

SGE and LGE media feed

[REDACTED]

[REDACTED]

[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]







[REDACTED]

[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED] [REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED] [REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED] [REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED] [REDACTED] [REDACTED] [REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]

[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]
[REDACTED]	[REDACTED]	[REDACTED]	[REDACTED]



Contents

	
	
	
Introduction	5
How and when to download files	5
When is the media feed available	5
How frequently is the media feed updated	5
From where can the media feed be downloaded	5
Description of file series and filenames	5
Zero files	5
SGE results files	6
LGE results files	7
By-election results files	7
Schema files	7
State LC Results XSD File	7
State LA Results XSD File	7
Local Government Results XSD File	7
SGE XML Schema Definition	8
Legislative Assembly	9
Legislative Council	41
LGE XML Schema Definition	68

Introduction

The Media Feed is made up of XML-based files that are progressively generated on election night. The results data extracted from NSWEC's Election Management Application (EMA) system database to a specified location at specified intervals.

The Media Feed is provided for both State and Local Government General Election events.

For State General Election (SGE) events, both Legislative Assembly and Legislative Council progressive result extracts are provided.

For Local Government Election (LGE) events, both Mayor and Councillor Contest progressive result extracts are provided.

The purpose of this document is to provide media agencies the following information:

- How to download the files, and what files are made available
- The detailed structure and contents of the files.

How and when to download files

When is the media feed available

The live media feed is usually available from approximately 7pm on election night and is updated continuously throughout the night. Zero files are generally made available before election night for reference purposes (see below).

How frequently is the media feed updated

The media feed extract files are produced at 5-minute intervals for an election event. During each 5 minute interval, if results in the EMA database have changed, new extract files will be generated and placed on the FTP site for download. However, if no change of in results data is recorded in a 5 minute interval, then no extract file would be produced.

From where can the media feed be downloaded

The media feed extract files are made available from NSWEC's access-restricted FTP site at <ftpsecure.elections.nsw.gov.au>. Separate step-by-step instructions are available for arranging access and connecting to this site.

Description of file series and filenames

Zero files

Zero files are generally published by NSWEC for LA, LC and LG events before election day. These files do not contain results data, but only the reference data for the election event (Contests, Venues, Candidates and Groups). These are only produced by NSWEC after that reference data becomes available within the EMA database. They have the same format as the corresponding LA, LC or LG results files, only the zero files do not contain vote count result. The zero files are identified as follows:

- LA Zero file, e.g., in SGE 2019 elections has the filename "00001-SG1901-LA.xml.zip"
- LC Zero file, e.g., in SGE 2019 elections has the filename "00001-SG1901-LC.xml.zip"
- LG Zero files, these are the first XML file generated for each of the LG areas. In LGE 2021, the zero files for some of the areas are:
 - 00001-Albury.xml
 - 00002-Armidale.xml
 - 00003-Ballina.xml

SGE results files

For SGEs, the following two file series are published, each file in the series containing election results at a particular point in time:

- Legislative Assembly (LA) result extract files
- Legislative Council (LC) result extract files

Legislative Assembly (LA) Media Feed File Extracts

LA Results Extract files are placed in a sub-folder named 'LA' on the FTP server. LA results for all State districts at a point in time are stored in a single XML file. The latest result extract file is also placed in a subfolder named 'Current' within the LA folder.

The 'Current' folder only ever contains the most recent result extract file, superseded files are removed from this folder but they remain available from the parent 'LA' folder.

The file name is structured as follows:

<extractsequencenumber>-<electioneventcode>-LA.xml.zip

Where

<extractsequencenumber> commencing from 000001

<electioneventcode> is the general election event code,

e.g., SG1901 for the March 2019 State General Election, example LA extract filenames:

000001-SG1901-LA.xml.zip

000002-SG1901-LA.xml.zip

Legislative Council (LC) Media Feed File Extracts

LC Results Extract files are placed in a sub-folder named 'LC' on the FTP server. All LC results at a point in time are stored in a single XML file. The latest result extract file is also placed inside a subfolder named 'Current' within the LC folder.

The 'Current' folder only ever contains the most recent result extract file, superseded files are removed from this folder but they remain available from the parent 'LC' folder.

The file name is structured as follows:

<extractsequencenumber>-<electioneventcode>-LC.xml.zip

Where

<extractsequencenumber> commencing from 000001

<electioneventcode> is the general election event code,

e.g., SG1901 for the March 2019 State General Election, example LC extract filenames:

000001-SG1901-LC.xml.zip

000002-SG1901-LC.xml.zip

LGE results files

For LGEs, the one file series contains both the Mayor and Councillor Contests results for all council areas, with each file in the series containing results for only one council area, and at a particular point in time.

LG Results Extract files are placed in a folder named after the LG election event ID (e.g., LG1701) on the FTP server. Results for each council area are stored in separate XML files. LG by-election result extracts (for the areas covered by the by-election) are identical to LG General Election result extracts. The most recent results file for each area is also placed inside a subfolder named 'Current' inside the LG election event folder.

The 'Current' folder only ever contains the most recent result extract files for each council area, superseded files are removed from this folder, but they remain available from the parent LG election event folder.

The file name is structured as follows:

<extractsequencenumber>-<councilareacode>.xml

Where

<extractsequencenumber> commencing from 000001

<councilareacode> is the council area code

Example LG extract filenames:

000001-Burwood.xml

000002-Ashfield.xml

By-election results files

By-election results files follow the same format as for the relevant type of general election.

Schema files

State LC Results XSD File



State_LC_Results_2021_V2.xsd

State LA Results XSD File



State_LA_Results_2021_V2.xsd

Local Government Results XSD File



LGE_Results_2021_V1.xsd

SGE XML Schema Definition

The Media Feed Extracts for SGE use 2 separate schema files, one for Legislative Assembly (LA) and another for Legislative Council (LC) XML file extracts. (Refer to sections 1.3.1 Legislative Assembly (LA) Media Feed File Extracts and 1.3.2 Legislative Council (LC) Media Feed File Extracts for the file names.)

Notes:

1. This document does not provide rules for validating the data types within an XML element.
2. Election Identifier "Id" attribute values:
 - LC = Legislative Council
 - LA = Legislative Assembly
3. Election Category values
 - Legislative Council (Senate element)
 - Legislative Assembly (House element)
4. Group Name element will not be populated for groups that are not endorsed by registered political parties
5. The LC group votes reported in these extracts are only for the First Preference above the line (ATL) votes. The votes for a Group without Group Voting Square are included in the Other votes.
6. Votes received by Ungrouped candidates is not reported, however, this are reported under the Other Votes.
7. The Legislative Council will have Informal votes included with the other votes and blank votes are also reported with other votes.
8. The number of votes to meet Quota will not be provided.
9. The results media feed data will not report TicketVotes since Tickets are not used in NSW
10. The election system will provide data from manual counts only during the Initial Count election phases. The check count data are provided from the VTR only using html reports.
 - ElectionNight = reports votes counted on Election Night
 - CountContinuing = reports votes counted after Election Night
 - InitialCountComplete = reports that all LA and LC First Preference votes have been counted and LA TCP, prior to check count.
11. Two candidates Preferred are provided for Polling Places on election night and for declaration vote types for close districts on the discretion of the NSW EC post-election night.
12. Vote Types and Venues.

Every Venue name is held in the Polling Place element. Each Polling Place element is given a Type which is an Attribute of the Polling Place element. The Types are;

 - PP – Polling Place Ordinary votes
 - PR – Pre-poll Ordinary votes
 - DV – Declaration Vote

The Polling Place element is the Venue name for all PP Types.

PR Types will use the venue name for each pre-poll where ordinary votes are taken. However, one venue in each district are called Declared Institution. This venue will hold all the ordinary votes for all the declared institutions in the district.

DV Types will use only the following Polling Place names in each district.

- Absent – votes taken outside the enrolled district
- Enrolment – votes taken for electors not enrolled in the district they are voting
- iVote – voted using the iVote system (if applicable)
- Postal – voted using a postal vote
- Enrolment/Provisional – either an Enrolment or Name Already Marked as Voted (NAMAV).

Legislative Assembly

Element MediaFeed

diagram						
properties	content complex					
children	EventIdentifier Election					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Verbose	xs:boolean	required			
	Id	xs:string	required			
	Created	xs:dateTime	required			
	Phase	xs:string	required			

attribute MediaFeed/@Verbose

type	xs:boolean
properties	use required
source	<xs:attribute name="Verbose" type="xs:boolean" use="required"/>

attribute **MediaFeed/@Id**

type	xs:string
properties	use required
source	<code><xs:attribute name="Id" type="xs:string" use="required"/></code>

attribute **MediaFeed/@Created**

type	xs:dateTime
properties	use required
source	<code><xs:attribute name="Created" type="xs:dateTime" use="required"/></code>

attribute **MediaFeed/@Phase**

type	xs:string
properties	use required
source	<code><xs:attribute name="Phase" type="xs:string" use="required"/></code>

Element **MediaFeed/EventIdentifier**

diagram													
properties	content complex												
children	EventName												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Id</td> <td>xs:unsignedShort</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Id	xs:unsignedShort	required			
Name	Type	Use	Default	Fixed	Annotation								
Id	xs:unsignedShort	required											
source	<pre> <xs:element name="EventIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="EventName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> </pre>												

attribute **MediaFeed/EventIdentifier/@Id**

type	xs:unsignedShort
properties	use required
source	<code><xs:attribute name="Id" type="xs:string" use="required"/></code>

Element MediaFeed/EventIdentifier/EventName

diagram	
type	xs:string
properties	content simple
source	<code><xs:element name="EventName" type="xs:string"/></code>

Element MediaFeed/Election

diagram	
properties	content complex
children	ElectionIdentifier House


Element MediaFeed/Election/ElectionIdentifier

diagram													
properties	content complex												
children	ElectionName ElectionCategory												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Id</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Id	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Id	xs:string	required											
source	<pre> <xs:element name="ElectionIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="ElectionName" type="xs:string"/> <xs:element name="ElectionCategory" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> </pre>												


attribute MediaFeed/Election/ElectionIdentifier/@Id

type	xs:string
properties	use required
source	<code><xs:attribute name="Id" type="xs:string" use="required"/></code>

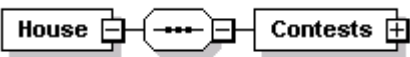
Element MediaFeed/Election/ElectionIdentifier/ElectionName

diagram	
type	xs:string
properties	content simple
source	<code><xs:element name="ElectionName" type="xs:string"/></code>

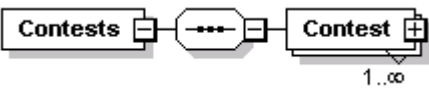
Element MediaFeed/Election/ElectionIdentifier/ElectionCategory

diagram	
type	xs:string
properties	content simple
source	<code><xs:element name="ElectionCategory" type="xs:string"/></code>

Element House

diagram	
properties	content complex
children	Contests

Element House/Contests

diagram	
properties	content complex
children	Contest

Element Contest

diagram						
properties	minOcc	1				
	maxOcc	unbounded				
	content	complex				
children	ContestIdentifier PollingDistrictIdentifier Declared NumberOfPositions Enrolment FirstPreferences TwoCandidatePreferred PollingPlaces					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Updated	xs:dateTime	required			
	Status	xs:string	optional			

attribute House/Contests/Contest/@Updated

type	xs:dateTime
properties	use required
source	<code><xs:attribute name="Updated" type="xs:dateTime" use="required"/></code>

attribute House/Contests/Contest/@Status

type	xs:string
properties	use optional
source	<code><xs:attribute name="Updated" type="xs:string" use="required"/></code>

Element Contest/ContestIdentifier

diagram						
properties	content complex					
children	ContestName					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Id	xs:unsignedShort	required			
source	<pre> <xs:element name="ContestIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="ContestName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> </pre>					

attribute House/Contests/Contest/ContestIdentifier/@Id

type	xs:unsignedShort
properties	use required
source	<pre><xs:attribute name="Id" type="xs:string" use="required"/></pre>

Element Contest/ContestIdentifier/ContestName

diagram						
type	xs:string					
properties	content simple					
source	<pre> <xs:element name="ContestName" type="xs:string"/> </pre>					

Element Contest/PollingDistrictIdentifier

diagram						
properties	content complex					
children	Name					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Id	xs:unsignedShort	required			
source	<pre> <xs:element name="PollingDistrictIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="Name" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> </pre>					

attribute House/Contests/Contest/PollingDistrictIdentifier/@Id

type	xs:unsignedShort
properties	use required
source	<pre><xs:attribute name="Id" type="xs:string" use="required"/></pre>

Element Contest/PollingDistrictIdentifier/Name

diagram						
type	xs:string					
properties	content simple					
source	<pre><xs:element name="Name" type="xs:string"/></pre>					

Element Contest/Declared

diagram						
type	xs:boolean					
properties	content simple					
source	<pre><xs:element name="Declared" type="xs:boolean"/></pre>					

Element Contest/NumberOfPositions

diagram	
type	xs:unsignedByte
properties	content simple
source	<code><xs:element name="NumberOfPositions" type="xs:unsignedByte"/></code>

Element Contest/Enrolment

diagram	
type	xs:unsignedShort
properties	content simple
source	<code><xs:element name="Enrolment" type="xs:unsignedShort"/></code>

Element Contest/FirstPreferences

diagram						
properties	content complex					
children	Candidate Formal Informal Total					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Updated	xs:dateTime	required			
	PollingPlacesReturned	xs:unsignedByte	required			
	PollingPlacesExpected	xs:unsignedByte	required			

attribute **House/Contests/Contest/FirstPreferences/@Updated**

type	xs:dateTime
properties	use required
source	<code><xs:attribute name="Updated" type="xs:dateTime" use="required"/></code>

attribute **House/Contests/Contest/FirstPreferences/@PollingPlacesReturned**

type	xs:unsignedByte
properties	use required
source	<code><xs:attribute name="PollingPlacesReturned" type="xs:unsignedByte" use="required"/></code>

attribute **House/Contests/Contest/FirstPreferences/@PollingPlacesExpected**

type	xs:unsignedByte
properties	use required
source	<code><xs:attribute name="PollingPlacesExpected" type="xs:unsignedByte" use="required"/></code>

