



## Intergovernmental Committee on Surveying and Mapping



# ePlan Protocol Schema Architecture

Document Status: Released

Version: 2.1

Version Date: 19/11/2010

## Copyright

© The Intergovernmental Committee for Surveying and Mapping, Australia, 2010

## Amendment History

Version	Date	Author	Comments
1.0	10 July 2009	Nevil Cumerford	Initial implementation
2.0	15 September, 2010	Vivian Farrell	Update for ICSM ePlan
2.0.1		Vivian Farrell Robert Green Nevil Cumerford	Update for release as ICSM ePlan Schema Architecture
2.1	19 November	Vivian Farrell	Further touchups. Completed appendices.

## Table of Contents

<b>1</b>	<b>Introduction</b> .....	<b>1</b>
1.1	Background.....	1
1.2	Purpose.....	1
1.3	Scope.....	1
1.4	References.....	2
1.5	Definitions.....	2
1.6	Audience.....	3
<b>2</b>	<b>CIF Schema Architecture</b> .....	<b>4</b>
2.1	Overview.....	4
2.1.1	Schema and data file relationship.....	4
2.2	ICSM ePlan CIF Schema.....	4
2.3	Jurisdictional CIF Schemas.....	5
2.3.1	Jurisdictional CIF Schema Components.....	5
<b>3</b>	<b>Configuration and Version Control</b> .....	<b>8</b>
3.1	Reference Data File Names.....	8
3.2	Reference Data Context File.....	8
3.2.1	Structure.....	8
3.2.2	RDC Versions.....	8
3.2.3	Specifying in the CIF.....	9
3.2.4	Overlapping version dates.....	9
3.3	Administrative Areas.....	9
3.4	File Accessibility and Storage.....	9

## Table of Appendices

Appendix A	Annotations Schema.....	A—1
Appendix B	Surveyor Certificates Schema.....	B—19
Appendix C	Administrative Area Schema.....	C—1
Appendix D	Enumerated Types Schema.....	D—1
Appendix E	Reference Data Context Schema.....	E—1
Appendix F	Jurisdictional CIF Schema Implementation Guidelines.....	F—1
Appendix G	File Naming.....	G—2

## Table of Figures

Figure 1: LandXML to ICSM ePlan schema.....	4
Figure 2: ePlan Schema Architecture showing the relationship between national and jurisdictional components. ....	6

# 1 Introduction

The ePlan Model is a single conceptual model for cadastral survey data in Australia and New Zealand. LandXML has been chosen as the schema to implement the ePlan Model.

An ePlan Protocol has been developed that maps the components of the ePlan model to LandXML elements and attributes. This protocol has been implemented as an XML schema that is a subset of LandXML.

The ePlan schema is extensible so that it can be adapted for specific jurisdictional requirements while still conforming to the LandXML standard. This document describes how the ePlan schema is extended for use in different jurisdictions to create ePlan Cadastral Information Files (CIFs).

A CIF can be defined as:

“A LandXML Instance that complies with a Jurisdictional ePlan Protocol using the values defined in the Reference Data Set”

This document sets out the schema architecture developed to support the creation of ePlan CIFs that meet jurisdictional requirements and are valid LandXML files.

## 1.1 Background

The Intergovernmental Committee for Surveying and Mapping (ICSM) ePlan Model is a general model for cadastral surveying data in Australian and New Zealand. The model was developed by a working committee with members from each jurisdiction represented by the ICSM.

The working group also produced the ePlan Protocol that maps data from the Model to LandXML. The protocol allows variations for some data elements that may be required in one jurisdiction but not another. It also allows for variations such as lists of data values that differ between jurisdictions (e.g. types of monuments placed at cadastral boundary corners) and differing templates for textural components of the model, such as annotations and certificates.

XML schemas were chosen as the method to implement these differing requirements while maintaining conformance with LandXML.

## 1.2 Purpose

This document describes the components of the ePlan Schema Architecture, their relationships and how they are used to create an ePlan file (a Cadastral Information File or CIF).

## 1.3 Scope

This document describes:

1. The various XML documents that comprise the ePlan Protocol

2. The relationships between the documents
3. Suggested file structures for storage
4. File maintenance

## 1.4 References

1. ICSM, *ePlan Protocol – Data Model*, version 1.0, 10 September, 2009  
[http://www.icsm.gov.au/icsm/ePlan/Schema-1.2/ePlan\\_Model.pdf](http://www.icsm.gov.au/icsm/ePlan/Schema-1.2/ePlan_Model.pdf)
2. ICSM, *ePlan Protocol – LandXML Structural Requirements*, version 1.0, 15 November, 2010  
<http://icsm-eplan.govspace.gov.au/eplan-protocol/>
3. Recommended XML Namespaces Schema for Australian Government Organisations (Australian Government Information Management Office) <http://icsm-eplan.govspace.gov.au/eplan-protocol/>
4. ICSM, *ePlan Protocol – LandXML Mapping Requirements*, version 2.1, 16 November, 2010  
<http://icsm-eplan.govspace.gov.au/eplan-protocol/>

## 1.5 Definitions

### **CIF**

Cadastral Information File, a LandXML file used to store the data specified by the ePlan Model in a structure defined by the ePlan Protocol.

### **Enumeration**

A set of values, one of which must be used as the value of certain data items.

### **ePlan**

A model for cadastral survey information defined by the ICSM.

### **ICSM**

The Intergovernmental Committee on Surveying and Mapping.

### **LandXML**

An XML schema used as the basis of the ePlan schema.

### **XML**

Extensible mark–up language.

### **XSD**

Extensible mark–up language schema, an XML document that describes the structure and requirements of another XML document.

### **Reference Data**

The collection of XML files containing the set of valid values that must be used for specified fields.

## **1.6 Audience**

The primary audience of this document is:

- Software developers creating applications for ICSM member organisations that are using the ePlan Protocol
- Software developers creating products for the surveying industry to create ePlan files
- Technical staff within ICSM member organisations that manage and distribute data using the ePlan Protocol

## 2 CIF Schema Architecture

### 2.1 Overview

The ePlan Schema Architecture consists of a number of XML schema and XML data files that together define the structure and types of data that are in an ePlan Cadastral Information File (CIF). These files are published at both the national and jurisdictional level.

LandXML is used as the base schema on which the ePlan CIF schemas are built. LandXML is an international standard that is widely adopted in spatial software packages. Every ePlan CIF is a LandXML compliant XML file.

To accommodate the differences between jurisdictions that use ePlan, a single ICSM ePlan CIF Schema has been developed that meets the requirements of all ICSM member jurisdictions. This schema is essentially a subset of LandXML containing only the sections used for creating ePlan CIFs. This schema is subsequently used as the template for jurisdictional CIF schemas (§ 2.3). By structuring the schemas in this way, a CIF conforming with a jurisdictional schema will also be a valid ICSM ePlan and LandXML file.

The ePlan Schema Architecture also provides for the creation of data value lists known as reference data and enumeration lists. These allow jurisdictions to restrict the format and values of data fields to conform with specific jurisdictional requirements.

#### 2.1.1 Schema and data file relationship

A schema defines the structure and, in some cases, enumerations for a related data file. A data file has a structure defined by a schema and is used only for data. The set of data files used to create a CIF are called reference data files.

As the ePlan schema architecture is XML based, Schema files have an XSD file extension and data files have an XML file extension.

### 2.2 ICSM ePlan CIF Schema

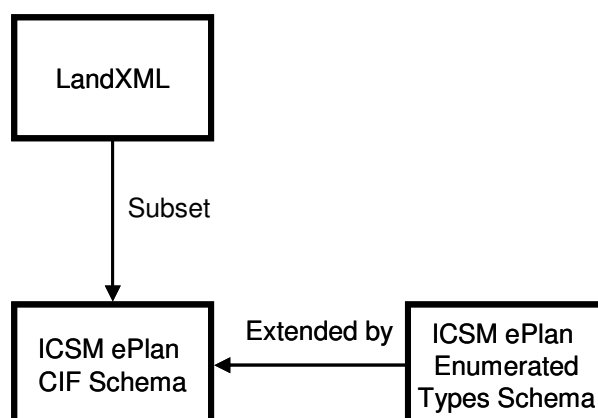


Figure 1: LandXML to ICSM ePlan schema

The ICSM ePlan CIF Schema is a subset of the LandXML schema. It defines only the elements and attributes required for the ePlan Model and also defines some elements and attributes that are optional in LandXML as being required in ePlan.

The detailed requirements for the ePlan Schema are defined in the ePlan Protocol (see references 1 and 2).

## **2.3 Jurisdictional CIF Schemas**

Jurisdictional CIF Schemas are very similar to the ICSM ePlan CIF Schema, the differences are that some optional elements and attributes in the ICSM ePlan CIF Schema may be omitted from a Jurisdictional CIF Schema and others may be made mandatory.

