type="ns19:facilityData" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="facilityData">
  <xs:sequence>
    <xs:element name="facilityId" type="xs:string" />
    <xs:element name="jurisdictionFacilityId" type="xs:string" />
    <xs:element name="facilityName" type="xs:string" />
    <xs:element name="siteAddress" type="xs:string" />
    <xs:element name="state" type="xs:string" />
    <xs:element name="epaLicenceNumber" type="xs:string" />
    <xs:element xmlns:ns20="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="reportData" type="ns20:reportData" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="reportData">
  <xs:sequence>
    <xs:element name="reportingYearId" type="xs:string" />
    <xs:element name="reportingYearName" type="xs:string" />
  </xs:sequence>

```

```

</xs:complexType>
<xs:element xmlns:ns21="http://services.report.ws.lucinda.apl.env.gov.au/" name="getReferenceData"
type="ns21:getReferenceData" />
<xs:complexType name="getReferenceData">
  <xs:sequence>
    <xs:element xmlns:ns22="http://services.report.ws.lucinda.apl.env.gov.au/" name="authData"
type="ns22:userAuthenticationData" />
  </xs:sequence>
</xs:complexType>
<xs:element xmlns:ns23="http://services.report.ws.lucinda.apl.env.gov.au/" name="getReferenceDataResponse"
type="ns23:getReferenceDataResponse" />
<xs:complexType name="getReferenceDataResponse">
  <xs:sequence>
    <xs:element xmlns:ns24="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
name="referenceData" type="ns24:referenceData" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="referenceData">
  <xs:sequence>
    <xs:element xmlns:ns25="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="australianState" type="ns25:australianStateRef" />
    <xs:element xmlns:ns26="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="energyFuel" type="ns26:energyFuelRef" />
    <xs:element xmlns:ns27="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="reductionActivity" type="ns27:reductionActivityRef" />
    <xs:element xmlns:ns28="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="reductionInstallation" type="ns28:reductionInstallationRef" />
    <xs:element xmlns:ns29="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="reportingYear" type="ns29:reportingYearRef" />
    <xs:element xmlns:ns30="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="substance" type="ns30:substanceRef" />
    <xs:element xmlns:ns31="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="title" type="ns31:titleRef" />
    <xs:element xmlns:ns32="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="transferDestinationOrContainment" type="ns32:transferDestinationOrContainmentRef" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="australianStateRef">

```

```

<xs:sequence>
  <xs:element name="id" type="xs:string" />
  <xs:element name="name" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="energyFuelRef">
  <xs:sequence>
    <xs:element name="id" type="xs:string" />
    <xs:element name="name" type="xs:string" />
    <xs:element xmlns:ns33="http://services.report.ws.lucinda.apl.env.gov.au/" minOccurs="0"
maxOccurs="unbounded" name="allowedUnit" type="ns33:unitRef" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="unitRef">
  <xs:sequence>
    <xs:element name="id" type="xs:string" />
    <xs:element name="name" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="reductionActivityRef">
  <xs:sequence>
    <xs:element name="id" type="xs:string" />
    <xs:element name="description" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="reductionInstallationRef">
  <xs:sequence>
    <xs:element name="id" type="xs:string" />
    <xs:element name="description" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="reportingYearRef">
  <xs:sequence>
    <xs:element name="id" type="xs:string" />
    <xs:element name="name" type="xs:string" />
  </xs:sequence>
</xs:complexType>

```

```
<xs:complexType name="substanceRef">
  <xs:sequence>
    <xs:element name="id" type="xs:string" />
    <xs:element name="name" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="titleRef">
  <xs:sequence>
    <xs:element name="id" type="xs:string" />
    <xs:element name="name" type="xs:string" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="transferDestinationOrContainmentRef">
  <xs:sequence>
    <xs:element name="id" type="xs:string" />
    <xs:element name="name" type="xs:string" />
  </xs:sequence>
</xs:complexType>
</xs:schema>
```