Element Contest/FirstPreferences/Candidate

diagram													
properties	<p>minOcc 1</p> <p>maxOcc unbounded</p> <p>content complex</p>												
children	CandidateIdentifier Affiliation Elected BallotPosition Votes VotesByType												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Independent</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Independent	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Independent	xs:string	required											

attribute **House/Contests/Contest/FirstPreferences/Candidate/@Independent**

type	xs:string
properties	use required
source	<code><xs:attribute name="Independent" type="xs:string" use="required"/></code>

Element **Contest/FirstPreferences/Candidate/CandidateIdentifier**

diagram						
properties	content complex					
children	CandidateName					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Id	xs:unsignedShort	required			
source	<pre> <xs:element name="CandidateIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="CandidateName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> </pre>					

attribute **House/Contests/Contest/FirstPreferences/Candidate/CandidateIdentifier/@Id**

type	xs:unsignedShort
properties	use required
source	<code><xs:attribute name="Id" type="xs:string" use="required"/></code>

Element **Contest/FirstPreferences/Candidate/CandidateIdentifier/CandidateName**

diagram						
type	xs:string					
properties	content simple					
source	<code><xs:element name="CandidateName" type="xs:string"/></code>					

Element House/Contests/Contest/FirstPreferences/Candidate/Affiliation

diagram	
properties	<p>minOcc 0</p> <p>maxOcc 1</p> <p>content complex</p>
children	AffiliationIdentifier
source	<pre><xs:element name="Affiliation" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="AffiliationIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="RegisteredName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:unsignedByte" use="required"/> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

Element Contest/FirstPreferences/Candidate/Affiliation/AffiliationIdentifier

diagram																			
properties	content complex																		
children	RegisteredName																		
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Id</td> <td>xs:unsignedInt</td> <td>optional</td> <td></td> <td></td> <td></td> </tr> <tr> <td>ShortCode</td> <td>xs:string</td> <td>optional</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Id	xs:unsignedInt	optional				ShortCode	xs:string	optional			
Name	Type	Use	Default	Fixed	Annotation														
Id	xs:unsignedInt	optional																	
ShortCode	xs:string	optional																	
source	<pre><xs:element name="AffiliationIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="RegisteredName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:unsignedInt" use="optional"/> <xs:attribute name="ShortCode" type="xs:string" use="optional"/> </xs:complexType> </xs:element></pre>																		

attribute **House/Contests/Contest/FirstPreferences/Candidate/Affiliation/AffiliationIdentifier/@Id**

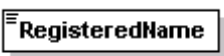
type	xs:unsignedInt The attribute provides the numeric id for a party
properties	use required
source	<code><xs:attribute name="Id" type="xs: unsignedInt " use=" optional "/></code>

attribute


House/Contests/Contest/FirstPreferences/Candidate/Affiliation/AffiliationIdentifier/@ShortCode

type	xs: string , The attribute provides the abbreviated party code
properties	use optional
source	<code><xs:attribute name="Id" type="xs: string " use=" optional "/></code>

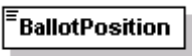
Element Contest/FirstPreferences/Candidate/Affiliation/AffiliationIdentifier/RegisteredName

diagram	
type	xs:string
properties	content simple
source	<code><xs:element name="RegisteredName" type="xs:string"/></code>

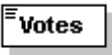
Element Contest/FirstPreferences/Candidate/Elected

diagram	
type	xs:boolean
properties	content simple
source	<code><xs:element name="Elected" type="xs:boolean"/></code>

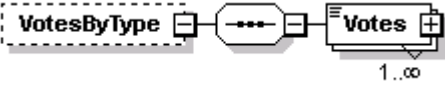
Element Contest/FirstPreferences/Candidate/BallotPosition

diagram	
type	xs:unsignedByte
properties	content simple
source	<code><xs:element name="BallotPosition" type="xs:unsignedByte"/></code>

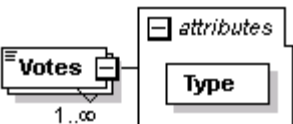
Element Contest/FirstPreferences/Candidate/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<code><xs:element name="Votes" type="xs:unsignedShort"/></code>

Element Contest/FirstPreferences/Candidate/VotesByType

diagram	
properties	minOcc 0 maxOcc 1 content complex
children	Votes
source	<pre><xs:element name="VotesByType" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

Element Contest/FirstPreferences/Candidate/VotesByType/Votes

diagram	
type	extension of xs:unsignedShort
properties	minOcc 1 maxOcc unbounded content complex

attributes	Name	Type	Use	Default	Fixed	Annotation
	Type	xs:string	required			
source	<pre><xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element></pre>					

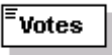
attribute **House/Contests/Contest/FirstPreferences/Candidate/VotesByType/Votes/@Type**

type	xs:string
properties	use required
source	<pre><xs:attribute name="Type" type="xs:string" use="required"/></pre>

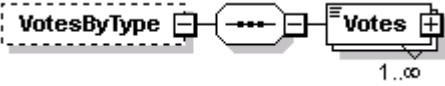
Element Contest/FirstPreferences/Formal

diagram	
properties	content complex
children	Votes VotesByType
source	<pre><xs:element name="Formal"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> <xs:element name="VotesByType" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

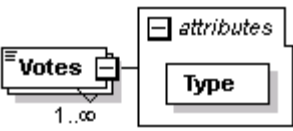
Element Contest/FirstPreferences/Formal/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<code><xs:element name="Votes" type="xs:unsignedShort"/></code>

Element Contest/FirstPreferences/Formal/VotesByType

diagram	
properties	<p>minOcc 0</p> <p>maxOcc 1</p> <p>content complex</p>
children	Votes
source	<pre> <xs:element name="VotesByType" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element Contest/FirstPreferences/Formal/VotesByType/Votes

diagram	
type	extension of xs:unsignedShort
properties	<p>minOcc 1</p> <p>maxOcc unbounded</p> <p>content complex</p>

attributes	Name	Type	Use	Default	Fixed	Annotation
	Type	xs:string	required			
source	<pre><xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element></pre>					

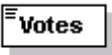
attribute **House/Contests/Contest/FirstPreferences/Formal/VotesByType/Votes/@Type**

type	xs:string
properties	use required
source	<pre><xs:attribute name="Type" type="xs:string" use="required"/></pre>

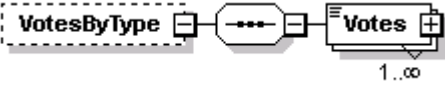
Element Contest/FirstPreferences/Informal

diagram	
properties	content complex
children	Votes VotesByType
source	<pre><xs:element name="Informal"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> <xs:element name="VotesByType" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

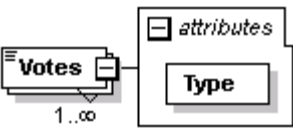
Element House/Contests/Contest/FirstPreferences/Informal/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<code><xs:element name="Votes" type="xs:unsignedShort"/></code>

Element Contest/FirstPreferences/Informal/VotesByType

diagram	
properties	<p>minOcc 0</p> <p>maxOcc 1</p> <p>content complex</p>
children	Votes
source	<pre> <xs:element name="VotesByType" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element Contest/FirstPreferences/Informal/VotesByType/Votes

diagram	
type	extension of xs:unsignedShort
properties	<p>minOcc 1</p> <p>maxOcc unbounded</p> <p>content complex</p>

attributes	Name	Type	Use	Default	Fixed	Annotation
	Type	xs:string	required			
source	<pre><xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element></pre>					

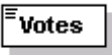
attribute **House/Contests/Contest/FirstPreferences/Informal/VotesByType/Votes/@Type**

type	xs:string
properties	use required
source	<pre><xs:attribute name="Type" type="xs:string" use="required"/></pre>

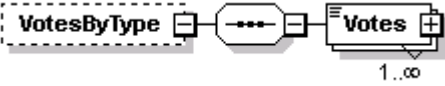
Element Contest/FirstPreferences/Total

diagram	
properties	content complex
children	Votes VotesByType
source	<pre><xs:element name="Total"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> <xs:element name="VotesByType" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

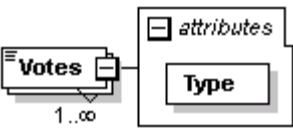
Element Contest/FirstPreferences/Total/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<code><xs:element name="Votes" type="xs:unsignedShort"/></code>

Element Contest/FirstPreferences/Total/VotesByType

diagram	
properties	minOcc 0 maxOcc 1 content complex
children	Votes
source	<pre><xs:element name="VotesByType" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

Element Contest/FirstPreferences/Total/VotesByType/Votes

diagram													
type	extension of xs:unsignedShort												
properties	minOcc 1 maxOcc unbounded content complex												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Type</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Type	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Type	xs:string	required											

source	<pre> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </pre>
--------	---

attribute **House/Contests/Contest/FirstPreferences/Total/VotesByType/Votes/@Type**

type	xs:string
properties	use required
source	<pre><xs:attribute name="Type" type="xs:string" use="required"/></pre>

Element House/Contests/Contest/TwoCandidatePreferred

diagram	<pre> classDiagram class TwoCandidatePreferred { +PollingPlacesReturned +PollingPlacesExpected +Updated } class Candidate TwoCandidatePreferred "1" -- "1..∞" Candidate </pre>					
properties	content complex					
children	Candidate					
attributes	Name	Type	Use	Default	Fixed	Annotation
	PollingPlacesReturned	xs:unsignedByte	required			
	PollingPlacesExpected	xs:unsignedByte	required			
	Updated	xs:dateTime	required			

source	<pre> <xs:element name="TwoCandidatePreferred"> <xs:complexType> <xs:sequence> <xs:element name="Candidate" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="CandidateIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="CandidateName" type="xs:string"/> </xs:sequence> </xs:complexType> </xs:element> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="Votes" type="xs:unsignedShort"/> <xs:element name="VotesByType" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> <xs:attribute name="PollingPlacesReturned" type="xs:unsignedByte" use="required"/> <xs:attribute name="PollingPlacesExpected" type="xs:unsignedByte" use="required"/> <xs:attribute name="Updated" type="xs:dateTime" use="required"/> </xs:complexType> </xs:element> </pre>
--------	---

attribute **House/Contests/Contest/TwoCandidatePreferred/@PollingPlacesReturned**

type	xs:unsignedByte
properties	use required
source	<xs:attribute name="PollingPlacesReturned" type="xs:unsignedByte" use="required"/>

attribute **House/Contests/Contest/TwoCandidatePreferred/@PollingPlacesExpected**

type	xs:unsignedByte
properties	use required
source	<xs:attribute name="PollingPlacesExpected" type="xs:unsignedByte" use="required"/>

attribute **House/Contests/Contest/TwoCandidatePreferred/@Updated**

type	xs:dateTime
properties	use required
source	<code><xs:attribute name="Updated" type="xs:dateTime" use="required"/></code>

Element Contest/TwoCandidatePreferred/Candidate

diagram	
properties	<p>minOcc 1</p> <p>maxOcc unbounded</p> <p>content complex</p>
children	CandidateIdentifier Votes VotesByType
source	<pre> <xs:element name="Candidate" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="CandidateIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="CandidateName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> <xs:element name="Votes" type="xs:unsignedShort"/> <xs:element name="VotesByType" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element Contest/TwoCandidatePreferred/Candidate/CandidateIdentifier

diagram						
properties	content complex					
children	CandidateName					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Id	xs:unsignedShort	required			
source	<pre><xs:element name="CandidateIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="CandidateName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element></pre>					

attribute House/Contests/Contest/TwoCandidatePreferred/Candidate/CandidateIdentifier/@Id

type	xs:unsignedShort
properties	use required
source	<pre><xs:attribute name="Id" type="xs:string" use="required"/></pre>

Element Contest/TwoCandidatePreferred/Candidate/CandidateIdentifier/CandidateName

diagram						
type	xs:string					
properties	content simple					
source	<pre><xs:element name="CandidateName" type="xs:string"/></pre>					

Element House/Contests/Contest/TwoCandidatePreferred/Candidate/Votes

diagram						
type	xs:unsignedShort					
properties	content simple					
source	<pre><xs:element name="Votes" type="xs:unsignedShort"/></pre>					

Element Contest/TwoCandidatePreferred/Candidate/VotesByType

diagram	<p>The diagram shows a container box labeled 'VotesByType' with a dashed border. Inside it is a sequence of three boxes, each labeled 'Votes'. The 'Votes' box has a '1..∞' cardinality indicator below it.</p>
properties	<p>minOcc 0</p> <p>maxOcc 1</p> <p>content complex</p>
children	<p>Votes</p>
source	<pre><xs:element name="VotesByType" minOccurs="0"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

Element Contest/TwoCandidatePreferred/Candidate/VotesByType/Votes

diagram	<p>The diagram shows a box labeled 'Votes' with a '1..∞' cardinality indicator below it. To its right is a box labeled 'attributes' containing a box labeled 'Type'.</p>												
type	<p>extension of xs:unsignedShort</p>												
properties	<p>minOcc 1</p> <p>maxOcc unbounded</p> <p>content complex</p>												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Type</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Type	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Type	xs:string	required											
source	<pre><xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element></pre>												

attribute **House/Contests/Contest/TwoCandidatePreferred/Candidate/VotesByType/Votes/@Type**

type	xs:string
properties	use required
source	<code><xs:attribute name="Type" type="xs:string" use="required"/></code>