For example, the attribute for //Survey/SurveyHeader@desc is:

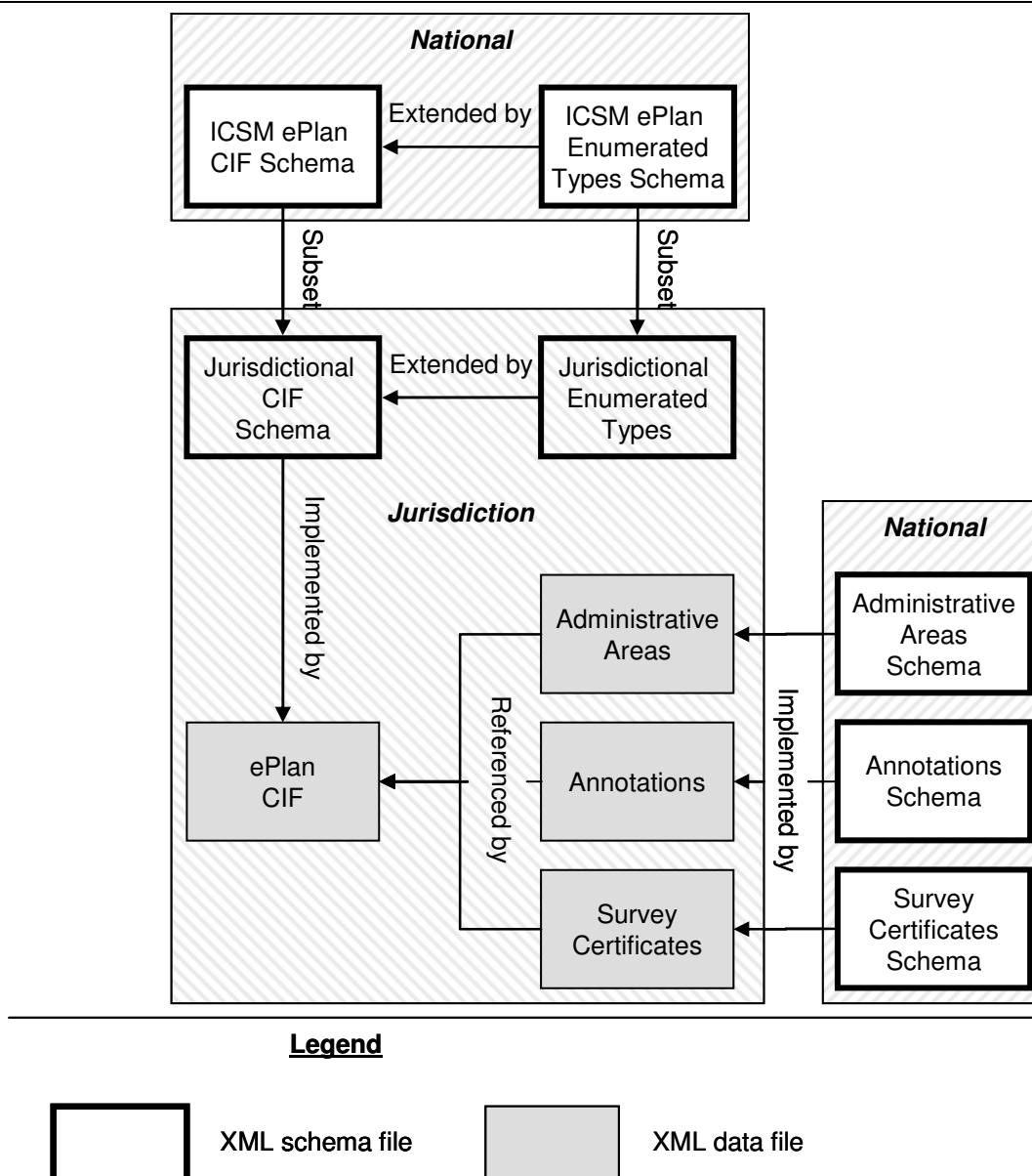
- optional in the ICSM ePlan CIF schema
- omitted from the Victorian schema, and
- mandatory in the Queensland schema

An element or attribute that is mandatory in the ICSM ePlan CIF schema can't be made optional or omitted from a Jurisdictional CIF Schema, it must remain mandatory.

### **2.3.1 Jurisdictional CIF Schema Components**

Jurisdictional CIF Schemas have a number of components to accommodate the differences between jurisdictions and for easier updating. For example, Administrative Areas change frequently so new administration area files are published regularly, whereas survey certificates rarely change and so don't need to be updated as often.





**Figure 2: ePlan Schema Architecture showing the relationship between national and jurisdictional components.**

The following definitions should be used with the above diagram:

- **Subset** – The schema is based on a parent schema. A data file created with the sub schema will pass XML validation against the super schema.
- **Extended** – The schema has extended functionality provided by a secondary schema. The enumerated types schema provides extended enumerations to the CIF schema.
- **Implemented** – Where an XML data file uses a schema to define its structure.
- **Referenced** – The ePlan CIF references data in 3 files to obtain mandated data values from jurisdictions.

### 2.3.1.1 Schema component descriptions

Each jurisdiction is responsible for creating and maintaining the files shown outside the "national" boxes. These files are created according to the jurisdiction's specific requirements.

The following table summarises the components in Figure 2.

Item	Description
ICSM ePlan CIF Schema	A subset of LandXML that meets the requirements of all ICSM member jurisdictions.
ICSM ePlan Enumerated Types Schema	Defines the enumerated types in the ICSM ePlan CIF Schema. This schema also defines all the enumerated types used in other schemas and data files.
Jurisdictional CIF Schema	A jurisdictionally specific version of the ICSM ePlan CIF Schema
Jurisdictional Enumerated types Schema	A jurisdictionally specific version of the ICSM ePlan Enumerated Types Schema populated with enumeration values.
Administrative Areas Schema	Defines the structure of the related administration area data file.
Annotations Schema	Defines the structure of the related annotations data file.
Survey Certificates Schema	Defines the structure of the related certificates data file.
Administrative Area data file	Stores the names and relationships of the various types of administration area. Note that the types of administration area are defined in the enumerations schema. This file is used to provide valid values for the //Survey/SurveyHeader/AdminArea and //Parcels/Parcel/LocationAddress/AdminArea
Annotations data file	Stores the various annotations used by surveyors in the jurisdiction. This file gives valid values to the //Survey/PlanHeader/Annotations Element.
Survey Certificates data file	Stores the various certificates used by surveyors in the jurisdiction. This file gives valid values to the //Survey/PlanHeader/SurveyCertificate Element.

### 2.3.1.2 CIF Construction

The jurisdictional schemas and reference data files provide the complete definition of the CIF for the jurisdiction. CIFs are validated to ensure they adhere to these files. However, all externally submitted CIFs **must use LandXML 1.2 as the base schema**. Third-parties may use any other schemas and reference data files to assist in the construction of a CIF. There are no restrictions on how the files are used externally.

The specific set of files to be used when creating a CIF change from time to time (e.g. local government area names in the Administrative Area data file change frequently), the mechanism for determining which files to use is described in § 3.

## 3 Configuration and Version Control

Reference data files change from time to time as the data within them is updated. The list of files that are current for a particular date is a reference data context (RDC). Each RDC has a date range for which it is valid.

Administrative areas change frequently, sometimes daily, therefore the administrative areas reference data file is updated independently of the RDC (see § 3.3).

### 3.1 Reference Data File Names

File naming conventions are described in Appendix G. Note that reference data files are specified by full URI in the Reference Data Context, e.g.

```
http://www.lpma.nsw.gov.au/__data/assets/xml_file/0010/137368/xml-gov-au-nsw-icsm-cif-eplan-referencedata-101013.xml
```

### 3.2 Reference Data Context File

An RDC file is published by each jurisdiction to specify all current and previous RDCs for the jurisdiction.

#### 3.2.1 Structure

The structure of the RDC file is defined by the RDC Schema (see Appendix E). An RDC file contains one or more `ReferenceDataContext` elements. Each element specifies an RDC, which is a set of reference data files that are to be used for CIFs created within the date range in the `ApplicableFrom` and `ApplicableTo` elements.

When a new RDC is required, it is added as a new `ReferenceDataContext` element in the RDC file and the updated RDC file is published.

#### 3.2.2 RDC Versions

RDC files are backward compatible, i.e. the latest version can be used for any previous date. Therefore, only the latest version of the RDC file is needed to determine the correct set of reference data files for any relevant date.

Version control for the RDC file is provided by the `LastUpdated` element, which specifies the date that the file was last updated. If the RDC file is stored locally, then frequent checks should be made with the jurisdiction's published version to ensure that the latest version is used.

### 3.2.3 Specifying in the CIF

To specify the RDC version (and hence the versions of reference data files) used to create a CIF, the `ReferenceDataContext/ContextId` value from the RDC file is stored in the CIF `LandXML/FeatureDictionary@version` attribute, e.g.

```
<FeatureDictionary name="ReferenceDataContext" version="NRWQLD-RD-3"/>
```

Jurisdictions may use this attribute to check that a valid RDC has been used to create a CIF for the applicable date. It will also be used where RDC applicable dates overlap to determine which RDC was chosen when creating a CIF.

### 3.2.4 Overlapping version dates

RDCs may have overlapping `ApplicableFrom` and `ApplicableTo` dates. This indicates that either RDC may be used for the overlapping period.

Overlapping dates may be used for transition periods, e.g. when a new set of regulations is introduced, projects started under the previous regulations can be completed but new projects are covered by the new regulations.

## 3.3 Administrative Areas

The administrative area data file is backward compatible, i.e. the latest file can be used for any previous date.

As with the RDC file, the file name is always the same. The internal version number can be used to check if it is the latest version, which can be downloaded from the jurisdiction's public server.

## 3.4 File Accessibility and Storage

The latest versions of all ePlan schema and data files are available from publicly accessible URLs specified by each jurisdiction. The reference data context always references these versions of reference data files. If the files are to be downloaded for offline usage, the third-party should be aware that the reference data context will continue to point to the remote URLs.

Local storage of reference data files can be in a single directory. All versioned files contain their version number in the file name and will not clash with other versions. The Reference Data Context is an expanding file and can replace any old versions. The same applies for Admin Areas if the backward compatibility features are used.

## Appendix A Annotations Schema

### A.1 Implementation Guidelines

The Annotations schema instance is defined at the jurisdiction level using the annotations schema (xml-gov-au-icsm-eplan-cif-annotations-1.0.xsd). This file provides a list of acceptable annotations for use in the jurisdiction. Each jurisdiction will validate their annotations against the list they supply in this file. The annotation values can be a static text value or a string with text substitution.