Element House/Contests/Contest/PollingPlaces

diagram	
properties	minOcc 0 maxOcc 1 content complex
children	PollingPlace

Element House/Contests/Contest/PollingPlaces/PollingPlace

diagram													
properties	minOcc 1 maxOcc unbounded content complex												
children	PollingPlacelIdentifier FirstPreferences TwoCandidatePreferred												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Updated</td> <td>xs:dateTime</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Updated	xs:dateTime	required			
Name	Type	Use	Default	Fixed	Annotation								
Updated	xs:dateTime	required											

attribute **House/Contests/Contest/PollingPlaces/PollingPlace/@Updated**

type	xs:dateTime
properties	use required
source	<code><xs:attribute name="Updated" type="xs:dateTime" use="required"/></code>

Element Contest/PollingPlaces/PollingPlace/PollingPlacelIdentifier

diagram						
properties	content	complex				
attributes	Name	Type	Use	Default	Fixed	Annotation
	Id	xs:unsignedShort	required			
	Name	xs:string	required			
source	<pre><xs:element name="PollingPlacelIdentifier"> <xs:complexType> <xs:attribute name="Id" type="xs:string" use="required"/> <xs:attribute name="Name" type="xs:string" use="required"/> </xs:complexType> </xs:element></pre>					

attribute House/Contests/Contest/PollingPlaces/PollingPlace/PollingPlacelIdentifier/@Id

type	xs:unsignedShort
properties	use required
source	<pre><xs:attribute name="Id" type="xs:string" use="required"/></pre>

attribute House/Contests/Contest/PollingPlaces/PollingPlace/PollingPlacelIdentifier/@Name

type	xs:string
properties	use required
source	<pre><xs:attribute name="Name" type="xs:string" use="required"/></pre>

Element Contest/PollingPlaces/PollingPlace/FirstPreferences

diagram						
---------	--	--	--	--	--	--

properties	content complex												
children	Candidate Formal Informal Total												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Updated</td> <td>xs:dateTime</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Updated	xs:dateTime	required			
Name	Type	Use	Default	Fixed	Annotation								
Updated	xs:dateTime	required											
source	<pre> <xs:element name="FirstPreferences"> <xs:complexType> <xs:sequence> <xs:element name="Candidate" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="CandidateIdentifier"> <xs:complexType> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="Formal"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="Informal"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="Total"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> <xs:attribute name="Updated" type="xs:dateTime" use="required"/> </xs:complexType> </xs:element> </pre>												

attribute **House/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/@Updated**

type	xs:dateTime
properties	use required
source	<pre><xs:attribute name="Updated" type="xs:dateTime" use="required"/></pre>

Element Contest/PollingPlaces/PollingPlace/FirstPreferences/Candidate

diagram	
properties	<p>minOcc 1</p> <p>maxOcc unbounded</p> <p>content complex</p>
children	<p>CandidateIdentifier Votes</p>
source	<pre><xs:element name="Candidate" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="CandidateIdentifier"> <xs:complexType> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element></pre>

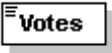
Element Contest/PollingPlaces/PollingPlace/FirstPreferences/Candidate/CandidateIdentifier

diagram													
properties	<p>content complex</p>												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Id</td> <td>xs:unsignedShort</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Id	xs:unsignedShort	required			
Name	Type	Use	Default	Fixed	Annotation								
Id	xs:unsignedShort	required											
source	<pre><xs:element name="CandidateIdentifier"> <xs:complexType> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element></pre>												


attribute House/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Candidate/CandidateIdentifier/@Id

type	xs:unsignedShort
properties	use required
source	<xs:attribute name="Id" type="xs:string" use="required"/>

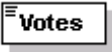
Element Contest/PollingPlaces/PollingPlace/FirstPreferences/Candidate/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<code><xs:element name="Votes" type="xs:unsignedShort"/></code>

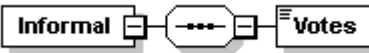
Element Contest/PollingPlaces/PollingPlace/FirstPreferences/Formal

diagram	
properties	content complex
children	Votes
source	<code><xs:element name="Formal"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element></code>

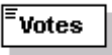
Element Contest/PollingPlaces/PollingPlace/FirstPreferences/Formal/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<code><xs:element name="Votes" type="xs:unsignedShort"/></code>


Element Contest/PollingPlaces/PollingPlace/FirstPreferences/Informal

diagram	
properties	content complex
children	Votes
source	<code><xs:element name="Informal"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element></code>

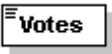
Element Contest/PollingPlaces/PollingPlace/FirstPreferences/Informal/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<xs:element name="Votes" type="xs:unsignedShort"/>

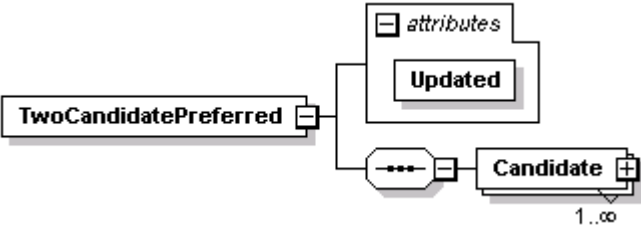
Element Contest/PollingPlaces/PollingPlace/FirstPreferences/Total

diagram	
properties	content complex
children	Votes
source	<xs:element name="Total"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element>

Element Contest/PollingPlaces/PollingPlace/FirstPreferences/Total/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<xs:element name="Votes" type="xs:unsignedShort"/>

Element Contest/PollingPlaces/PollingPlace/TwoCandidatePreferred

diagram													
properties	content complex												
children	Candidate												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Updated</td> <td>xs:dateTime</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Updated	xs:dateTime	required			
Name	Type	Use	Default	Fixed	Annotation								
Updated	xs:dateTime	required											

source	<pre> <xs:element name="TwoCandidatePreferred"> <xs:complexType> <xs:sequence> <xs:element name="Candidate" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="CandidateIdentifier"> <xs:complexType> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> <xs:attribute name="Updated" type="xs:dateTime" use="required"/> </xs:complexType> </xs:element> </pre>
--------	---

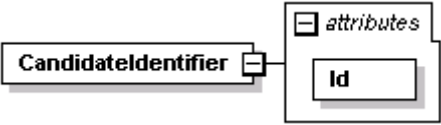
attribute **House/Contests/Contest/PollingPlaces/PollingPlace/TwoCandidatePreferred/@Updated**

type	xs:dateTime
properties	use required
source	<pre><xs:attribute name="Updated" type="xs:dateTime" use="required"/></pre>

Element Contest/PollingPlaces/PollingPlace/TwoCandidatePreferred/Candidate

diagram	
properties	<p>minOcc 1</p> <p>maxOcc unbounded</p> <p>content complex</p>
children	CandidateIdentifier Votes
source	<pre> <xs:element name="Candidate" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="CandidateIdentifier"> <xs:complexType> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

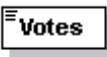
Element Contest/PollingPlaces/PollingPlace/TwoCandidatePreferred/Candidate/CandidateIdentifier

diagram													
properties	content complex												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Id</td> <td>xs:unsignedShort</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Id	xs:unsignedShort	required			
Name	Type	Use	Default	Fixed	Annotation								
Id	xs:unsignedShort	required											
source	<pre><xs:element name="CandidateIdentifier"> <xs:complexType> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element></pre>												

attribute House/Contests/Contest/PollingPlaces/PollingPlace/TwoCandidatePreferred/Candidate/CandidateIdentifier/@Id

type	xs:unsignedShort
properties	use required
source	<pre><xs:attribute name="Id" type="xs:string" use="required"/></pre>

Element Contest/PollingPlaces/PollingPlace/TwoCandidatePreferred/Candidate/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<pre><xs:element name="Votes" type="xs:unsignedShort"/></pre>

Legislative Council

Element MediaFeed

diagram	<pre> classDiagram class MediaFeed { Verbose Id Created Phase EventIdentifier Election } </pre>																														
properties	content complex																														
children	EventIdentifier Election																														
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Verbose</td> <td>xs:boolean</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Id</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Created</td> <td>xs:dateTime</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Phase</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Verbose	xs:boolean	required				Id	xs:string	required				Created	xs:dateTime	required				Phase	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation																										
Verbose	xs:boolean	required																													
Id	xs:string	required																													
Created	xs:dateTime	required																													
Phase	xs:string	required																													

attribute MediaFeed/@Verbose

type	xs:boolean
properties	use required
source	<code><xs:attribute name="Verbose" type="xs:boolean" use="required"/></code>

attribute MediaFeed/@Id

type	xs:string
properties	use required
source	<code><xs:attribute name="Id" type="xs:string" use="required"/></code>

attribute MediaFeed/@Created

type	xs:dateTime
properties	use required
source	<code><xs:attribute name="Created" type="xs:dateTime" use="required"/></code>

attribute **MediaFeed/@Phase**

type	xs:string
properties	use required
source	<code><xs:attribute name="Phase" type="xs:string" use="required"/></code>

Element MediaFeed/EventIdentifier

diagram						
properties	content complex					
children	EventName					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Id	xs:unsignedShort	required			
source	<pre> <xs:element name="EventIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="EventName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> </pre>					

attribute **MediaFeed/EventIdentifier/@Id**

type	xs:unsignedShort
properties	use required
source	<code><xs:attribute name="Id" type="xs:string" use="required"/></code>

Element MediaFeed/EventIdentifier/EventName

diagram						
type	xs:string					
properties	content simple					
source	<code><xs:element name="EventName" type="xs:string"/></code>					

Element MediaFeed/Election

diagram	
properties	content complex
children	ElectionIdentifier Senate

Element MediaFeed/Election/ElectionIdentifier

diagram													
properties	content complex												
children	ElectionName ElectionCategory												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Id</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Id	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Id	xs:string	required											
source	<pre><xs:element name="ElectionIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="ElectionName" type="xs:string"/> <xs:element name="ElectionCategory" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element></pre>												

attribute MediaFeed/Election/ElectionIdentifier/@Id

type	xs:string
properties	use required
source	<pre><xs:attribute name="Id" type="xs:string" use="required"/></pre>

Element MediaFeed/Election/ElectionIdentifier/ElectionName

diagram	
type	xs:string
properties	content simple
source	<pre><xs:element name="ElectionName" type="xs:string"/></pre>

Element MediaFeed/Election/ElectionIdentifier/ElectionCategory

diagram	
type	xs:string
properties	content simple
source	<xs:element name="ElectionCategory" type="xs:string"/>

Element Senate

diagram	
properties	content complex
children	Contests

Element Senate/Contests

diagram	
properties	content complex
children	Contest

Element Senate/Contests/Contest

diagram	
---------	--

properties	minOcc 1					
	maxOcc unbounded					
	content complex					
children	ContestIdentifier PollingDistrictIdentifier Declared NumberOfPositions Enrolment Quota FirstPreferences PollingPlaces					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Updated	xs:dateTime	required			

attribute **Senate/Contests/Contest/@Updated**

type	xs:dateTime
properties	use required
source	<xs:attribute name="Updated" type="xs:dateTime" use="required"/>


Element **Senate/Contests/Contest/ContestIdentifier**

diagram													
properties	content complex												
children	ContestName												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Id</td> <td>xs:unsignedShort</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Id	xs:unsignedShort	required			
Name	Type	Use	Default	Fixed	Annotation								
Id	xs:unsignedShort	required											
source	<pre><xs:element name="ContestIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="ContestName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element></pre>												

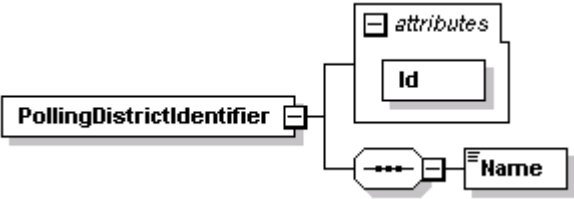
attribute **Senate/Contests/Contest/ContestIdentifier/@Id**

type	xs:unsignedShort
properties	use required
source	<xs:attribute name="Id" type="xs:string" use="required"/>

Element Senate/Contests/Contest/ContestIdentifier/ContestName

diagram	
type	xs:string
properties	content simple
source	<code><xs:element name="ContestName" type="xs:string"/></code>


Element Senate/Contests/Contest/PollingDistrictIdentifier

diagram													
properties	content complex												
children	Name												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Id</td> <td>xs:unsignedShort</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Id	xs:unsignedShort	required			
Name	Type	Use	Default	Fixed	Annotation								
Id	xs:unsignedShort	required											
source	<pre> <xs:element name="PollingDistrictIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="Name" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> </pre>												


attribute Senate/Contests/Contest/PollingDistrictIdentifier/@Id

type	xs:unsignedShort
properties	use required
source	<code><xs:attribute name="Id" type="xs:string" use="required"/></code>


Element Senate/Contests/Contest/PollingDistrictIdentifier/Name

diagram	
type	xs:string
properties	content simple
source	<code><xs:element name="Name" type="xs:string"/></code>


Element Senate/Contests/Contest/Declared

diagram	
type	xs:boolean
properties	content simple
source	<code><xs:element name="Declared" type="xs:boolean"/></code>

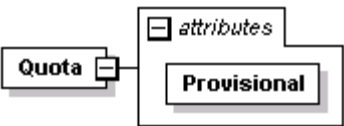
Element Senate/Contests/Contest/NumberOfPositions

diagram	
type	xs:unsignedByte
properties	content simple
source	<code><xs:element name="NumberOfPositions" type="xs:unsignedByte"/></code>

Element Senate/Contests/Contest/Enrolment

diagram	
type	xs:unsignedInt
properties	content simple
source	<code><xs:element name="Enrolment" type="xs:unsignedInt"/></code>