An example file name for this file conforming to Appendix G is:

```
xml-gov-au-nsw-icsm-eplan-cif-annotations-1.0-20101225.xml
```

The following is an annotated skeleton template:

```
<ActionStatementList time="" date="" version="" xmlns="urn:xml-gov-
au:icsm:eplan:cif:annotations:1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:xml-gov-au:icsm:eplan:cif:annotations:1.0
U:\ePlan\Specifications\BUSINE~1\National\EPLANP~3\xml-gov-au-icsm-eplan-cif-annotations-
1.0.xsd">

  <ActionStatement displaySequence="1"> <!-- display order -->
    <Id></Id> <!-- surrogate ID -->
    <Name></Name> <!-- the Type of the annotation - annotationType in enumerations schema
-->
    <HeadOfPower></HeadOfPower> <!-- Act or legislation this applies to -->
    <PresentationType></PresentationType> <!-- one of: StaticText, TextWithSubstitution,
FreeFormText -->
    <Form>
      <Id></Id> <!-- surrogate ID -->
      <Name></Name> <!-- same as name above -->
      <Version></Version> <!-- version of this annotation -->
      <FormattedName></FormattedName>
      <FormTemplateText><![CDATA[]]></FormTemplateText> <!-- Text of the annotation
contained in CDATA -->
      <FormField> <!-- if text substitution is used, each tag is defined with a
formfield -->
        <Id></Id> <!-- tag value -->
        <Name></Name> <!-- same as ID -->
        <Description></Description> <!-- desc of the value -->
        <MinCardinality>1</MinCardinality> <!-- the range of values required -->
        <MaxCardinality>1</MaxCardinality>
        <FieldType></FieldType> <!-- the type e.g. string, int, double etc. -->
      </FormField>
    </Form>
  </ActionStatement>
</ActionStatementList>
```

There are 3 main types of annotations, listed below:

- Static text – A fixed text string that is wholly inserted into the CIF by the surveyor. Static text requires no FormFields.
- Text with substitution – A text string with tags that can be substituted with the necessary values. Requires FormFields for each tag

- Free form text – A free text field. Surveyor is not bound to any text format. Requires no FormFields.

### ***Interpreting Text With Substitution***

The following is an example text with substitution string:

```
<![CDATA[Lot {parcelName} is a {parcelFormat} format remainder lot]]>
```

CDATA allows all text characters to be escaped within the square brackets.

Tags are shown in braces ({}). The value of the tag maps directly to the ID and Name of the FormField that defines its parameters. Below is an example FormField for parcelName.

```
<FormField>
  <Id>parcelName</Id>
  <Name>parcelName</Name>
  <Description>The ePlan parcel name</Description>
  <MinCardinality>1</MinCardinality>
  <MaxCardinality>1</MaxCardinality>
  <FieldType>string</FieldType>
</FormField>
```

## **A.2 Structure**

The annotations schema has the following structure. Each element name is a link to the relevant section of the document.

```
ActionStatementList
  ActionStatement
    Id
    Name
    HeadOfPower
    Explanation
    PresentationType
    Form
      Id
      Name
      Version
      FormattedName
      FormHelp
      FormTemplateUri
      FormTemplateText
      FormField
        Id
        Name
        Description
        FieldHelp
        FieldRequired
        MinCardinality
        MaxCardinality
        FieldType
        FieldTypeChoice
          FieldType
```

### A.3 ActionStatementList

<b>Description</b>	The <code>ActionStatementList</code> element is the root element of an annotation file and is a grouping element for action statements (annotations).		
<b>Example</b>	<pre>&lt;ActionStatementList version="1.0" date="2010-10-31" time="18:23:15"&gt;   &lt;ActionStatement...&gt;     ...   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt;</pre>		
<b>Parent Elements</b>	None		
<b>Child Elements</b>	<b>Cardinality</b>		
<code>ActionStatement</code>	0 - *		
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
version	string	R	The version of this file. e.g. 1.0
date	date	R	Date that this version of the administrative areas file was created. ISO 8601 format. e.g. 2010-10-31
time	time	R	Time that this version of the administrative areas file was created. ISO 8601 format. e.g. 18:23:32

### A.4 ActionStatement

<b>Description</b>	The <code>ActionStatement</code> element contains an action statement or annotation for a cadastral survey plan.		
<b>Example</b>	<pre>&lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     &lt;Id&gt;...&lt;/Id&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;HeadOfPower&gt;...&lt;/HeadOfPower&gt;     &lt;Explanation&gt;...&lt;/Explanation&gt;     &lt;PresentationType&gt;...&lt;/PresentationType&gt;     &lt;Form&gt;...&lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt;</pre>		
<b>Parent Elements</b>	<code>ActionStatementList</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
<code>Id</code>	1		
<code>Name</code>	1		
<code>HeadOfPower</code>	1 - *		
<code>Explanation</code>	0 - 1		
<code>PresentationType</code>	1		
<code>Form</code>	0 - *		

Attribute	Type	Required	Description
displaySequence	string	O	The version of this file. e.g. 1.0

## A.5 Id

<b>Description</b>	The <code>Id</code> element contains a unique identifier for an action statement, form or form field.		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     &lt;Id&gt;...&lt;/Id&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;HeadOfPower&gt;...&lt;/HeadOfPower&gt;     &lt;Explanation&gt;...&lt;/Explanation&gt;     &lt;PresentationType&gt;...&lt;/PresentationType&gt;     &lt;Form&gt;       &lt;Id&gt;...&lt;/Id&gt;       ...       &lt;FormField&gt;         &lt;Id&gt;...&lt;/Id&gt;         ...       &lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	ActionStatement Form FormField		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			



**A.6 Name**

<b>Description</b>	The <b>Name</b> element a unique name for an action statement, form or form field.		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     &lt;Id&gt;...&lt;/Id&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;HeadOfPower&gt;...&lt;/HeadOfPower&gt;     &lt;Explanation&gt;...&lt;/Explanation&gt;     &lt;PresentationType&gt;...&lt;/PresentationType&gt;     &lt;Form&gt;       &lt;Name&gt;...&lt;/Name&gt;       ...       &lt;FormField&gt;         &lt;Name&gt;...&lt;/Name&gt;         ...       &lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	ActionStatement Form FormField		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**A.7 HeadOfPower**

<b>Description</b>	The <b>HeadOfPower</b> element contains the Head of Power (such as an Act or regulation) related to the action statement.		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     &lt;Id&gt;...&lt;/Id&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;HeadOfPower&gt;...&lt;/HeadOfPower&gt;     &lt;Explanation&gt;...&lt;/Explanation&gt;     &lt;PresentationType&gt;...&lt;/PresentationType&gt;     &lt;Form&gt;...&lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	ActionStatement		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

## A.8 Explanation

<b>Description</b>	The <code>Explanation</code> element contains an explanation of the action statement. The content might be displayed to a user to assist in understanding the purpose or use of an action statement.		
<b>Example</b>	<pre>&lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     &lt;Id&gt;...&lt;/Id&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;HeadOfPower&gt;...&lt;/HeadOfPower&gt;     &lt;Explanation&gt;...&lt;/Explanation&gt;     &lt;PresentationType&gt;...&lt;/PresentationType&gt;     &lt;Form&gt;...&lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt;</pre>		
<b>Parent Elements</b>	<code>ActionStatement</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

## A.9 PresentationType

<b>Description</b>	The <code>PresentationType</code> element specifies how the annotation is completed, e.g. <code>TextWithSubstitution</code> , <code>StaticText</code> , <code>FreeFormText</code>		
<b>Example</b>	<pre>&lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     &lt;Id&gt;...&lt;/Id&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;HeadOfPower&gt;...&lt;/HeadOfPower&gt;     &lt;Explanation&gt;...&lt;/Explanation&gt;     &lt;PresentationType&gt;...&lt;/PresentationType&gt;     &lt;Form&gt;...&lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt;</pre>		
<b>Parent Elements</b>	<code>ActionStatement</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**A.10 Form**

<b>Description</b>	The <code>Form</code> element contains a form that can be used as an annotation. Note that a <code>Form</code> element must have one of either <code>FormTemplateUri</code> or <code>FormTemplateText</code> .		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     &lt;Id&gt;...&lt;/Id&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;HeadOfPower&gt;...&lt;/HeadOfPower&gt;     &lt;Explanation&gt;...&lt;/Explanation&gt;     &lt;PresentationType&gt;...&lt;/PresentationType&gt;     &lt;Form&gt;       &lt;Id&gt;...&lt;/Id&gt;       &lt;Name&gt;...&lt;/Name&gt;       &lt;Version&gt;...&lt;/Version&gt;       &lt;FormattedName&gt;...&lt;/FormattedName&gt;       &lt;FormHelp&gt;...&lt;/FormHelp&gt;       &lt;FormTemplateUri&gt;...&lt;/FormTemplateUri&gt;       &lt;FormTemplateText&gt;...&lt;/FormTemplateText&gt;       &lt;FormField&gt;...&lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	<code>ActionStatement</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
<code>Id</code>	1		
<code>Name</code>	1		
<code>Version</code>	1		
<code>FormattedName</code>	1		
<code>FormHelp</code>	0 - 1		
<code>FormTemplateUri</code>	1 (choice with <code>FormTemplateText</code> )		
<code>FormTemplateText</code>	1 (choice with <code>FormTemplateUri</code> )		
<code>FormField</code>	0 - *		
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**A.11 Version**