Element Senate/Contests/Contest/Quota

diagram													
properties	content complex												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Provisional</td> <td>xs:boolean</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Provisional	xs:boolean	required			
Name	Type	Use	Default	Fixed	Annotation								
Provisional	xs:boolean	required											
source	<pre> <xs:element name="Quota"> <xs:complexType> <xs:attribute name="Provisional" type="xs:boolean" use="required"/> </xs:complexType> </xs:element> </pre>												

attribute **Senate/Contests/Contest/Quota/@Provisional**

type	xs:boolean
properties	use required
source	<code><xs:attribute name="Provisional" type="xs:boolean" use="required"/></code>

Element **Senate/Contests/Contest/FirstPreferences**

diagram						
properties	content complex					
children	Group Formal Informal Total					
attributes	Name	Type	Use	Default	Fixed	Annotation
	PollingPlacesReturned	xs:unsignedShort	required			
	PollingPlacesExpected	xs:unsignedShort	required			
	Updated	xs:dateTime	required			

attribute **Senate/Contests/Contest/FirstPreferences/@PollingPlacesReturned**

type	xs:unsignedShort
properties	use required
source	<code><xs:attribute name="PollingPlacesReturned" type="xs:unsignedShort" use="required"/></code>

attribute **Senate/Contests/Contest/FirstPreferences/@PollingPlacesExpected**

type	xs:unsignedShort
properties	use required
source	<code><xs:attribute name="PollingPlacesExpected" type="xs:unsignedShort" use="required"/></code>

attribute **Senate/Contests/Contest/FirstPreferences/@Updated**

type	xs:dateTime
properties	use required
source	<code><xs:attribute name="Updated" type="xs:dateTime" use="required"/></code>

Element Senate/Contests/Contest/FirstPreferences/Group

diagram	
properties	<p>minOcc 1</p> <p>maxOcc unbounded</p> <p>content complex</p>
children	GroupIdentifier TicketVotes GroupVotes


Element Senate/Contests/Contest/FirstPreferences/Group/GroupIdentifier

diagram													
properties	content complex												
children	GroupName												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td><u>Id</u></td> <td>xs:unsignedShort</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	<u>Id</u>	xs:unsignedShort	required			
Name	Type	Use	Default	Fixed	Annotation								
<u>Id</u>	xs:unsignedShort	required											
source	<pre> <xs:element name="GroupIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="GroupName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> </pre>												

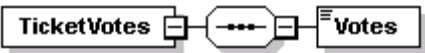
attribute **Senate/Contests/Contest/FirstPreferences/Group/GroupIdentifier/@Id**

type	xs:unsignedShort
properties	use required
source	<code><xs:attribute name="Id" type="xs:string" use="required"/></code>

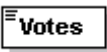
Element **Senate/Contests/Contest/FirstPreferences/Group/GroupIdentifier/GroupName**

diagram	
type	xs:string
properties	content simple
source	<code><xs:element name="GroupName" type="xs:string"/></code>

Element **Senate/Contests/Contest/FirstPreferences/Group/TicketVotes**

diagram	
properties	content complex
children	Votes
source	<code><xs:element name="TicketVotes"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedInt"/> </xs:sequence> </xs:complexType> </xs:element></code>

Element **Senate/Contests/Contest/FirstPreferences/Group/TicketVotes/Votes**

diagram	
type	xs:unsignedInt
properties	content simple
source	<code><xs:element name="Votes" type="xs:unsignedInt"/></code>

Element Senate/Contests/Contest/FirstPreferences/Group/GroupVotes

diagram	
properties	content complex
children	Votes VotesByType
source	<pre> <xs:element name="GroupVotes"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedInt"/> <xs:element name="VotesByType"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedInt"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element Senate/Contests/Contest/FirstPreferences/Group/GroupVotes/Votes

diagram	
type	xs:unsignedInt
properties	content simple
source	<pre> <xs:element name="Votes" type="xs:unsignedInt"/> </pre>

Element Senate/Contests/Contest/FirstPreferences/Group/GroupVotes/VotesByType

diagram	
properties	content complex
children	Votes

source	<pre> <xs:element name="VotesByType"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedInt"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--------	---

Element Senate/Contests/Contest/FirstPreferences/Group/GroupVotes/VotesByType/Votes

diagram													
type	extension of xs:unsignedInt												
properties	minOcc 1 maxOcc unbounded content complex												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Type</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Type	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Type	xs:string	required											
source	<pre> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedInt"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </pre>												

attribute Senate/Contests/Contest/FirstPreferences/Group/GroupVotes/VotesByType/Votes/@Type

type	xs:string
properties	use required
source	<pre><xs:attribute name="Type" type="xs:string" use="required"/></pre>

Element Senate/Contests/Contest/FirstPreferences/Formal

diagram	
properties	content complex
children	Votes VotesByType
source	<pre> <xs:element name="Formal"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedInt"/> <xs:element name="VotesByType"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedInt"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element Senate/Contests/Contest/FirstPreferences/Formal/Votes

diagram	
type	xs:unsignedInt
properties	content simple
source	<pre> <xs:element name="Votes" type="xs:unsignedInt"/> </pre>

Element Senate/Contests/Contest/FirstPreferences/Formal/VotesByType

diagram	
properties	content complex
children	Votes

source	<pre> <xs:element name="VotesByType"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedInt"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--------	---

Element Senate/Contests/Contest/FirstPreferences/Formal/VotesByType/Votes

diagram													
type	extension of xs:unsignedInt												
properties	minOcc 1 maxOcc unbounded content complex												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Type</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Type	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Type	xs:string	required											
source	<pre> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedInt"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </pre>												

attribute Senate/Contests/Contest/FirstPreferences/Formal/VotesByType/Votes/@Type

type	xs:string
properties	use required
source	<pre><xs:attribute name="Type" type="xs:string" use="required"/></pre>

Element Senate/Contests/Contest/FirstPreferences/Others

diagram	
properties	content complex
children	Votes VotesByType
source	<pre> <xs:element name="Others"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> <xs:element name="VotesByType"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element Senate/Contests/Contest/FirstPreferences/Others/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<pre> <xs:element name="Votes" type="xs:unsignedShort"/> </pre>

Element Senate/Contests/Contest/FirstPreferences/Others/VotesByType

diagram	
properties	content complex
children	Votes

source	<pre> <xs:element name="VotesByType"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--------	---

Element Senate/Contests/Contest/FirstPreferences/Others/VotesByType/Votes

diagram													
type	extension of xs:unsignedShort												
properties	minOcc 1 maxOcc unbounded content complex												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Type</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Type	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Type	xs:string	required											
source	<pre> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </pre>												

attribute MediaFeed/Election/Senate/Contests/Contest/FirstPreferences/Others/VotesByType/Votes/@Type

type	xs:string
properties	use required
source	<pre><xs:attribute name="Type" type="xs:string" use="required"/></pre>

Element Senate/Contests/Contest/FirstPreferences/Blank

diagram	
properties	content complex
children	Votes VotesByType
source	<pre> <xs:element name="Blank"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> <xs:element name="VotesByType"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element MediaFeed/Election/Senate/Contests/Contest/FirstPreferences/Blank/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<pre> <xs:element name="Votes" type="xs:unsignedShort"/> </pre>

Element MediaFeed/Election/Senate/Contests/Contest/FirstPreferences/Blank/VotesByType

diagram	
properties	content complex
children	Votes

source	<pre> <xs:element name="VotesByType"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--------	---

Element MediaFeed/Election/Senate/Contests/Contest/FirstPreferences/Blank/VotesByType/Votes

diagram													
type	extension of xs:unsignedShort												
properties	minOcc 1 maxOcc unbounded content complex												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Type</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Type	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Type	xs:string	required											
source	<pre> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedShort"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </pre>												

attribute MediaFeed/Election/Senate/Contests/Contest/FirstPreferences/Blank/VotesByType/Votes/@Type

type	xs:string
properties	use required
source	<pre><xs:attribute name="Type" type="xs:string" use="required"/></pre>

Element Senate/Contests/Contest/FirstPreferences/Total

diagram	
properties	content complex
children	Votes VotesByType
source	<pre> <xs:element name="Total"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedInt"/> <xs:element name="VotesByType"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedInt"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element Senate/Contests/Contest/FirstPreferences/Total/Votes

diagram	
type	xs:unsignedInt
properties	content simple
source	<pre> <xs:element name="Votes" type="xs:unsignedInt"/> </pre>

Element Senate/Contests/Contest/FirstPreferences/Total/VotesByType

diagram	
properties	content complex
children	Votes

source	<pre> <xs:element name="VotesByType"> <xs:complexType> <xs:sequence> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedInt"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--------	---

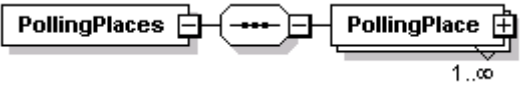
Element Senate/Contests/Contest/FirstPreferences/Total/VotesByType/Votes

diagram													
type	extension of xs:unsignedInt												
properties	minOcc 1 maxOcc unbounded content complex												
attributes	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Type</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Type	xs:string	required			
Name	Type	Use	Default	Fixed	Annotation								
Type	xs:string	required											
source	<pre> <xs:element name="Votes" maxOccurs="unbounded"> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:unsignedInt"> <xs:attribute name="Type" type="xs:string" use="required"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> </pre>												

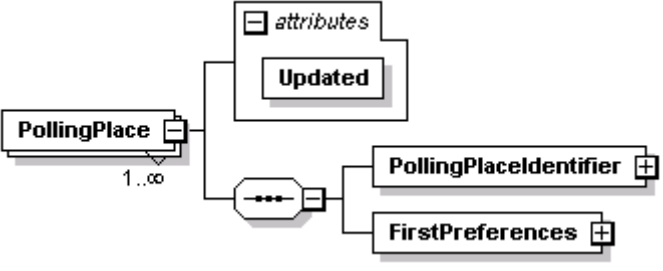
attribute Senate/Contests/Contest/FirstPreferences/Total/VotesByType/Votes/@Type

type	xs:string
properties	use required
source	<pre><xs:attribute name="Type" type="xs:string" use="required"/></pre>

Element Senate/Contests/Contest/PollingPlaces

diagram	
properties	content complex
children	PollingPlace

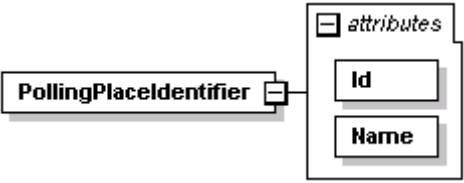
Element Senate/Contests/Contest/PollingPlaces/PollingPlace

diagram													
properties	minOcc 1 maxOcc unbounded content complex												
children	PollingPlacelIdentifier FirstPreferences												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Updated</td> <td>xs:dateTime</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Updated	xs:dateTime	required			
Name	Type	Use	Default	Fixed	Annotation								
Updated	xs:dateTime	required											

attribute Senate/Contests/Contest/PollingPlaces/PollingPlace/@Updated

type	xs:dateTime
properties	use required
source	<xs:attribute name="Updated" type="xs:dateTime" use="required"/>

Element Senate/Contests/Contest/PollingPlaces/PollingPlace/PollingPlacelIdentifier

diagram	
properties	content complex

attributes	Name	Type	Use	Default	Fixed	Annotation
	Id	xs:unsignedShort	required			
	Name	xs:string	required			

source
<pre><xs:element name="PollingPlaceIdentifier"> <xs:complexType> <xs:attribute name="Id" type="xs:string" use="required"/> <xs:attribute name="Name" type="xs:string" use="required"/> </xs:complexType> </xs:element></pre>

attribute **Senate/Contests/Contest/PollingPlaces/PollingPlace/PollingPlaceIdentifier/@Id**

type	xs:unsignedShort
properties	use required
source	<pre><xs:attribute name="Id" type="xs:string" use="required"/></pre>

attribute **Senate/Contests/Contest/PollingPlaces/PollingPlace/PollingPlaceIdentifier/@Name**

type	xs:string
properties	use required
source	<pre><xs:attribute name="Name" type="xs:string" use="required"/></pre>

Element Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences

diagram													
properties	content complex												
children	Group Formal Informal Total												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Updated</td> <td>xs:dateTime</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Updated	xs:dateTime	required			
Name	Type	Use	Default	Fixed	Annotation								
Updated	xs:dateTime	required											

attribute **Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/@Updated**

type	xs:dateTime
properties	use required
source	<code><xs:attribute name="Updated" type="xs:dateTime" use="required"/></code>

Element Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Group

diagram	<p>The diagram shows a class 'Group' with a multiplicity of '1..∞'. It contains a sequence of three elements: 'GroupIdentifier', 'TicketVotes', and 'GroupVotes'. Each of these elements is represented by a box with a plus sign in the top right corner, indicating they are complex types.</p>
properties	<p>minOcc 1</p> <p>maxOcc unbounded</p> <p>content complex</p>
children	GroupIdentifier TicketVotes GroupVotes
source	<pre> <xs:element name="Group" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="GroupIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="GroupName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element> <xs:element name="TicketVotes"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="GroupVotes"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Group/GroupIdentifier

diagram						
properties	content complex					
children	GroupName					
attributes	Name	Type	Use	Default	Fixed	Annotation
	Id	xs:unsignedShort	required			
source	<pre><xs:element name="GroupIdentifier"> <xs:complexType> <xs:sequence> <xs:element name="GroupName" type="xs:string"/> </xs:sequence> <xs:attribute name="Id" type="xs:string" use="required"/> </xs:complexType> </xs:element></pre>					

attribute Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Group/GroupIdentifier/@Id

type	xs:unsignedShort
properties	use required
source	<pre><xs:attribute name="Id" type="xs:string" use="required"/></pre>

Element Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Group/GroupIdentifier/GroupName

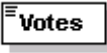
diagram						
type	xs:string					
properties	content simple					
source	<pre><xs:element name="GroupName" type="xs:string"/></pre>					

Element Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Group/GroupVotes


diagram						
properties	content complex					
children	Votes					

source	<pre><xs:element name="GroupVotes"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element></pre>
--------	--