<b>Description</b>	The <i>Version</i> element contains the version of its parent <i>Form</i> element.		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       &lt;Id&gt;...&lt;/Id&gt;       &lt;Name&gt;...&lt;/Name&gt;       &lt;Version&gt;...&lt;/Version&gt;       &lt;FormattedName&gt;...&lt;/FormattedName&gt;       &lt;FormHelp&gt;...&lt;/FormHelp&gt;       &lt;FormTemplateUri&gt;...&lt;/FormTemplateUri&gt;       &lt;FormTemplateText&gt;...&lt;/FormTemplateText&gt;       &lt;FormField&gt;...&lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	Form		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**A.12 FormattedName**

<b>Description</b>	The <i>FormattedName</i> element contains the formatted name of the action statement, i.e. the "human readable" name which may be defined by the legislation or regulation that requires the action statement (see <i>HeadOfPower</i> ) and may be used as a title for the action statement in a visualisation.		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       &lt;Id&gt;...&lt;/Id&gt;       &lt;Name&gt;...&lt;/Name&gt;       &lt;Version&gt;...&lt;/Version&gt;       &lt;FormattedName&gt;...&lt;/FormattedName&gt;       &lt;FormHelp&gt;...&lt;/FormHelp&gt;       &lt;FormTemplateUri&gt;...&lt;/FormTemplateUri&gt;       &lt;FormTemplateText&gt;...&lt;/FormTemplateText&gt;       &lt;FormField&gt;...&lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	Form		

Child Elements			Cardinality
None			
Attribute	Type	Required	Description
None			

### A.13 FormHelp

<b>Description</b>	The <code>FormHelp</code> element contains text that can be displayed to assist with completion of the form.		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       &lt;Id&gt;...&lt;/Id&gt;       &lt;Name&gt;...&lt;/Name&gt;       &lt;Version&gt;...&lt;/Version&gt;       &lt;FormattedName&gt;...&lt;/FormattedName&gt;       &lt;FormHelp&gt;...&lt;/FormHelp&gt;       &lt;FormTemplateUri&gt;...&lt;/FormTemplateUri&gt;       &lt;FormTemplateText&gt;...&lt;/FormTemplateText&gt;       &lt;FormField&gt;...&lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	<code>Form</code>		
Child Elements			Cardinality
None			
Attribute	Type	Required	Description
None			

**A.14 FormTemplateUri**

<b>Description</b>	The <code>FormTemplateUri</code> element contains a URI to the form template. The template may use text with substitution. A form must have either a <code>FormTemplateUri</code> or <code>FormTemplateText</code> child element.		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       &lt;Id&gt;...&lt;/Id&gt;       &lt;Name&gt;...&lt;/Name&gt;       &lt;Version&gt;...&lt;/Version&gt;       &lt;FormattedName&gt;...&lt;/FormattedName&gt;       &lt;FormHelp&gt;...&lt;/FormHelp&gt;       &lt;FormTemplateUri&gt;...&lt;/FormTemplateUri&gt;       &lt;FormTemplateText&gt;...&lt;/FormTemplateText&gt;       &lt;FormField&gt;...&lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	Form		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**A.15 FormTemplateText**

<b>Description</b>	The <code>FormTemplateText</code> element contains the text to be used. A form must have either a <code>FormTemplateUri</code> or <code>FormTemplateText</code> child element.		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       &lt;Id&gt;...&lt;/Id&gt;       &lt;Name&gt;...&lt;/Name&gt;       &lt;Version&gt;...&lt;/Version&gt;       &lt;FormattedName&gt;...&lt;/FormattedName&gt;       &lt;FormHelp&gt;...&lt;/FormHelp&gt;       &lt;FormTemplateUri&gt;...&lt;/FormTemplateUri&gt;       &lt;FormTemplateText&gt;...&lt;/FormTemplateText&gt;       &lt;FormField&gt;...&lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	Form		

Child Elements			Cardinality
None			
Attribute	Type	Required	Description
None			

## A.16 FormField

<b>Description</b>	The <code>FormField</code> element contains a field component of a form, i.e. a text area that a user completes.		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       ...       &lt;FormField&gt;         &lt;Id&gt;...&lt;/Id&gt;         &lt;Name&gt;...&lt;/Name&gt;         &lt;Description&gt;...&lt;/Description&gt;         &lt;FieldHelp&gt;...&lt;/FieldHelp&gt;         &lt;FieldRequired&gt;...&lt;/FieldRequired&gt;         &lt;MinCardinality&gt;...&lt;/MinCardinality&gt;         &lt;MaxCardinality&gt;...&lt;/MaxCardinality&gt;         &lt;FieldType&gt;...&lt;/FieldType&gt;         &lt;FieldTypeChoice&gt;...&lt;/FieldTypeChoice&gt;       &lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	<code>Form</code>		
Child Elements		Cardinality	
<code>Id</code>			1
<code>Name</code>			1
<code>Description</code>			1
<code>FieldHelp</code>			0 - 1
<code>FieldRequired</code>			0 - 1
<code>MinCardinality</code>			1
<code>MaxCardinality</code>			1
<code>FieldType</code>			1 (Choice with <code>FieldTypeChoice</code> )
<code>FieldTypeChoice</code>			1 (Choice with <code>FieldType</code> )
Attribute	Type	Required	Description
None			

**A.17 Description**

<b>Description</b>	The <code>Description</code> element contains a text description of the related <code>FormField</code> .		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       ...       &lt;FormField&gt;         &lt;Id&gt;...&lt;/Id&gt;         &lt;Name&gt;...&lt;/Name&gt;         &lt;Description&gt;...&lt;/Description&gt;         &lt;FieldHelp&gt;...&lt;/FieldHelp&gt;         &lt;FieldRequired&gt;...&lt;/FieldRequired&gt;         &lt;MinCardinality&gt;...&lt;/MinCardinality&gt;         &lt;MaxCardinality&gt;...&lt;/MaxCardinality&gt;         &lt;FieldType&gt;...&lt;/FieldType&gt;         &lt;FieldTypeChoice&gt;...&lt;/FieldTypeChoice&gt;       &lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	<code>FormField</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			



**A.18 FieldHelp**

<b>Description</b>	The <code>FieldHelp</code> element contains help text that may be displayed to assist completing of the related <code>FormField</code> .		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       ...       &lt;FormField&gt;         &lt;Id&gt;...&lt;/Id&gt;         &lt;Name&gt;...&lt;/Name&gt;         &lt;Description&gt;...&lt;/Description&gt;         &lt;FieldHelp&gt;...&lt;/FieldHelp&gt;         &lt;FieldRequired&gt;...&lt;/FieldRequired&gt;         &lt;MinCardinality&gt;...&lt;/MinCardinality&gt;         &lt;MaxCardinality&gt;...&lt;/MaxCardinality&gt;         &lt;FieldType&gt;...&lt;/FieldType&gt;         &lt;FieldTypeChoice&gt;...&lt;/FieldTypeChoice&gt;       &lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	<code>FormField</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**A.19 FieldRequired**

<b>Description</b>	The <code>FieldRequired</code> element contains a boolean value that indicates whether the associated <code>FormField</code> is required. Its value is "true" or "false".		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       ...       &lt;FormField&gt;         &lt;Id&gt;...&lt;/Id&gt;         &lt;Name&gt;...&lt;/Name&gt;         &lt;Description&gt;...&lt;/Description&gt;         &lt;FieldHelp&gt;...&lt;/FieldHelp&gt;         &lt;FieldRequired&gt;...&lt;/FieldRequired&gt;         &lt;MinCardinality&gt;...&lt;/MinCardinality&gt;         &lt;MaxCardinality&gt;...&lt;/MaxCardinality&gt;         &lt;FieldType&gt;...&lt;/FieldType&gt;         &lt;FieldTypeChoice&gt;...&lt;/FieldTypeChoice&gt;       &lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	<code>FormField</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

## A.20 MinCardinality

<b>Description</b>	The <code>MinCardinality</code> element contains an integer value that specifies the minimum number of this type of <code>FormField</code> that may be in a <code>Form</code> .		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       ...       &lt;FormField&gt;         &lt;Id&gt;...&lt;/Id&gt;         &lt;Name&gt;...&lt;/Name&gt;         &lt;Description&gt;...&lt;/Description&gt;         &lt;FieldHelp&gt;...&lt;/FieldHelp&gt;         &lt;FieldRequired&gt;...&lt;/FieldRequired&gt;         &lt;MinCardinality&gt;...&lt;/MinCardinality&gt;         &lt;MaxCardinality&gt;...&lt;/MaxCardinality&gt;         &lt;FieldType&gt;...&lt;/FieldType&gt;         &lt;FieldTypeChoice&gt;...&lt;/FieldTypeChoice&gt;       &lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	<code>FormField</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

### A.21 MaxCardinality

<b>Description</b>	The <code>MaxCardinality</code> element contains an integer value that specifies the maximum number of this type of <code>FormField</code> that may be in a <code>Form</code> .		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       ...       &lt;FormField&gt;         &lt;Id&gt;...&lt;/Id&gt;         &lt;Name&gt;...&lt;/Name&gt;         &lt;Description&gt;...&lt;/Description&gt;         &lt;FieldHelp&gt;...&lt;/FieldHelp&gt;         &lt;FieldRequired&gt;...&lt;/FieldRequired&gt;         &lt;MinCardinality&gt;...&lt;/MinCardinality&gt;         &lt;MaxCardinality&gt;...&lt;/MaxCardinality&gt;         &lt;FieldType&gt;...&lt;/FieldType&gt;         &lt;FieldTypeChoice&gt;...&lt;/FieldTypeChoice&gt;       &lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	<code>FormField</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**A.22 FieldType**

<b>Description</b>	<p>The <code>FieldType</code> element defines the type of the content for the field, the content may be one of: boolean, date, dateTime, decimal, double, duration, float, string or time.</p> <p>A <code>FormField</code> must have either a <code>FieldTypeChoice</code> or <code>FieldType</code> child element.</p>		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     ...     &lt;Form&gt;       ...       &lt;FormField&gt;         &lt;Id&gt;...&lt;/Id&gt;         &lt;Name&gt;...&lt;/Name&gt;         &lt;Description&gt;...&lt;/Description&gt;         &lt;FieldHelp&gt;...&lt;/FieldHelp&gt;         &lt;FieldRequired&gt;...&lt;/FieldRequired&gt;         &lt;MinCardinality&gt;...&lt;/MinCardinality&gt;         &lt;MaxCardinality&gt;...&lt;/MaxCardinality&gt;         &lt;FieldType&gt;...&lt;/FieldType&gt;         &lt;FieldTypeChoice&gt;...&lt;/FieldTypeChoice&gt;       &lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	<p><code>FormField</code> <code>FieldTypeChoice</code></p>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**A.23 FieldTypeChoice**

<b>Description</b>	<p>The <code>FieldTypeChoice</code> element allows a field to have more than one type, i.e. it may be two or more of the types specified by <code>FieldType</code>.</p> <p>A <code>FormField</code> must have either a <code>FieldTypeChoice</code> or <code>FieldType</code> child element.</p>		
<b>Example</b>	<pre> &lt;ActionStatementList ...&gt;   &lt;ActionStatement displaySequence="..."&gt;     &lt;Id&gt;...&lt;/Id&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;HeadOfPower&gt;...&lt;/HeadOfPower&gt;     &lt;Explanation&gt;...&lt;/Explanation&gt;     &lt;PresentationType&gt;...&lt;/PresentationType&gt;     &lt;Form&gt;       ...       &lt;FormField&gt;         &lt;FieldType&gt;...&lt;/FieldType&gt;         &lt;FieldTypeChoice&gt;           &lt;FieldType&gt;...&lt;/FieldType&gt;           ...         &lt;/FieldTypeChoice&gt;       &lt;/FormField&gt;     &lt;/Form&gt;   &lt;/ActionStatement&gt;   ... &lt;/ActionStatementList&gt; </pre>		
<b>Parent Elements</b>	<code>FormField</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
<code>FieldType</code>	1 - *		
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

## Appendix B Surveyor Certificates Schema

### B.1 Implementation Guidelines

The Survey Certificates schema instance is defined at the jurisdiction level using the survey certificates schema (xml-gov-au-icsm-eplan-cif-survey-certificates-1.0.xsd). This file provides a list of acceptable certificates for use in the jurisdiction. Each jurisdiction will validate their certificates against the list they supply in this file. The values can be a static text value or a string with text substitution.

An example file name for this file conforming to Appendix G is:

```
xml-gov-au-wa-icsm-eplan-cif-survey-certificates-1.0-20101225.xsd
```

The format of the certificates file is the same as for annotations (see Appendix A *Interpreting Text With Substitution*).

The following is an annotated skeleton template:

```
<SurveyCertificateList time="" date="" version="" xmlns="urn:xml-gov-
au:icsm:eplan:cif:survey-certificates:1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="urn:xml-gov-au:icsm:eplan:cif:survey-certificates:1.0
U:\ePlan\Specifications\BUSINE~1\National\EPLANP~3\xml-gov-au-icsm-eplan-cif-survey-
certificates-1.0.xsd">
  <SurveyCertificate displaySequence="1"> <!-- display order -->
    <Id></Id> <!-- surrogate ID -->
    <Name></Name> <!-- certificate category -->
    <HeadOfPower></HeadOfPower> <!-- Act or legislation this applies to -->
    <PresentationType></PresentationType> <!-- one of: StaticText, TextWithSubstitution,
FreeFormText -->
    <Form>
      <Id></Id> <!-- surrogate ID -->
      <Name></Name> <!-- specific certificate name -->
      <Version></Version> <!-- certificate version -->
      <FormattedName></FormattedName> <!-- full name with version -->
      <FormTemplateText><![CDATA[]]></FormTemplateText> <!-- certificate content in
CDATA -->
      <FormField> <!-- if text substitution is used, each tag is defined with a
formfield -->
        <Id></Id> <!-- tag value -->
        <Name></Name> <!-- same as ID -->
        <Description></Description> <!-- desc of the value -->
        <MinCardinality></MinCardinality> <!-- the range of values required -->
        <MaxCardinality></MaxCardinality>
        <FieldType></FieldType> <!-- the type e.g. string, int, double etc. -->
      </FormField>
    </Form>
  </SurveyCertificate>
</SurveyCertificateList>
```

### **Interpreting Text With Substitution**

See section A.1 *Interpreting Text With Substitution*.

### B.2 Structure

The Survey Certificates schema uses the same structure as the Annotations schema in Appendix A.





## Appendix C Administrative Area Schema

The administrative area schema defines the structure and relationships for the administrative areas data file. It applies restrictions to some value so that they can only be used with other values, e.g. restricting the use of certain localities depending on the local government area.

### C.1 Implementation Guidelines

The Administrative Areas schema instance specifies the valid values for the types of administrative areas used in the jurisdiction. For example, the type of LGA contains the values of "Melbourne City Council" and "Moonee Valley City Council" in Victoria.

An example file name for this file conforming to Appendix G is:

```
xml-gov-au-nt-icsm-eplan-cif-administrative-area-1.0-20101225.xml
```

The following is a skeleton of an Administrative Area XML instance:

```
<AdministrativeAreasAndRelationships
  xmlns="urn:xml-gov-au:icsm:eplan:cif:administrative-area:1.0"
  xmlns:xsi="" xsi:schemaLocation="" version="" date="" time="">
  <AdministrativeAreaGroup type="">
    <AdministrativeArea createDate="" retiredDate="">
      <Id><!--Jurisdictional ID--></Id>
      <Name><!--Formal Name--></Name>
      <RestrictedTo type="">
        <AdministrativeArea>
          <Id><!--ID of the super admin area--></Id>
        </AdministrativeArea>
      </RestrictedTo>
    </AdministrativeArea>

    <AdministrativeArea createDate="" retiredDate="">
      ...
    </AdministrativeArea>
    <!--List of AdministrativeArea elements-->
  </AdministrativeAreaGroup>

  <AdministrativeAreaGroup type="">
    ...
  </AdministrativeAreaGroup>
  <!--List of AdministrativeAreaGroup elements-->
</AdministrativeAreasAndRelationships>
```

### Administrative Area Type

The `AdministrativeAreaGroup type` attribute values are predefined by the Enumerations Schema, such as LGA and Parish. The Administrative Area Schema imports the Enumerations schema to allow access to these values. Below is the import statement from the Administrative Area Schema. It must be set by each jurisdiction to import the correct Enumerations Schema.

```
<xs:import namespace="urn:xml-gov-au:qld:icsm:eplan:cif:enumerated-
types:1.0" schemaLocation="../enumerated-types/xml-gov-au-qld-icsm-eplan-
cif-enumerated-types-1.0.xsd"/>
```

### Administrative Area Restriction

The `RestrictedTo` element allows the record to be restricted to other administrative area records. It specifies that the record is only valid in the CIF if the `RestrictedTo` administrative areas are also specified. Uses include where an administrative area spatially falls within another. For example, where a Parish falls inside a County or a Locality falls inside an LGA.

### Versioning

The Administrative Areas file is versioned internally to facilitate backward compatibility. Every Administrative area record contains a `createDate` and `retiredDate` attribute which specifies whether records are active or retired. All records must have a `createDate` value. A record is deemed active if it has no `retiredDate` value. Retired records must have a `retiredDate` value.

The Administrative Areas file is a continuously updated list. Additions and amendments are made as updates to the file enabling backward compatibility. This negates the need for maintaining multiple versions of the file. When new versions are released, third-party consumers can simply override the existing file as contents of the old file are always carried forward in the newer version.

Jurisdictions may choose at their discretion whether to maintain official version numbers for the Administrative Areas file. Alternatively, a date stamp can be used for versioning.

## C.2 Structure

The administrative area schema has the following structure. Each element name is a link to the relevant section of the document.

```
AdministrativeAreasAndRelationships
  AdministrativeAreaGroup
    AdministrativeArea
      Name
      Id
      RestrictedTo
```

## C.3 AdministrativeAreasAndRelationships

<b>Description</b>	The <code>AdministrativeAreasAndRelationships</code> element is the root type for an administrative areas data file.	
<b>Example</b>	<pre>&lt;AdministrativeAreasAndRelationships version="1.0" date="2010-210-31" time="18:23:32"&gt;   ... &lt;/AdministrativeAreasAndRelationships&gt;</pre>	
<b>Parent Element</b>	None	
<b>Child Elements</b>	<b>Cardinality</b>	
<code>AdministrativeAreaGroup</code>	1 - *	

Attribute	Type	Required	Description
version	string	R	The version of this file. e.g. 1.0
date	date	R	Date that this version of the administrative areas file was created. ISO 8601 format. e.g. 2010-10-31
time	time	R	Time that this version of the administrative areas file was created. ISO 8601 format. e.g. 18:23:32

#### C.4 AdministrativeAreaGroup

<b>Description</b>	The <code>AdministrativeAreaGroup</code> element is grouping element for <code>AdministrativeArea</code> elements of the same type.		
<b>Example</b>	<pre>&lt;AdministrativeAreasAndRelationships...&gt;   &lt;AdministrativeAreaGroup type="..."&gt;     ...   &lt;AdministrativeAreaGroup type="..."&gt;     ... &lt;/AdministrativeAreasAndRelationships &gt;</pre>		
<b>Parent Element</b>	<code>AdministrativeAreasAndRelationships</code>		
<b>Child Elements</b>			<b>Cardinality</b>
	<code>AdministrativeArea</code>		1 - *
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
type	string	O	The type of administrative area in this group. e.g. County

#### C.5 AdministrativeArea

<b>Description</b>	The <code>AdministrativeArea</code> element has the name for an administrative area, such as a locality.		
<b>Example</b>	<pre>&lt;AdministrativeAreasAndRelationships...&gt;   &lt;AdministrativeAreaGroup ...&gt;     &lt;AdministrativeArea createDate="2005-01-01"       retireDate="2010-06-30"&gt;       ...     &lt;/AdministrativeArea&gt;     ...   &lt;/AdministrativeAreaGroup&gt;   ... &lt;/AdministrativeAreasAndRelationships &gt;</pre>		
<b>Parent Element</b>	<code>AdministrativeAreaGroup</code> <code>RestrictedTo</code>		
<b>Child Elements</b>			<b>Cardinality</b>
	<code>Id</code>		1
	<code>Name</code>		0 - 1
	<code>RestrictedTo</code>		0 - *

Attribute	Type	Required	Description
createDate	date	O	Date that the administrative area name comes into force.
retireDate	date	O	Date after which the administrative area name no longer exists.