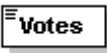
Element Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Group/GroupVotes/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<pre><xs:element name="Votes" type="xs:unsignedShort"/></pre>


Element Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Formal

diagram	
properties	content complex
children	Votes
source	<pre><xs:element name="Formal"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element></pre>

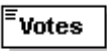
Element Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Formal/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<pre><xs:element name="Votes" type="xs:unsignedShort"/></pre>


Element MediaFeed/Election/Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Others

diagram	
properties	content complex
children	Votes
source	<pre><xs:element name="Others"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element></pre>

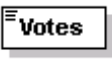
Element MediaFeed/Election/Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Others/ Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<pre><xs:element name="Votes" type="xs:unsignedShort"/></pre>

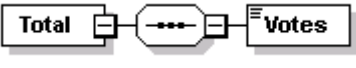
Element MediaFeed/Election/Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Blank

diagram	
properties	content complex
children	Votes
source	<pre><xs:element name="Blank"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element></pre>

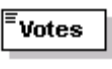
Element MediaFeed/Election/Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Blank/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<code><xs:element name="Votes" type="xs:unsignedShort"/></code>

Element Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Total

diagram	
properties	content complex
children	Votes
source	<pre><xs:element name="Total"> <xs:complexType> <xs:sequence> <xs:element name="Votes" type="xs:unsignedShort"/> </xs:sequence> </xs:complexType> </xs:element></pre>

Element Senate/Contests/Contest/PollingPlaces/PollingPlace/FirstPreferences/Total/Votes

diagram	
type	xs:unsignedShort
properties	content simple
source	<code><xs:element name="Votes" type="xs:unsignedShort"/></code>

LGE XML Schema Definition

Area

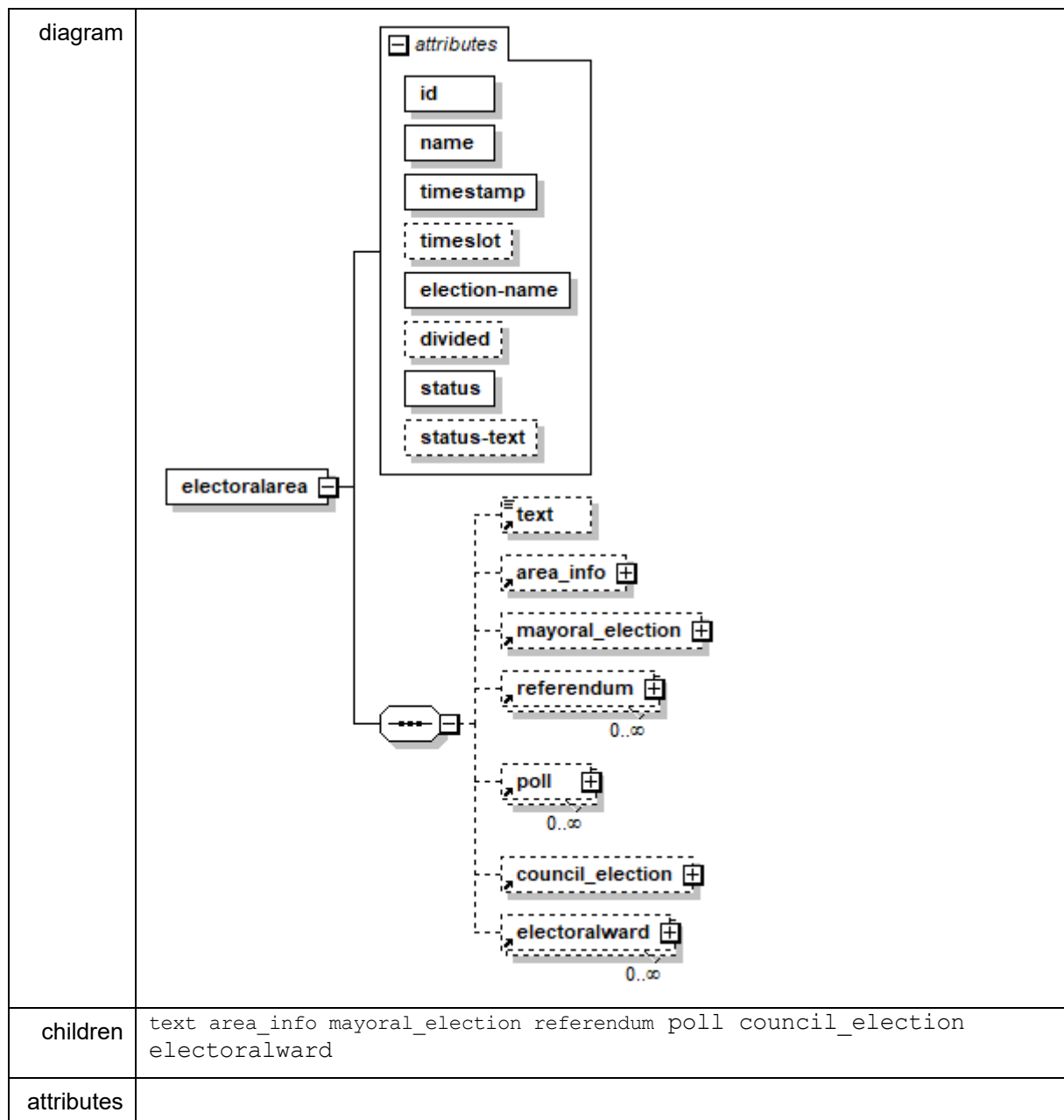
Element electoralarea

This is a container for the results in one LGA. An electoral Area may be divided or undivided.

For the Local Government Election 2021, there are 128 electoral areas plus the Unincorporated Area. Some areas are not clients of the NSWEC, these are listed on the NSWEC website.

Where an LGA is not a client of NSWEC, VTR will display no results, but will display a message. The message can be tailored for each LGA (or the Unincorporated Area) and is to appear in element <text>.

The attribute @status-description may also be used. This defines the text that will appear in VTR when the user rolls the mouse over the text "Status:".



source	<pre> <xs:element name="electoralarea"> <xs:complexType> <xs:sequence> <xs:element ref="text" minOccurs="0"/> <xs:element ref="area_info" minOccurs="0"/> <xs:element ref="mayoral_election" minOccurs="0"/> <xs:element ref="referendum" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="poll" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="council_election" minOccurs="0"/> <xs:element ref="electoralward" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence> <xs:attribute name="id" type="xs:ID" use="required"/> <xs:attribute name="name" type="xs:anySimpleType" use="required"/> <xs:attribute name="timestamp" type="xs:dateTime" use="required"/> <xs:attribute name="timeslot" type="xs:anySimpleType"/> <xs:attribute name="election-name" type="xs:anySimpleType" use="required"/> <xs:attribute name="divided" default="N"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="N"/> <xs:enumeration value="Y"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="status" use="required"> </pre>
--------	---

	<pre> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="Council-Run"/> <xs:enumeration value="Deferred"/> <xs:enumeration value="NSWEC-Run"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="status-text" type="xs:anySimpleType"/> </xs:complexType> </xs:element> </pre>
--	--

Element area_info

Contains fixed information about an electoral area. This information will not change during the course of the election.

diagram	
children	total_number_electors total_number_polling_places
used by	electoralarea
attributes	
source	<pre> <xs:element name="area_info"> <xs:complexType> <xs:sequence> <xs:element ref="total_number_electors"/> <xs:element ref="total_number_polling_places"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

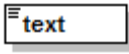
Element electoralward

This is a container for the results in one Ward in a divided LGA. Element <text> can be used to display specific text for this ward only.

<p>diagram</p>													
<p>children</p>	<p>text ward_info council_election</p>												
<p>used by</p>	<p>electoralarea</p>												
<p>attributes</p>	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>id</td> <td>optional</td> <td>Unique ID for the Ward</td> </tr> <tr> <td>name</td> <td>required</td> <td>Name of the ward. For example "North Ward"</td> </tr> <tr> <td>timestamp</td> <td>required</td> <td>Date and time that this data was updated.</td> </tr> </tbody> </table>	Name	Use	Description	id	optional	Unique ID for the Ward	name	required	Name of the ward. For example "North Ward"	timestamp	required	Date and time that this data was updated.
Name	Use	Description											
id	optional	Unique ID for the Ward											
name	required	Name of the ward. For example "North Ward"											
timestamp	required	Date and time that this data was updated.											
<p>source</p>	<pre> <xs:element name="electoralward"> <xs:complexType> <xs:sequence> <xs:element ref="text" minOccurs="0"/> <xs:element ref="ward_info" minOccurs="0"/> <xs:element ref="council_election"/> </xs:sequence> <xs:attribute name="id" type="xs:ID" use="optional"/> <xs:attribute name="name" type="xs:anySimpleType" use="required"/> <xs:attribute name="timestamp" type="xs:dateTime" use="required"/> </xs:complexType> </xs:element> </pre>												

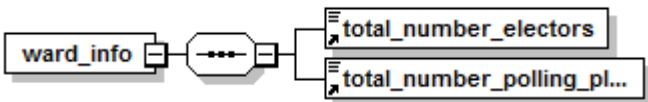
Element text

Contains fixed information about a ward in a divided LGA, which will not change during the course of the election.

diagram	
children	None
used by	electoralarea
attributes	
source	<pre><xs:element name="text"> <xs:complexType mixed="true"/> </xs:element></pre>


Element ward_info

Contains fixed information about a ward in a divided LGA, which will not change during the course of the election.

diagram	
children	total_number_electors total_number_polling_places
used by	electoralward
attributes	
source	<pre><xs:element name="ward_info"> <xs:complexType> <xs:sequence> <xs:element ref="total_number_electors"/> <xs:element ref="total_number_polling_places"/> </xs:sequence> </xs:complexType> </xs:element></pre>


Element total_number_electors

The number of electors enrolled in an LGA or the total number of electors enrolled in a ward, on a specified date prior to the election.

diagram	 total_number_electors
children	None
used by	area_info ward_info poll_info
attributes	
source	<pre><xs:element name="total_number_electors"> <xs:complexType mixed="true"/> </xs:element></pre>

Element total_number_polling_places

The total number of polling places in an LGA or the total number of polling places in an Ward.

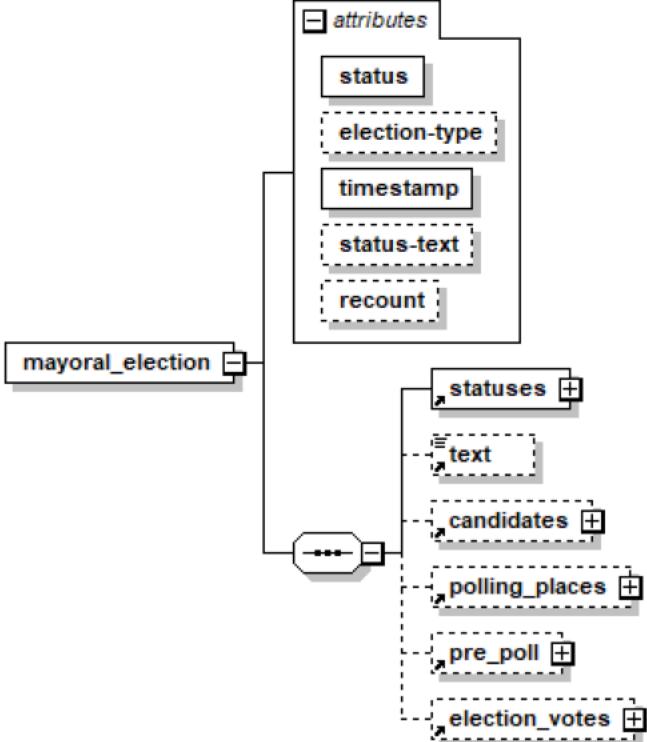
diagram	 total_number_polling_pl...
children	None
used by	area_info ward_info poll_info
attributes	
source	<pre><xs:element name="total_number_polling_places"> <xs:complexType mixed="true"/> </xs:element></pre>

Elections

Element mayor_election

The top-level element for a mayoral election. Optional @election-type takes the value “OP” for all Mayoral elections and may be omitted from the XML.

Child element <text> can be used to display a specific message relating to this contest. For example, if the contest is uncontested.

<p>diagram</p>										
<p>children</p>	<p>status text candidates polling_places pre_poll election_votes</p>									
<p>used by</p>	<p>electoralarea</p>									
<p>attributes</p>	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>status</td> <td>required</td> <td>Allowed values: No-Status Uncontested Deferred FC-Enabled Final-for-Saturday CC-Enabled CC-Dec-Complete Publish-Results Declare-Election Uncontested-Declare-Election</td> </tr> <tr> <td>election-type</td> <td>required</td> <td>Allowed values OP – Optional Preferential. (Will always be OP for Mayoral election). (PR – Does not apply for Mayoral Election)</td> </tr> </tbody> </table>	Name	Use	Description	status	required	Allowed values: No-Status Uncontested Deferred FC-Enabled Final-for-Saturday CC-Enabled CC-Dec-Complete Publish-Results Declare-Election Uncontested-Declare-Election	election-type	required	Allowed values OP – Optional Preferential. (Will always be OP for Mayoral election). (PR – Does not apply for Mayoral Election)
Name	Use	Description								
status	required	Allowed values: No-Status Uncontested Deferred FC-Enabled Final-for-Saturday CC-Enabled CC-Dec-Complete Publish-Results Declare-Election Uncontested-Declare-Election								
election-type	required	Allowed values OP – Optional Preferential. (Will always be OP for Mayoral election). (PR – Does not apply for Mayoral Election)								

	<p>timestamp required Date and time that this data was updated.</p> <p>recount required A non-zero value indicates a recount has commenced</p> <p>status-text optional Can be used to define LGA and contest specific text for the given value of @status. This defines the text that will appear in VTR when the user rolls the mouse over the text "Status:". However, see also child element <text></p>
source	<pre> <xs:element name="mayoral_election"> <xs:complexType> <xs:sequence> <xs:element ref="statuses"/> <xs:element ref="text" minOccurs="0"/> <xs:element ref="candidates" minOccurs="0"/> <xs:element ref="polling_places" minOccurs="0"/> <xs:element ref="pre_poll" minOccurs="0"/> <xs:element ref="election_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="status" use="required"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="CC-Dec-Complete"/> <xs:enumeration value="Uncontested"/> <xs:enumeration value="CC-Enabled"/> <xs:enumeration value="No-Status"/> <xs:enumeration value="Deferred"/> <xs:enumeration value="Final-for-Saturday"/> <xs:enumeration value="FC-Enabled"/> <xs:enumeration value="Uncontested-Declare-Election"/> <xs:enumeration value="Publish-Results"/> <xs:enumeration value="Declare-Election"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="election-type"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="OP"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="timestamp" type="xs:dateTime" use="required"/> <xs:attribute name="status-text" type="xs:anySimpleType"/> <xs:attribute name="recount" type="xs:integer"/> </xs:complexType> </xs:element> </pre>

Element council_election

The top level element for a council election. Optional attribute @election-type will always be “PR” as of 2012, and may be omitted from the XML.