## C.6 Name

<b>Description</b>	The <code>Name</code> element has the name for the administrative area.		
<b>Example</b>	<pre> &lt;AdministrativeAreasAndRelationships ...&gt;   &lt;AdministrativeAreaGroup ...&gt;     &lt;AdministrativeArea ...&gt;       &lt;Name&gt;BARCOO SHIRE&lt;/Name&gt;       &lt;Id&gt;...&lt;/Id&gt;       &lt;RestrictedTo&gt;...&lt;/RestrictedTo&gt;     &lt;/AdministrativeArea&gt;     ...   &lt;/AdministrativeAreaGroup&gt;   ... &lt;/AdministrativeAreasAndRelationships &gt; </pre>		
<b>Parent Element</b>	<code>AdministrativeArea</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

## C.7 Id

<b>Description</b>	The <code>Id</code> element has an identifier for the administrative area. It may be used by a jurisdiction for internal processing and may not be suitable for display or end user functionality.		
<b>Example</b>	<pre> &lt;AdministrativeAreasAndRelationships...&gt;   &lt;AdministrativeAreaGroup ...&gt;     &lt;AdministrativeArea ...&gt;       &lt;Name&gt;...&lt;/Name&gt;       &lt;Id&gt;450&lt;/Id&gt;       &lt;RestrictedTo&gt;...&lt;/RestrictedTo&gt;     &lt;/AdministrativeArea&gt;     ...   &lt;/AdministrativeAreaGroup&gt;   ... &lt;/AdministrativeAreasAndRelationships &gt; </pre>		
<b>Parent Element</b>	<code>AdministrativeArea</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

## C.8 RestrictedTo

<b>Description</b>	<p>The <code>RestrictedTo</code> defines a restriction on the associated administrative area. It can be used to restrict the use of an administrative area so that it must be used with another administrative area, e.g. restricting the use of a locality to a certain value for Local Government Area.</p> <p>Note that an administrative area can't be restricted to another administrative area of the same type (e.g. a locality can't be restricted to another locality).</p>		
<b>Example</b>	<pre>&lt;AdministrativeAreasAndRelationships...&gt;   &lt;AdministrativeAreaGroup ...&gt;     &lt;AdministrativeArea ...&gt;       &lt;Name&gt;...&lt;/Name&gt;       &lt;Id&gt;...&lt;/Id&gt;       &lt;RestrictedTo type="..."&gt;         &lt;AdministrativeArea ...&gt;...&lt;/AdministrativeArea&gt;       &lt;/RestrictedTo&gt;     &lt;/AdministrativeArea&gt;     ...   &lt;/AdministrativeAreaGroup&gt;   ... &lt;/AdministrativeAreasAndRelationships &gt;</pre>		
<b>Parent Element</b>	AdministrativeArea		
<b>Child Elements</b>			<b>Cardinality</b>
	AdministrativeArea		0 - *
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
type	string	O	An administrativeAreaGroupType e.g. county

## C.9 Example Administrative Area data file generated using the schema

```
<?xml version="1.0" encoding="UTF-8"
  xmlns:urn="urn:xml-gov-au:icsm:eplan:cif:administrative-area:1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:xml-gov-au:icsm:eplan:cif:administrative-
area:1.0 .\xml-gov-au-icsm-eplan-cif-administrative-area-1.0.xsd">
  <!-- Define a group called "Local Government Area" -->
  <urn:AdministrativeAreaGroup type="Local Government Area">
    <!-- Define the values for "Local Government Area" -->
    <urn:AdministrativeArea>
      <Id>760</Id>
      <Name>Blackall Tambo Regional Council</Name>
    </urn:AdministrativeArea>
    ...
  </urn:AdministrativeAreaGroup>
  <!-- Define another group called "Locality" -->
  <urn:AdministrativeAreaGroup type="Locality">
    <!-- Define the values for "Locality" -->
```

```
<urn:AdministrativeArea>
<Id>5147</Id>
<Name>Bayrick</Name>
  <!-- Restrict the use of this value of Locality to a
  particular value for "Local Government Area" -->
  <RestrictedTo type="Local Government Area">
    <urn:AdministrativeArea>
      <Id>760</Id>
    </urn:AdministrativeArea>
  </RestrictedTo>
</urn:AdministrativeArea>
...
</urn:AdministrativeAreaGroup>
...
</urn:AdministrativeAreasAndRelationships>
```

## Appendix D Enumerated Types Schema

The Enumerated Types Schema provides defined sets of values for ePlan attribute types. It is used by various ePlan schemas for specifying a restricted set of values to be used for particular attribute types.

For example, the LandXML schema specifies an `AdministrativeArea` element with an `adminAreaType` attribute. The ePlan Protocol restricts the value to `adminAreaTypeType`. The enumerated types schema provides a list of values for `adminAreaTypeType`, in Queensland the list is: Local Government Area, Locality, County, Parish and Mining District. Therefore the value for `adminAreaType` in a Queensland CIF must be one of these values.

Other ePlan schemas also use the enumerated types schema, such as the Survey Certificates Schema.

### D.1 Implementation Guidelines

The following is a sample comparison of the template to a jurisdictional implementation.

```
<xs:simpleType name="addressTypeType">
  <xs:annotation>
    <xs:documentation></xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string"/>
</xs:simpleType>
```

**Template**

```
<xs:simpleType name="addressTypeType">
  <xs:annotation>
    <xs:documentation> </xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="Primary"/>
    <xs:enumeration value="Historical"/>
    <xs:enumeration value="Alias"/>
  </xs:restriction>
</xs:simpleType>
```

**Jurisdictional Implementation**

Jurisdictions should include all types contained in the template. If a type is not used, then it should be left as its default value. Jurisdictions are free to add their own `documentation` to each type.

### Namespace

The namespace of this schema should take the form of:

```
urn:xml-gov-au:<juris>:icsm:eplan:cif:enumerated-types:<version>
```

`<juris>` is replaced with the jurisdiction identifier e.g. qld or vic. `<version>` is replaced with the version number of this schema.

The file name convention must conform to the AGIMO standards for schema file names. See Appendix G for further details.

## D.2 Structure

The enumerated types schema is implemented as an XML schema (XSD), it does not enforce any structure, it only provides values and constraints for attribute values. It differs from other ePlan schemas in that the data is in the schema, there is no separate data file created from the schema.

The following structure is a basic W3C XSD. Each element name is a link to the relevant section of the document.

```

schema
  simpleType
    annotation
      documentation
    restriction
    enumeration

```

## D.3 schema

<b>Description</b>	The <code>schema</code> element defines the schemas used for this schema. It is the root element.		
<b>Example</b>	<pre> &lt;schema ...&gt;   &lt;simpleType ...&gt;   &lt;/simpleType&gt;   ... &lt;/schema&gt; </pre>		
<b>Parent Element</b>	None		
<b>Child Elements</b>	<b>Cardinality</b>		
<code>simpleType</code>	1 - *		
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
xmlns	string	R	Defines the base xml namespace for this schema.
targetNamespace	string	R	Defines the output names space for instances of the schema.
elementFormDefault	string	R	Set to "qualified"
attributeFormDefault	string	R	Set to "unqualified"
version	string	R	Version of this schema, e.g. "2.0"



**D.4 simpleType**

<b>Description</b>	The <code>simpleType</code> element defines a simple type of enumeration.		
<b>Example</b>	<pre> &lt;schema ...&gt;   &lt;simpleType ...&gt;     &lt;annotation&gt;       ...     &lt;/annotation&gt;     &lt;restriction ...&gt;       ...     &lt;/restriction&gt;   &lt;/simpleType&gt;   ... &lt;/schema&gt; </pre>		
<b>Parent Element</b>	<code>schema</code>		
<b>Child Elements</b>			<b>Cardinality</b>
<code>annotation</code> <code>restriction</code>			0 - * 1
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
name	string	R	The name of the enumeration type. e.g. " <code>addressTypeType</code> " is the name of the enumeration type used for <code>addressType</code> .