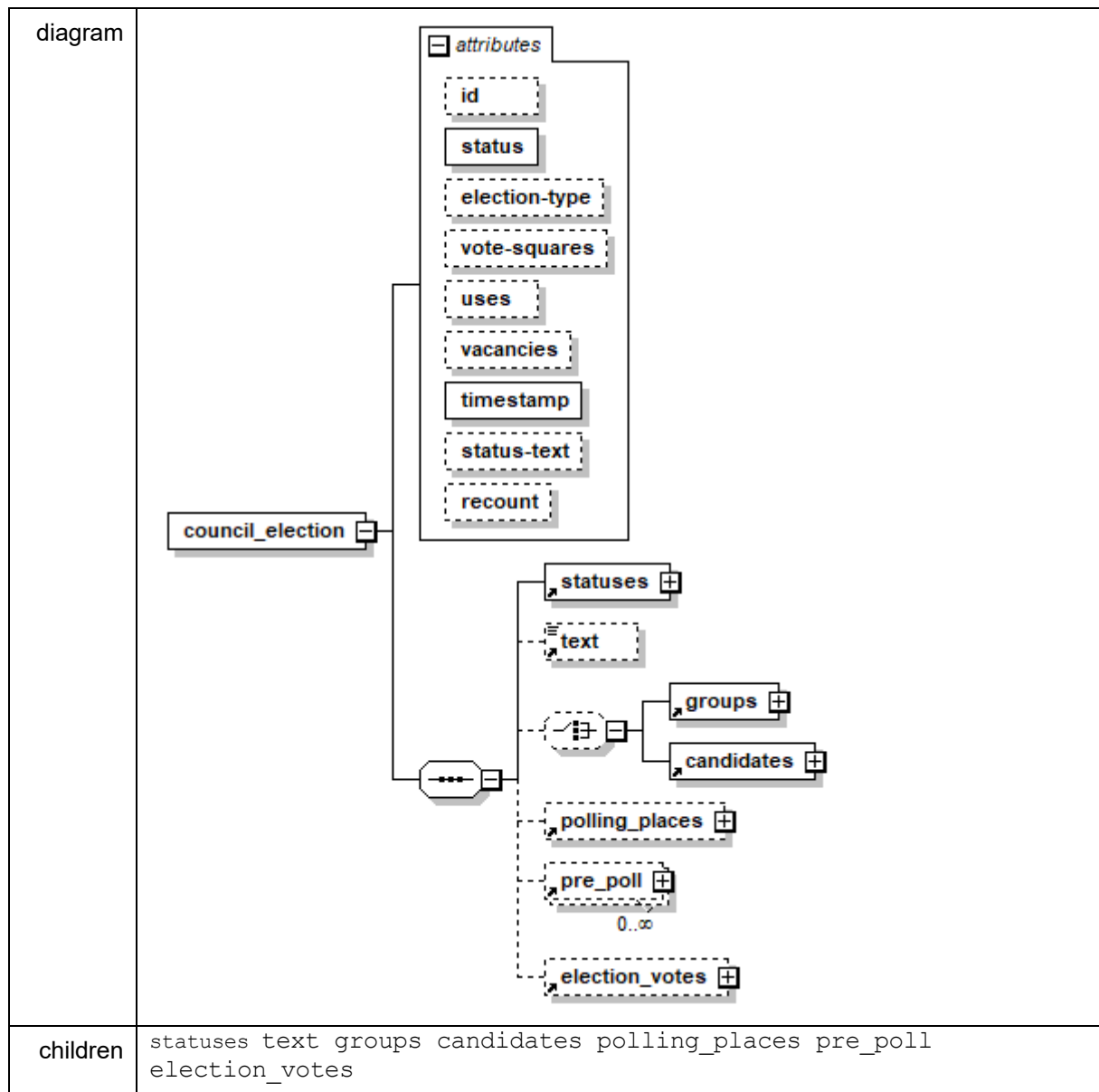
Attribute @uses will always take the value “prcc” in 2012 and may be omitted from the XML.

Element //polling-places are specified unless the election is deferred or uncontested.

Attribute @vote-squares indicates whether or not the ballot paper contains group voting squares. Group voting squares may be above the line (ATL) or below the line (BTL).

Where there are a large number of candidates, it is possible that results for some candidates may not be reported on election night. Votes for “Unreported Candidates” are bundled in “Other Ballots”, which will also include informal votes. This could apply equally for ballot papers with both ATL and BTL squares.

Child element <text> can be used to display a specific message relating to this contest. For example, if the contest is uncontested.

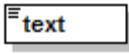


used by	electoralarea electoralward		
attributes	Name	Use	Description
	status	required	Allowed values: No-Status Uncontested Deferred FC-Enabled Final-for-Saturday Dec-Complete Publish-Results Declare-Election Uncontested-Declare-Election
	election-type	optional	Allowed values PR – Proportional Representation OP – Optional Preferential However, all elections in 2012 are of type “PR” and this attribute may be omitted completely from the XML in 2012
	vote-square	optional	Allowed values ATL – Above The Line voting squares BTL – Below The Line voting squares
	uses	optional	Allowed values: prcc This attribute may be omitted completely from the XML in 2012.
	vacancies	optional	number of vacancies to be filled in the election
	Id	optional	Indicates the contest identifier.
	timestamp	required	Date and time that this data was updated.
	status-text	optional	Can be used to define LGA and contest specific text for the given value of @status. This defines the text that will appear in VTR when the user rolls the mouse over the text “Status:”. However, see also child element <text>

source	<pre> <xs:element name="council_election"> <xs:complexType> <xs:sequence> <xs:element ref="statuses"/> <xs:element ref="text" minOccurs="0"/> <xs:choice minOccurs="0"> <xs:element ref="groups"/> <xs:element ref="candidates"/> </xs:choice> <xs:element ref="polling_places" minOccurs="0"/> <xs:element ref="pre_poll" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="election_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="id" type="xs:string"/> <xs:attribute name="status" use="required"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="Uncontested"/> <xs:enumeration value="Dec-Complete"/> <xs:enumeration value="No-Status"/> <xs:enumeration value="Deferred"/> <xs:enumeration value="Final-for-Saturday"/> <xs:enumeration value="FC-Enabled"/> <xs:enumeration value="Uncontested-Declared-Election"/> <xs:enumeration value="Publish-Results"/> <xs:enumeration value="Declare-Election"/> <xs:enumeration value="CC-Enabled"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="election-type"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="OP"/> <xs:enumeration value="PR"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="vote-squares"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="BTL"/> <xs:enumeration value="ATL"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="uses"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="prcc"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="vacancies" type="xs:NMTOKEN"/> <xs:attribute name="timestamp" type="xs:dateTime" use="required"/> <xs:attribute name="status-text" type="xs:anySimpleType"/> <xs:attribute name="recount" type="xs:integer"/> </xs:complexType> </xs:element> </pre>
--------	---

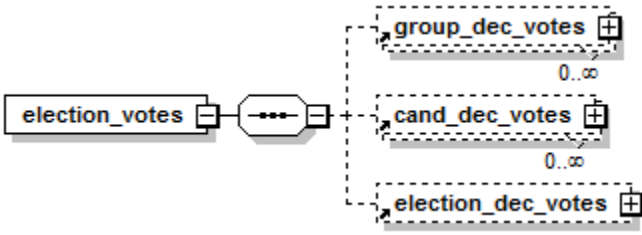
Element text

This is text that can be displayed for a specific LGA and contest. For example this might be a statement of the reason why the election event has been deferred.

diagram	
children	None
used by	electoralarea electoralward mayoral_election council_election referendum poll
attributes	
source	<pre><xs:element name="text"> <xs:complexType mixed="true"/> </xs:element></pre>

Element election_votes

Votes associated with the contest, but not with a specified PP.

diagram	
children	group_dec_votes cand_dec_votes election_dec_votes
used by	mayoral_election council_election
attributes	
source	<pre><xs:element name="election_votes"> <xs:complexType> <xs:sequence> <xs:element ref="group_dec_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="cand_dec_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="election_dec_votes" minOccurs="0"/> </xs:sequence> </xs:complexType> </xs:element></pre>

Element polling_places

A wrapper element for one or more PPs.

diagram	
children	polling_place_mayor polling_place_council polling_place_poll polling_place_ref
used by	mayoral_election council_election question
attributes	
source	<pre> <xs:element name="polling_places"> <xs:complexType> <xs:choice> <xs:element ref="polling_place_mayor" maxOccurs="unbounded"/> <xs:element ref="polling_place_council" maxOccurs="unbounded"/> <xs:element ref="polling_place_poll" maxOccurs="unbounded"/> <xs:element ref="polling_place_ref" maxOccurs="unbounded"/> </xs:choice> </xs:complexType> </xs:element> </pre>

Element polling_place_mayor

The element wrapper for a PP in the context of a mayoral election.

diagram	
children	cand_votes informal_votes
used by	polling_places

attributes	Name	Use	Description
	gazetted_name	required	The official gazetted name of the PP.
	Id	Optional	Venue identifier
	Status	Optional	Indicates the result status of the venue
	gazetted_name	required	The official gazetted name of the PP.
source	<pre> <xs:element name="polling_place_mayor"> <xs:complexType> <xs:sequence> <xs:element ref="question_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="informal_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="gazetted_name" type="xs:anySimpleType" use="required"/> <xs:attribute name="Id" type="xs:anySimpleType" use="optional"/> <xs:attribute name="status" type="xs:anySimpleType" use="optional"/> </xs:complexType> </xs:element> </pre>		

Element polling_place_council

The element wrapper for a PP in the context of a council election.

diagram													
children	group_votes cand_votes informal_votes												
used by	polling_places												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>gazetted_name</td> <td>required</td> <td>The official gazetted name of the PP.</td> </tr> <tr> <td>Id</td> <td>Optional</td> <td>Venue identifier</td> </tr> <tr> <td>Status</td> <td>Optional</td> <td>Indicates the result status of the venue</td> </tr> </tbody> </table>	Name	Use	Description	gazetted_name	required	The official gazetted name of the PP.	Id	Optional	Venue identifier	Status	Optional	Indicates the result status of the venue
Name	Use	Description											
gazetted_name	required	The official gazetted name of the PP.											
Id	Optional	Venue identifier											
Status	Optional	Indicates the result status of the venue											

source	<pre> <xs:element name="polling_place_council"> <xs:complexType> <xs:sequence> <xs:element ref="question_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="informal_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="gazetted_name" type="xs:anySimpleType" use="required"/> <xs:attribute name="Id" type="xs:anySimpleType" use="optional"/> <xs:attribute name="status" type="xs:anySimpleType" use="optional"/> </xs:complexType> </xs:element> </pre>
---------------	--

Element polling_place_poll

The element wrapper for a PP in the context of a referendum or a poll.

diagram													
children	question_votes informal_votes												
used by	polling_places												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>gazetted_name</td> <td>required</td> <td>The official gazetted name of the PP.</td> </tr> <tr> <td>Id</td> <td>Optional</td> <td>Venue identifier</td> </tr> <tr> <td>Status</td> <td>Optional</td> <td>Indicates the result status of the venue</td> </tr> </tbody> </table>	Name	Use	Description	gazetted_name	required	The official gazetted name of the PP.	Id	Optional	Venue identifier	Status	Optional	Indicates the result status of the venue
Name	Use	Description											
gazetted_name	required	The official gazetted name of the PP.											
Id	Optional	Venue identifier											
Status	Optional	Indicates the result status of the venue											
source	<pre> <xs:element name="polling_place_poll"> <xs:complexType> <xs:sequence> <xs:element ref="question_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="informal_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="gazetted_name" type="xs:anySimpleType" use="required"/> <xs:attribute name="Id" type="xs:anySimpleType" use="optional"/> <xs:attribute name="status" type="xs:anySimpleType" use="optional"/> </xs:complexType> </xs:element> </pre>												

Element polling_place_ref

The element wrapper for a PP in the context of a referendum or a poll.