**D.5 annotation**

<b>Description</b>	The <code>annotation</code> element is a grouping element for <code>documentation</code> elements that provide notes about an enumeration.		
<b>Example</b>	<pre> &lt;schema ...&gt;   &lt;simpleType ...&gt;     &lt;annotation&gt;       &lt;Documentation&gt;...&lt;/Documentation&gt;     &lt;/annotation&gt;     &lt;restriction ...&gt;       ...     &lt;/restriction&gt;   &lt;/simpleType&gt;   ... &lt;/schema&gt; </pre>		
<b>Parent Element</b>	<code>simpleType</code>		
<b>Child Elements</b>			<b>Cardinality</b>
<code>documentation</code>			1 - *
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**D.6 documentation**

<b>Description</b>	The <code>Documentation</code> element provides notes about the annotation.		
<b>Example</b>	<pre> &lt;schema ...&gt;   &lt;simpleType ...&gt;     &lt;annotation&gt;       &lt;documentation&gt;...&lt;/documentation&gt;     &lt;/annotation&gt;     &lt;restriction ...&gt;       ...     &lt;/restriction&gt;   &lt;/simpleType&gt;   ... &lt;/schema&gt; </pre>		
<b>Parent Element</b>	<code>annotation</code>		
<b>Child Elements</b>		<b>Cardinality</b>	
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**D.7 restriction**

<b>Description</b>	The <code>restriction</code> element defines the base type of content (e.g. string, date) for the enumeration and the values that may be used.		
<b>Example</b>	<pre> &lt;schema ...&gt;   &lt;simpleType ...&gt;     ...     &lt;restriction base="string"&gt;       &lt;enumeration .../&gt;     &lt;/restriction&gt;   &lt;/simpleType&gt;   ... &lt;/schema&gt; </pre>		
<b>Parent Element</b>	<code>simpleType</code>		
<b>Child Elements</b>		<b>Cardinality</b>	
<code>enumeration</code>		1 - *	
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
<code>base</code>	<code>string</code>	R	Defines the base type of the enumeration values, e.g. string, date, .

**D.8 enumeration**

<b>Description</b>	The <code>enumeration</code> element contains a value for the related enumeration type.		
<b>Example</b>	<pre> &lt;schema ...&gt;   &lt;simpleType name="adminAreaTypeType"&gt;     &lt;annotation&gt;       &lt;documentation&gt; This is a jurisdictionally specific         list of types and may include parish, town, local         government, locality etc.&lt;/documentation&gt;     &lt;/annotation&gt;     &lt;restriction base="string"&gt;       &lt;enumeration value="Local Government Area"/&gt;       &lt;enumeration value="Locality"/&gt;       &lt;enumeration value="Parish"/&gt;       &lt;enumeration value="County"/&gt;       ...     &lt;/restriction&gt;   &lt;/simpleType&gt;   ... &lt;/schema&gt; </pre>		
<b>Parent Element</b>	<code>restriction</code>		
<b>Child Elements</b>		<b>Cardinality</b>	
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
value	string	R	A value for the related enumeration type.

## Appendix E Reference Data Context Schema

The reference data context schema defines the reference data context file, see § 3.1.

### E.1 Structure

The reference data context schema has the following structure. Each element name is a link to its description.

```

ReferenceDataContextList
  LastUpdated
  ReferenceDataContext
    ContextId
    ContextStatus
    SupersedingContextId
    Name
    Description
    ApplicableFrom
    ApplicableTo
    Content
      ContentId
      Source
      SourceVersion
      Description
      FileType

```

### E.2 ReferenceDataContextList

<b>Description</b>	The <code>ReferenceDataContextList</code> element is the root element. It contains a series of reference data context elements that define a set of reference data files.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated .../&gt;   &lt;ReferenceDataContext&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	None		
<b>Child Elements</b>	<b>Cardinality</b>		
<code>LastUpdated</code>	1		
<code>ReferenceDataContext</code>	1 - *		
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

### E.3 LastUpdated

<b>Description</b>	The <code>LastUpdated</code> element content is the date that the file was last updated in ISO8601 format.		
<b>Example</b>	<pre>&lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt;</pre>		
<b>Parent Element</b>	<code>ReferenceDataContextList</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

### E.4 ReferenceDataContext

<b>Description</b>	The <code>ReferenceDataContext</code> element contains information on a reference file to be used with the related list of reference files.		
<b>Example</b>	<pre>&lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     &lt;ContextId&gt;...&lt;/ContextId&gt;     &lt;ContextStatus&gt;...&lt;/ContextStatus&gt;     &lt;SupersedingContextId&gt;...&lt;/SupersedingContextId&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;Description&gt;...&lt;/Description&gt;     &lt;ApplicableFrom&gt;...&lt;/ApplicableFrom&gt;     &lt;ApplicableTo&gt;...&lt;/ApplicableTo&gt;     &lt;Content&gt;       ...     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt;</pre>		
<b>Parent Element</b>	<code>ReferenceDataContextList</code>		
<b>Child Elements</b>	<b>Cardinality</b>		
<code>ContextId</code>	1		
<code>ContextStatus</code>	1		
<code>SupersedingContextId</code>	0 - 1		
<code>Name</code>	0 - 1		
<code>Description</code>	0 - 1		
<code>ApplicableFrom</code>	0 - 1		
<code>ApplicableTo</code>	0 - 1		
<code>Content</code>	1 - *		
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**E.5 ContextId**

<b>Description</b>	The <code>ContextId</code> element contains a unique identifier for the context.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     &lt;ContextId&gt;NRWQLD-RD-6&lt;/ContextId&gt;     &lt;ContextStatus&gt;...&lt;/ContextStatus&gt;     &lt;SupersedingContextId&gt;...&lt;/SupersedingContextId&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;Description&gt;...&lt;/Description&gt;     &lt;ApplicableFrom&gt;...&lt;/ApplicableFrom&gt;     &lt;ApplicableTo&gt;...&lt;/ApplicableTo&gt;     &lt;Content&gt;       ...     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	ReferenceDataContext		
<b>Child Elements</b>			<b>Cardinality</b>
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**E.6 ContextStatus**

<b>Description</b>	The <code>ContextStatus</code> element contains the status for the context, which may be Active or Superseded.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     &lt;ContextId&gt;...&lt;/ContextId&gt;     &lt;ContextStatus&gt;Active&lt;/ContextStatus&gt;     &lt;SupersedingContextId&gt;...&lt;/SupersedingContextId&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;Description&gt;...&lt;/Description&gt;     &lt;ApplicableFrom&gt;...&lt;/ApplicableFrom&gt;     &lt;ApplicableTo&gt;...&lt;/ApplicableTo&gt;     &lt;Content&gt;       ...     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	ReferenceDataContext		
<b>Child Elements</b>			<b>Cardinality</b>
None			

Attribute	Type	Required	Description
None			

## E.7 SupersedingContextId

<b>Description</b>	The <code>SupersedingContextId</code> element contains the identifier for the context that supersedes this context.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     &lt;ContextId&gt;NRWQLD-RD-6&lt;/ContextId&gt;     &lt;ContextStatus&gt;Superseded&lt;/ContextStatus&gt;     &lt;SupersedingContextId&gt;Q-RD-7&lt;/SupersedingContextId&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;Description&gt;...&lt;/Description&gt;     &lt;ApplicableFrom&gt;...&lt;/ApplicableFrom&gt;     &lt;ApplicableTo&gt;...&lt;/ApplicableTo&gt;     &lt;Content&gt;       ...     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	<code>ReferenceDataContext</code>		
<b>Child Elements</b>		<b>Cardinality</b>	
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**E.8 Name**

<b>Description</b>	The <b>Name</b> element contains the name of the context.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     &lt;ContextId&gt;...&lt;/ContextId&gt;     &lt;ContextStatus&gt;...&lt;/ContextStatus&gt;     &lt;SupersedingContextId&gt;...&lt;/SupersedingContextId&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;Description&gt;...&lt;/Description&gt;     &lt;ApplicableFrom&gt;...&lt;/ApplicableFrom&gt;     &lt;ApplicableTo&gt;...&lt;/ApplicableTo&gt;     &lt;Content&gt;       ...     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	ReferenceDataContext		
<b>Child Elements</b>			<b>Cardinality</b>
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**E.9 Description**

<b>Description</b>	The <b>Description</b> element contains a description of the context, e.g. "Introduces new survey certificates as required by Surveying and Mapping Act, 2005".		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     &lt;ContextId&gt;...&lt;/ContextId&gt;     &lt;ContextStatus&gt;...&lt;/ContextStatus&gt;     &lt;SupersedingContextId&gt;...&lt;/SupersedingContextId&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;Description&gt;...&lt;/Description&gt;     &lt;ApplicableFrom&gt;...&lt;/ApplicableFrom&gt;     &lt;ApplicableTo&gt;...&lt;/ApplicableTo&gt;     &lt;Content&gt;       ...     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	ReferenceDataContext		



Child Elements			Cardinality
None			
Attribute	Type	Required	Description
None			

## E.10 ApplicableFrom

<b>Description</b>	The <code>ApplicableFrom</code> element contains a date in ISO8601 format that specifies the date from which the context is applicable, see § 3.2.4. Note that <code>ApplicableFrom</code> and <code>ApplicableTo</code> dates can overlap.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     &lt;ContextId&gt;...&lt;/ContextId&gt;     &lt;ContextStatus&gt;...&lt;/ContextStatus&gt;     &lt;SupersedingContextId&gt;...&lt;/SupersedingContextId&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;Description&gt;...&lt;/Description&gt;     &lt;ApplicableFrom&gt;...&lt;/ApplicableFrom&gt;     &lt;ApplicableTo&gt;...&lt;/ApplicableTo&gt;     &lt;Content&gt;       ...     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	<code>ReferenceDataContext</code>		
Child Elements			Cardinality
None			
Attribute	Type	Required	Description
None			