<p>diagram</p>													
<p>children</p>	<p>question_votes informal_votes</p>												
<p>used by</p>	<p>polling_places</p>												
<p>attributes</p>	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>gazetted_name</td> <td>required</td> <td>The official gazetted name of the PP.</td> </tr> <tr> <td>Id</td> <td>Optional</td> <td>Venue identifier</td> </tr> <tr> <td>Status</td> <td>Optional</td> <td>Indicates the result status of the venue</td> </tr> </tbody> </table>	Name	Use	Description	gazetted_name	required	The official gazetted name of the PP.	Id	Optional	Venue identifier	Status	Optional	Indicates the result status of the venue
Name	Use	Description											
gazetted_name	required	The official gazetted name of the PP.											
Id	Optional	Venue identifier											
Status	Optional	Indicates the result status of the venue											
<p>source</p>	<pre><xs:element name="polling_place_ref"> <xs:complexType> <xs:sequence> <xs:element ref="question_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="informal_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="gazetted_name" type="xs:anySimpleType" use="required"/> <xs:attribute name="Id" type="xs:anySimpleType" use="optional"/> <xs:attribute name="status" type="xs:anySimpleType" use="optional"/> </xs:complexType> </xs:element></pre>												

Element pre_poll

A wrapper element for one or more PPs.

diagram	
children	pre_poll_mayor pre_poll_council pre_poll_poll pre_poll_ref
used by	mayoral_election council_election question
attributes	
source	<pre> <xs:element name="pre_poll"> <xs:complexType> <xs:choice> <xs:element ref="pre_poll_mayor" maxOccurs="unbounded"/> <xs:element ref="pre_poll_council" maxOccurs="unbounded"/> <xs:element ref="pre_poll_poll" maxOccurs="unbounded"/> <xs:element ref="pre_poll_ref" maxOccurs="unbounded"/> </xs:choice> </xs:complexType> </xs:element> </pre>

Element pre_poll_mayor

The element wrapper for a PP in the context of a mayoral election.

diagram	
children	cand_votes informal_votes
used by	Pre_poll

attributes	Name	Use	Description
	gazetted_name	required	The official gazetted name of the PP.
	Id	Optional	Venue identifier
	Status	Optional	Indicates the result status of the venue
source	<pre> <xs:element name="pre_poll_mayor"> <xs:complexType> <xs:sequence> <xs:element ref="cand_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="informal_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="gazetted_name" type="xs:anySimpleType" use="required"/> <xs:attribute name="Id" type="xs:string"/> <xs:attribute name="status" type="xs:string"/> </xs:complexType> </xs:element> </pre>		

Element polling_place_council

The element wrapper for a PP in the context of a council election.

diagram			
children	group_votes cand_votes informal_votes		
used by	Pre_poll		
attributes	Name	Use	Description
	gazetted_name	required	The official gazetted name of the PP.
	Id	Optional	Venue identifier
	Status	Optional	Indicates the result status of the venue

source	<pre> <xs:element name="pre_poll_council"> <xs:complexType> <xs:sequence> <xs:element ref="group_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="cand_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="informal_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="gazetted_name" type="xs:anySimpleType" use="required"/> <xs:attribute name="Id" type="xs:string"/> <xs:attribute name="status" type="xs:string"/> </xs:complexType> </xs:element> </pre>
--------	---

Element pre_poll_poll

The element wrapper for a PP in the context of a referendum or a poll.

diagram															
children	question_votes informal_votes														
used by	Pre_poll														
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>gazetted_name</td> <td>required</td> <td>The official gazetted name of the PP.</td> </tr> <tr> <td>Id</td> <td>Optional</td> <td>Venue identifier</td> </tr> <tr> <td>Status</td> <td>Optional</td> <td>Indicates the result status of the venue</td> </tr> </tbody> </table>			Name	Use	Description	gazetted_name	required	The official gazetted name of the PP.	Id	Optional	Venue identifier	Status	Optional	Indicates the result status of the venue
Name	Use	Description													
gazetted_name	required	The official gazetted name of the PP.													
Id	Optional	Venue identifier													
Status	Optional	Indicates the result status of the venue													
source	<pre> <xs:element name="pre_poll_poll"> <xs:complexType> <xs:sequence> <xs:element ref="question_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="informal_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="gazetted_name" type="xs:anySimpleType" use="required"/> <xs:attribute name="Id" type="xs:anySimpleType" use="optional"/> <xs:attribute name="status" type="xs:anySimpleType" use="optional"/> </xs:complexType> </xs:element> </pre>														

Element polling_place_ref

The element wrapper for a PP in the context of a referendum or a poll.

diagram													
children	question_votes informal_votes												
used by	Pre_poll												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>gazetted_name</td> <td>required</td> <td>The official gazetted name of the PP.</td> </tr> <tr> <td>Id</td> <td>Optional</td> <td>Venue identifier</td> </tr> <tr> <td>Status</td> <td>Optional</td> <td>Indicates the result status of the venue</td> </tr> </tbody> </table>	Name	Use	Description	gazetted_name	required	The official gazetted name of the PP.	Id	Optional	Venue identifier	Status	Optional	Indicates the result status of the venue
Name	Use	Description											
gazetted_name	required	The official gazetted name of the PP.											
Id	Optional	Venue identifier											
Status	Optional	Indicates the result status of the venue											
source	<pre> <xs:element name="pre_poll_ref"> <xs:complexType> <xs:sequence> <xs:element ref="question_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="informal_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="gazetted_name" type="xs:anySimpleType" use="required"/> <xs:attribute name="Id" type="xs:anySimpleType" use="optional"/> <xs:attribute name="status" type="xs:anySimpleType" use="optional"/> </xs:complexType> </xs:element> </pre>												

Element informal_votes

Number of informal votes. This includes the Other votes (ATL and BTL type of votes) for councillor election with groups. May include both votes counted on election night and the number of votes counted again during the check count.

diagram	
children	fp_votes cc_votes
used by	polling_place_mayor polling_place_council polling_place_poll
attributes	

source	<pre> <xs:element name="informal_votes"> <xs:complexType> <xs:sequence> <xs:element ref="fp_votes" minOccurs="0"/> <xs:element ref="cc_votes" minOccurs="0"/> </xs:sequence> </xs:complexType> </xs:element> </pre>
--------	---

Element fp_votes

The number of first preference Votes.

diagram	
children	None
used by	informal_votes cand_votes group_votes votes_yes votes_no
attributes	
source	<pre> <xs:element name="fp_votes"> <xs:complexType mixed="true"/> </xs:element> </pre>

Element cc_votes

The number of check count Votes. The actual number of votes may differ slightly to the number of \fp_votes.

diagram	
children	None
used by	informal_votes cand_votes group_votes votes_yes votes_no
attributes	
source	<pre> <xs:element name="cc_votes"> <xs:complexType mixed="true"/> </xs:element> </pre>

Candidates

Element candidates

This is a wrapper element for one or more candidates.

diagram	
---------	--

children	candidate
used by	mayoral_election council_election
attributes	
source	<pre> <xs:element name="candidates"> <xs:complexType> <xs:sequence> <xs:element ref="candidate" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element candidate

A candidate. The attribute @ballot_name is a unique identifier for each candidate within an electoral contest. The attribute @href is used elsewhere in the XML to identify each candidate. In each case where @href is used, its value must match a value of @ballot_name, which must be unique.

Where two candidates in one area share the same surname, it is essential that @ballot_name is unique within that area.

Where a candidate has a “double candidature” by being a candidate in both a Mayoral election and a Council election, then in each instance they have a different and unique ID.

In addition to the alphabetical characters “A” ... “Z”, a candidate’s Ballot Name may also contain the following characters: “ ” (space), “-” (hyphen), “'” (apostrophe). No other characters are allowed.

diagram	
children	surname givennames drawnumber partyaffiliation
used by	candidates group ungrouped


attributes	Name	Use	Description
	id	required	Candidate's unique ID.
	Ballot_name	required	Candidate's unique official ballot name
	elected	optional	<p>This candidate has been elected in a given count. If value is equal "1", then the candidate has been elected.</p> <p>If candidate has not been elected, then this attribute is unspecified and defaults to "0".</p>
	elected-mayor	optional	<p>This candidate (in a Council election) has been elected mayor in the mayoral election and is excluded. If value is equal "1", then the candidate has been elected mayor.</p> <p>If candidate has not been elected mayor, then this attribute is unspecified and defaults to "0".</p>
	elected-position	optional	This candidate has been elected in position "1", "2", "3", etc. @elected-position is only specified for Council elections, and only if @elected has been specified.
	elected-count	optional	The count in which the candidate was elected. @elected-count is only specified for Council elections, and only if @elected has been specified.
	elected-date	optional	<p>Gives the date elected if the candidate has @elected="1".</p> <p>Format is: DD:MM:YYYY. Local date.</p>
	election_night_count	required	<p>Indicates whether or not votes are to be counted for this candidate on election night. Values:</p> <ul style="list-style-type: none"> "1" (default) – count on election night. "0" – do not count on election night. <p>This attribute will be used primarily by processes that generate the blank Results Reports sheets used by RO's on election night and on Sunday.</p>

source	<pre> <xs:element name="candidate"> <xs:complexType> <xs:sequence> <xs:element ref="surname"/> <xs:element ref="givennames"/> <xs:element ref="drawnumber" minOccurs="0"/> <xs:element ref="partyaffiliation"/> </xs:sequence> <xs:attribute name="id" type="xs:integer" use="required"/> <xs:attribute name="ballot_name" type="xs:anySimpleType" use="required"/> <xs:attribute name="elected" default="0"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="0"/> <xs:enumeration value="1"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="elected-mayor" default="0"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="0"/> <xs:enumeration value="1"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="elected-count" type="xs:NMTOKEN"/> <xs:attribute name="elected-position" type="xs:NMTOKEN"/> <xs:attribute name="elected-date" type="xs:anySimpleType"/> <xs:attribute name="election_night_count" default="1"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="0"/> <xs:enumeration value="1"/> </xs:restriction> </xs:simpleType> </xs:attribute> </xs:complexType> </xs:element> </pre>
--------	--

Element surname


Candidate's surname.

In addition to the alphabetical characters "A" ... "Z", a candidate's Name may also contain the following characters: " " (space), "-" (hyphen), "'" (apostrophe).

diagram	 surname
children	None
used by	candidate
attributes	
source	<pre><xs:element name="surname"> <xs:complexType mixed="true"/> </xs:element></pre>


Element givennames

Candidate's first name(s).

diagram	 givennames
children	None
used by	candidate
attributes	
source	<pre><xs:element name="givennames"> <xs:complexType mixed="true"/> </xs:element></pre>

Element drawnumber

The Draw Number of the candidate. This has a numeric value from 1 to N.

diagram	 drawnumber
children	None
used by	candidate group
attributes	
source	<pre><xs:element name="drawnumber"> <xs:complexType mixed="true"/> </xs:element></pre>

Element partyaffiliation

The party to which the candidate is affiliated.

diagram	
children	long_party_name abbreviated_party_name
used by	candidate
attributes	
source	<pre> <xs:element name="partyaffiliation"> <xs:complexType> <xs:sequence> <xs:element ref="long_party_name"/> <xs:element ref="abbreviated_party_name" minOccurs="0"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element long_party_name

The full name of the party to which the candidate belongs.

diagram	
children	None
used by	partyaffiliation
attributes	
source	<pre> <xs:element name="long_party_name"> <xs:complexType mixed="true"/> </xs:element> </pre>

Element abbreviated_party_name

The abbreviation of the party to which the candidate belongs. Independant candidates might not be provided with an abbreviation, in which case the element will occur in the XML data as an empty element <abbrev/>.

diagram	
children	None
used by	partyaffiliation
attributes	

source	<pre><xs:element name="abbreviated_party_name"> <xs:complexType mixed="true"/> </xs:element></pre>
---------------	--

Element cand_votes

The votes cast for a candidate. First Preference Votes are recorded, but also the result of the check count.

Check Count Votes are included for Mayoral and Council (OP) elections once the check count is complete and the status of the electoral contest is “CC-Complete”.

diagram							
children	fp_votes cc_votes						
used by	polling_place_mayor polling_place_council						
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>href</td> <td>required</td> <td>The value of @href will match one of the values of @id of a candidate within the electoral contest.</td> </tr> </tbody> </table>	Name	Use	Description	href	required	The value of @href will match one of the values of @id of a candidate within the electoral contest.
Name	Use	Description					
href	required	The value of @href will match one of the values of @id of a candidate within the electoral contest.					
source	<pre><xs:element name="cand_votes"> <xs:complexType> <xs:sequence> <xs:element ref="fp_votes" minOccurs="0"/> <xs:element ref="cc_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="href" type="xs:IDREF" use="required"/> </xs:complexType> </xs:element></pre>						

Declaration Votes

Element group_dec_votes

Formal first preferences for each Group by Declaration Vote Type. These include prepoll votes and other votes not cast on the day of the election.

Note: This includes both SATL and RATL votes that are counted separately, however reported together as one figure.

There are no absent votes in the 2012 LGE and onwards.

diagram	
children	dec_votes
used by	election_votes
attributes	
source	<pre> <xs:element name="group_dec_votes"> <xs:complexType> <xs:sequence minOccurs="0" maxOccurs="unbounded"> <xs:element ref="dec_votes"/> </xs:sequence> <xs:attribute name="href" type="xs:IDREF" use="required"/> </xs:complexType> </xs:element> </pre>

Element cand_dec_votes

Formal first preferences for each Candidate by Declaration Vote Type. These include prepoll votes and other votes not cast on the day of the election.

Note: This includes both SATL and RATL votes that are counted separately, however reported together as one figure.

diagram	
children	dec_votes
used by	election_votes
attributes	

source	<pre> <xs:element name="cand_dec_votes"> <xs:complexType> <xs:sequence minOccurs="0" maxOccurs="unbounded"> <xs:element ref="dec_votes"/> </xs:sequence> <xs:attribute name="href" type="xs:IDREF" use="required"/> </xs:complexType> </xs:element> </pre>
--------	--

Element election_dec_votes

Declaration Votes are sent to the district office throughout the week following election night. Ballots are sorted and grouped into:

- Formal first preferences for each Candidate by Declaration Vote Type; and
- Informal votes by Declaration Vote Type.

diagram	
children	dec_votes
used by	election_votes referendum_votes
attributes	
source	<pre> <xs:element name="election_dec_votes"> <xs:complexType> <xs:sequence minOccurs="0" maxOccurs="unbounded"> <xs:element ref="dec_votes"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element `dec_votes`

Declaration votes.