## E.11 ApplicableTo

<b>Description</b>	The <code>ApplicableTo</code> element contains a date in ISO8601 format that specifies the date after which the context is not applicable, see § 3.2.4. Note that <code>ApplicableFrom</code> and <code>ApplicableTo</code> dates can overlap.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     &lt;ContextId&gt;...&lt;/ContextId&gt;     &lt;ContextStatus&gt;...&lt;/ContextStatus&gt;     &lt;SupersedingContextId&gt;...&lt;/SupersedingContextId&gt;     &lt;Name&gt;...&lt;/Name&gt;     &lt;Description&gt;...&lt;/Description&gt;     &lt;ApplicableFrom&gt;...&lt;/ApplicableFrom&gt;     &lt;ApplicableTo&gt;...&lt;/ApplicableTo&gt;     &lt;Content&gt;       ...     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	ReferenceDataContext		
<b>Child Elements</b>			<b>Cardinality</b>
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

## E.12 Content

<b>Description</b>	The <code>Content</code> element contains information on a reference data file to be used for the context.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     ...     &lt;Content&gt;       &lt;ContentId&gt;...&lt;/ContentId&gt;       &lt;Source&gt;...&lt;/Source&gt;       &lt;SourceVersion&gt;...&lt;/SourceVersion&gt;       &lt;Description&gt;...&lt;/Description &gt;       &lt;FileType&gt;...&lt;/FileType&gt;     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	ReferenceDataContext		

Child Elements			Cardinality
ContentId			1
Source			1
SourceVersion			1
Description			0 - 1
FileType			0 - 1
Attribute	Type	Required	Description
None			

### E.13 ContentId

<b>Description</b>	The <code>ContentId</code> element contains a unique identifier for the content.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     ...     &lt;Content&gt;       &lt;ContentId&gt;...&lt;/ContentId&gt;       &lt;Source&gt;...&lt;/Source&gt;       &lt;SourceVersion&gt;...&lt;/SourceVersion&gt;       &lt;Description&gt;...&lt;/Description &gt;       &lt;FileType&gt;...&lt;/FileType&gt;     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	Content		
Child Elements			Cardinality
None			
Attribute	Type	Required	Description
None			

**E.14 Source**

<b>Description</b>	The <code>ContentId</code> element contains a unique identifier for the content.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     ...     &lt;Content&gt;       &lt;ContentId&gt;...&lt;/ContentId&gt;       &lt;Source&gt;...&lt;/Source&gt;       &lt;SourceVersion&gt;...&lt;/SourceVersion&gt;       &lt;Description&gt;...&lt;/Description &gt;       &lt;FileType&gt;...&lt;/FileType&gt;     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	Content		
<b>Child Elements</b>			<b>Cardinality</b>
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**E.15 SourceVersion**

<b>Description</b>	The <code>SourceVersion</code> element contains the version of the reference document in this <code>Content</code> element.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     ...     &lt;Content&gt;       &lt;ContentId&gt;...&lt;/ContentId&gt;       &lt;Source&gt;...&lt;/Source&gt;       &lt;SourceVersion&gt;...&lt;/SourceVersion&gt;       &lt;Description&gt;...&lt;/Description &gt;       &lt;FileType&gt;...&lt;/FileType&gt;     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	Content		
<b>Child Elements</b>			<b>Cardinality</b>
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**E.16 Description**

<b>Description</b>	The <i>Description</i> element contains a brief description of this <i>Content</i> item.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     ...     &lt;Content&gt;       &lt;ContentId&gt;...&lt;/ContentId&gt;       &lt;Source&gt;...&lt;/Source&gt;       &lt;SourceVersion&gt;...&lt;/SourceVersion&gt;       &lt;Description&gt;...&lt;/Description &gt;       &lt;FileType&gt;...&lt;/FileType&gt;     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	Content		
<b>Child Elements</b>			<b>Cardinality</b>
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

**E.17 FileType**

<b>Description</b>	The <i>FileType</i> element describes the file type of file source file, e.g. xml, xsd.		
<b>Example</b>	<pre> &lt;ReferenceDataContextList ...&gt;   &lt;LastUpdated&gt;2010-10-30&lt;/LastUpdated&gt;   &lt;ReferenceDataContext&gt;     ...     &lt;Content&gt;       &lt;ContentId&gt;...&lt;/ContentId&gt;       &lt;Source&gt;...&lt;/Source&gt;       &lt;SourceVersion&gt;...&lt;/SourceVersion&gt;       &lt;Description&gt;...&lt;/Description &gt;       &lt;FileType&gt;...&lt;/FileType&gt;     &lt;/Content&gt;     ...   &lt;/ReferenceDataContext&gt;   ... &lt;/ReferenceDataContextList&gt; </pre>		
<b>Parent Element</b>	Content		
<b>Child Elements</b>			<b>Cardinality</b>
None			
<b>Attribute</b>	<b>Type</b>	<b>Required</b>	<b>Description</b>
None			

## E.18 Example Reference Data Context File

```
<ReferenceDataContextList
  xmlns="urn:xml:gov:au:icsm:eplan:cif:referencedata:2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:xml:gov:au:icsm:eplan:cif:referencedata:2.0
.\xml-gov-au-icsm-eplan-cif-referencedata-2.0.xsd">
  <LastUpdated>2010-11-16</LastUpdated>
  <ReferenceDataContext>
    <ContextId>QLD-RD-6</ContextId>
    <ContextStatus>Active</ContextStatus>
    <Name>LandXML1.2 Release</Name>
    <Description>First Release of LandXML1.2 Compliant Schemas for
Queensland</Description>
    <ApplicableFrom>2010-11-16</ApplicableFrom>
    <Content>
      <ContentId>enumerations</ContentId>
      <Source>file://path.to.some.location/xml-gov-au-qld-icsm-eplan-
cif-enumerated-types-1.0.xsd</Source>
      <SourceVersion>1</SourceVersion>
      <Description>First Release of 1.2 Compliant Schemas</Description>
      <FileType>xsd</FileType>
    </Content>
    <Content>
      <ContentId>certificates</ContentId>
      <Source> file://path.to.some.location/xml-gov-au-qld-icsm-eplan-
cif-survey-certificates-1.0-20101225.xml</Source>
      <SourceVersion>1</SourceVersion>
      <Description>First Release of 1.2 Compliant Schemas</Description>
      <FileType>xml</FileType>
    </Content>
    <Content>
      <ContentId>annotations</ContentId>
      <Source> file://path.to.some.location/xml-gov-au-qld-icsm-eplan-
cif-annotations-1.0-20101225.xml</Source>
      <SourceVersion>1</SourceVersion>
      <Description>First Release of 1.2 Compliant Schemas</Description>
      <FileType>xml</FileType>
    </Content>
  </ReferenceDataContext>
</ReferenceDataContextList>
```

## Appendix F Jurisdictional CIF Schema Implementation Guidelines

The Jurisdictional CIF Schema is a jurisdictional subset of the ePlan Protocol Schema that is specific to the CIF content requirements of the jurisdiction. Any CIF created against a jurisdictional CIF schema must also successfully validate against the ePlan Protocol Schema and LandXML 1.2.

The differences between the Jurisdictional CIF Schema and its super schemas are the following:

- The exclusion of optional elements specified in the ePlan Protocol Schema.
- The cardinality of elements.
- The usage requirements for attributes.

Optional elements in the ePlan Protocol Schema are those with a cardinality of 0 - \* or 0 - 1. The XML schema specifies a range using the attributes, `minOccurs` (first number in the range) and `maxOccurs` (last number in the range). Any optional element in the ePlan Protocol Schema can be omitted in the Jurisdictional CIF Schema. Elements cannot be introduced into the Jurisdictional CIF Schema if they do not already exist in the ePlan Protocol Schema.

The cardinality of elements can only be modified to a higher restriction level in the Jurisdictional CIF Schema. If an element has a cardinality of 0 - \* in the ePlan Protocol Schema, it can be modified to 1 - \* in the Jurisdictional CIF Schema. However, a cardinality can not be modified from 1 - \* down to 0 - \* as this would allow the exclusion of an element that is mandatory in the ePlan Protocol Schema.

Attribute usage follows the same principles as cardinality. An "optional" attribute can be made "required" in the Jurisdictional CIF Schema but a "required" attribute cannot be made "optional"

### F.1 Namespace

The namespace of this schema should take the form of:

urn:xml-gov-au:<juris>:icsm:eplan:cif:protocol:<version>

<juris> is replaced with the jurisdiction identifier e.g. "qld" or "vic". <version> is replaced with the version number of this schema.

## Appendix G File Naming

There are the following distinct groups of schemas and reference files:

1. LandXML—the LandXML schema that provides the basis for ePlan schema
2. ICSM—the various ICSM schemas that are the basis of jurisdictional schemas
3. Jurisdictional—the jurisdictional schemas and data files

Files have been named according to the *Recommended XML Namespace for Government Organizations* (ref.3).

The naming convention for jurisdictional schema files is:

```
xml-gov-au-<jurisdiction>-icsm-eplan-cif-<fileType>-<version>.xsd
```

where:

- jurisdiction is the standard abbreviation for the jurisdiction (e.g. SA for South Australia)
- fileType is the type of file (e.g. annotations)
- version is a version number like 1.0 or a date stamp like 20101225. It is a jurisdictional decision on the style of version number.

For example:

National enumerations:

```
xml-gov-au-icsm-eplan-cif-enumerations-2.0.xsd
```

Jurisdictional enumerations:

```
xml-gov-au-qld-icsm-eplan-cif-enumerations-3.6.xsd  
xml-gov-au-qld-icsm-eplan-cif-enumerations-20101225.xsd
```

Note the file name should match the namespace of the schema.

Reference data files are named after the schema file they are based on, but with an appended version number (e.g. datestamp) specific to that stream of reference data files. E.g.

Reference data file:

```
xml-gov-au-qld-icsm-eplan-cif-annotations-1.0-20101225.xml
```

is based on Schema file:

```
xml-gov-au-icsm-eplan-cif-annotations-1.0.xsd
```

Note that all reference data file names in an RDC file must be fully qualified URIs.