Where the parent element of `//dec_votes` is `//cand_dec_votes`, only the `@type` value "fp" is allowed.

Where the parent element of `//dec_votes` is `//area_dec_votes`, only the `@type` value "informal" is used.

`@type` value "exhausted" is not used for the 2012 election and onwards.

The value of `@dec_vote_type` is of type string and may vary from one election to another. Possible values are:

- Postal
- diprepoll (Declared Inst + Prepoll)
- prosilent (Provisional + Silent, if Silent is used in an event)
- ivote
- di_decvotes

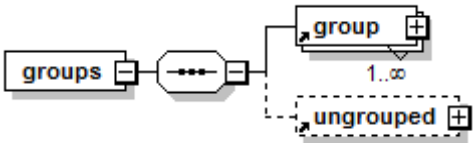
<p>diagram</p>													
<p>children</p>	<p><code>progressive_count</code></p>												
<p>used by</p>	<p><code>group_dec_votes</code> <code>cand_dec_votes</code> <code>election_dec_votes</code> <code>dec_votes_yes</code> <code>dec_votes_no</code></p>												
<p>attributes</p>	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td><code>dec_vote_type</code></td> <td>required</td> <td>The dec vote type. These may vary from one election to another.</td> </tr> <tr> <td><code>status</code></td> <td>optional</td> <td>Shows the status of results entry</td> </tr> <tr> <td><code>type</code></td> <td>required</td> <td>Values: fp informal exhausted (Not used)</td> </tr> </tbody> </table>	Name	Use	Description	<code>dec_vote_type</code>	required	The dec vote type. These may vary from one election to another.	<code>status</code>	optional	Shows the status of results entry	<code>type</code>	required	Values: fp informal exhausted (Not used)
Name	Use	Description											
<code>dec_vote_type</code>	required	The dec vote type. These may vary from one election to another.											
<code>status</code>	optional	Shows the status of results entry											
<code>type</code>	required	Values: fp informal exhausted (Not used)											

source	<pre> <xs:element name="dec_votes"> <xs:complexType mixed="true"> <xs:sequence> <xs:element ref="progressive_count" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence> <xs:attribute name="dec_vote_type" type="xs:anySimpleType" use="required"/> <xs:attribute name="status" type="xs:string"/> <xs:attribute name="type" use="required"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="exhausted"/> <xs:enumeration value="informal"/> <xs:enumeration value="fp"/> </xs:restriction> </xs:simpleType> </xs:attribute> </xs:complexType> </xs:element> </pre>
--------	--

Groups

Element groups

This element is a wrapper for the individual Groups for grouped candidates.

diagram	
children	group ungrouped
used by	council_election
attributes	
source	<pre> <xs:element name="groups"> <xs:complexType> <xs:sequence> <xs:element ref="group" maxOccurs="unbounded"/> <xs:element ref="ungrouped" minOccurs="0"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element group

A group of candidates.

<p>diagram</p>													
<p>children</p>	<p>groupname drawnumber candidate</p>												
<p>used by</p>	<p>groups</p>												
<p>attributes</p>	<table border="1"> <thead> <tr> <th data-bbox="338 936 735 992">Name</th> <th data-bbox="735 936 895 992">Use</th> <th data-bbox="895 936 1396 992">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="338 992 735 1048">id</td> <td data-bbox="735 992 895 1048">required</td> <td data-bbox="895 992 1396 1048">Unique ID for the group.</td> </tr> <tr> <td data-bbox="338 1048 735 1104">has_group_voting_Square</td> <td data-bbox="735 1048 895 1104">required</td> <td data-bbox="895 1048 1396 1104">Value = Y if group has voting square</td> </tr> <tr> <td data-bbox="338 1104 735 1469">election_night_count</td> <td data-bbox="735 1104 895 1469">required</td> <td data-bbox="895 1104 1396 1469"> Indicates whether or not votes are to be counted for this group on election night. Values: "1" (default) – count on election night. "0" – do not count on election night. This attribute was used (in 2008) by processes that generate the blank Results Reports sheets used by RO's on election night and on Sunday. </td> </tr> </tbody> </table>	Name	Use	Description	id	required	Unique ID for the group.	has_group_voting_Square	required	Value = Y if group has voting square	election_night_count	required	Indicates whether or not votes are to be counted for this group on election night. Values: "1" (default) – count on election night. "0" – do not count on election night. This attribute was used (in 2008) by processes that generate the blank Results Reports sheets used by RO's on election night and on Sunday.
Name	Use	Description											
id	required	Unique ID for the group.											
has_group_voting_Square	required	Value = Y if group has voting square											
election_night_count	required	Indicates whether or not votes are to be counted for this group on election night. Values: "1" (default) – count on election night. "0" – do not count on election night. This attribute was used (in 2008) by processes that generate the blank Results Reports sheets used by RO's on election night and on Sunday.											

source	<pre> <xs:element name="group"> <xs:complexType> <xs:sequence> <xs:element ref="groupname" minOccurs="0"/> <xs:element ref="drawnumber" minOccurs="0"/> <xs:element ref="candidate" maxOccurs="unbounded"/> </xs:sequence> <xs:attribute name="id" type="xs:integer" use="required"/> <xs:attribute name="election_night_count"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="0"/> <xs:enumeration value="1"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="has_group_voting_square"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="Y"/> <xs:enumeration value="N"/> </xs:restriction> </xs:simpleType> </xs:attribute> </xs:complexType> </xs:element> </pre>
--------	--

Element ungrouped

A wrapper element for a “group” of one or more ungrouped candidates.

diagram	
children	candidate
used by	groups
attributes	
source	<pre> <xs:element name="ungrouped"> <xs:complexType> <xs:sequence> <xs:element ref="candidate" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element group_votes

ATL votes counted on election night for a group of candidates. SATL and RATL combined.

diagram							
children	fp_votes cc_votes						
used by	polling_place_council						
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>idref</td> <td>required</td> <td>The value of @href will match one of the values of @id of a candidate within the electoral event.</td> </tr> </tbody> </table>	Name	Use	Description	idref	required	The value of @href will match one of the values of @id of a candidate within the electoral event.
Name	Use	Description					
idref	required	The value of @href will match one of the values of @id of a candidate within the electoral event.					
source	<pre> <xs:element name="group_votes"> <xs:complexType> <xs:sequence> <xs:element ref="fp_votes" minOccurs="0"/> <xs:element ref="cc_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="idref" type="xs:anySimpleType" use="required"/> </xs:complexType> </xs:element> </pre>						

Element groupname

The name of the group as it appears on the ballot paper.

diagram	
children	None
used by	group
attributes	
source	<pre> <xs:element name="groupname"> <xs:complexType mixed="true"/> </xs:element> </pre>

Referendums and Polls

Element referendum

The top-level element for a referendum. Referendums and Polls are structurally identical, but separate elements are used so that the two types of event can be easily distinguished.

diagram							
children	text question						
used by	electoralarea						
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>deferred</td> <td>optional</td> <td>The referendum has been deferred</td> </tr> </tbody> </table>	Name	Use	Description	deferred	optional	The referendum has been deferred
Name	Use	Description					
deferred	optional	The referendum has been deferred					
source	<pre> <xs:element name="referendum"> <xs:complexType> <xs:sequence> <xs:element ref="text" minOccurs="0"/> <xs:element ref="question" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence> <xs:attribute name="deferred" default="0"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="0"/> <xs:enumeration value="1"/> </xs:restriction> </xs:simpleType> </xs:attribute> </xs:complexType> </xs:element> </pre>						

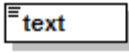
Element poll

The top-level element for a poll. Referendums and Polls are structurally identical, but separate elements are used so that the two types of event can be easily distinguished.

diagram							
children	text poll_info question						
used by	electoralarea						
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>deferred</td> <td>optional</td> <td>The poll has been deferred</td> </tr> </tbody> </table>	Name	Use	Description	deferred	optional	The poll has been deferred
Name	Use	Description					
deferred	optional	The poll has been deferred					
source	<pre> <xs:element name="poll"> <xs:complexType> <xs:sequence> <xs:element ref="text" minOccurs="0"/> <xs:element ref="poll_info" minOccurs="0"/> <xs:element ref="question" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence> <xs:attribute name="deferred" default="0"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="0"/> <xs:enumeration value="1"/> </xs:restriction> </xs:simpleType> </xs:attribute> </xs:complexType> </xs:element> </pre>						

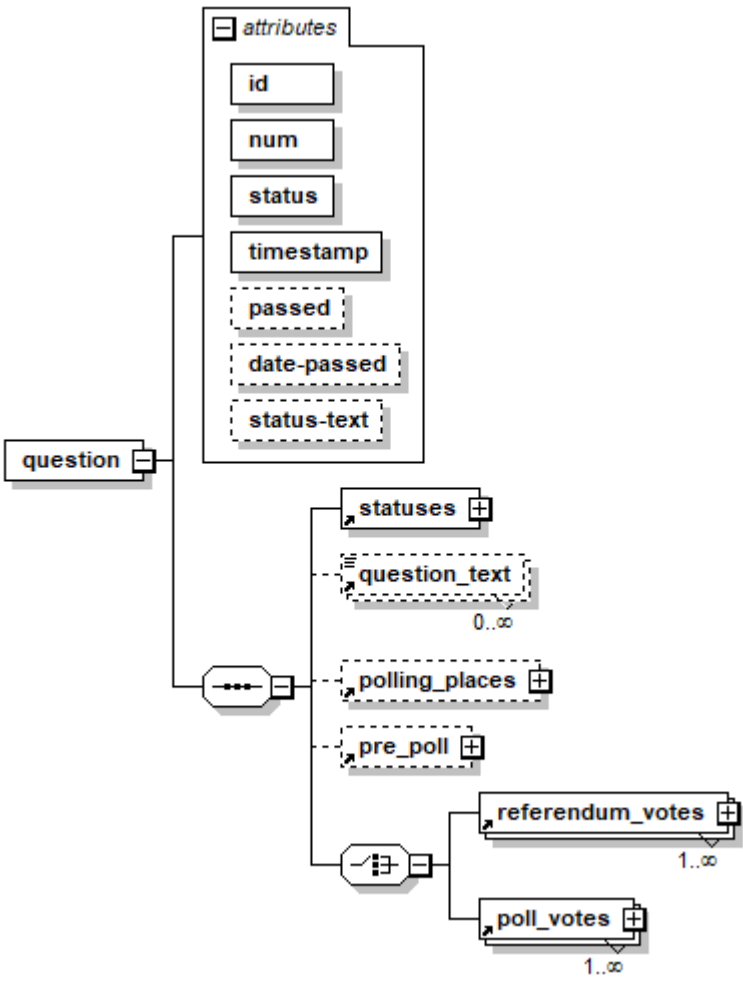
Element text

This is text that can be displayed for a specific Referendum or Poll. For example this might be a statement of the reason why the Referendum or poll has been cancelled.

diagram	
children	None
used by	electoralarea electoralward mayoral_election council_election referendum poll
attributes	
source	<pre><xs:element name="text"> <xs:complexType mixed="true"/> </xs:element></pre>

Element question

A numbered question in a referendum or poll.

diagram	
children	statuses question_text polling_places referendum_votes poll_votes
used by	referendum poll

attributes	Name	Use	Description
	id	required	The unique ID for the question
	num	required	The question number.
	status	required	Allowed values: No-Status Deferred CC-Enabled CC-Dec-Complete Declare-Election
	passed	optional	The question has been carried or passed "1". Or it has been rejected or not passed "0".
	date-passed	optional	If @passed="1", then the @date-passed is specified in the format "DD:MM:YYYY"
	timestamp	required	Date and time that this data was updated.
	status-text	optional	Can be used to define Question specific text for the given value of @status. This defines the text that will appear in VTR when the user rolls the mouse over the text "Status:". However, see also child element <text>
source	<pre> <xs:element name="question"> <xs:complexType> <xs:sequence> <xs:element ref="statuses"/> <xs:element ref="question_text" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="polling_places" minOccurs="0"/> <xs:element ref="pre_poll" minOccurs="0"/> <xs:choice> <xs:element ref="referendum_votes" maxOccurs="unbounded"/> <xs:element ref="poll_votes" maxOccurs="unbounded"/> </xs:choice> </xs:sequence> <xs:attribute name="id" type="xs:anySimpleType" use="required"/> <xs:attribute name="num" type="xs:NMTOKEN" use="required"/> <xs:attribute name="status" use="required"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="CC-Dec-Complete"/> <xs:enumeration value="CC-Enabled"/> <xs:enumeration value="No-Status"/> <xs:enumeration value="Final-for-Saturday"/> <xs:enumeration value="FC-Enabled"/> <xs:enumeration value="Declared-Election"/> <xs:enumeration value="Cancelled"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="timestamp" type="xs:dateTime" use="required"/> <xs:attribute name="passed" default="0"> </pre>		

	<pre> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="0"/> <xs:enumeration value="1"/> </xs:restriction> </xs:simpleType> </xs:attribute> <xs:attribute name="date-passed" type="xs:anySimpleType"/> <xs:attribute name="status-text" type="xs:anySimpleType"/> </xs:complexType> </xs:element> </pre>
--	--

Element question_text

The actual text of the referendum or poll question.

diagram	
children	None
used by	question
attributes	
source	<pre> <xs:element name="question_text"> <xs:complexType mixed="true"/> </xs:element> </pre>

Element poll_info

In a poll set in an arbitrary geographical area, the poll info provides the number of electors and the number of polling places.

diagram	
children	total_number_electors total_number_polling_places
used by	poll
attributes	
source	<pre> <xs:element name="poll_info"> <xs:complexType> <xs:sequence> <xs:element ref="total_number_electors"/> <xs:element ref="total_number_polling_places"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element referendum_votes

The final published results provides the totals of Yes/No and informal votes.

Returning Officers may be required to perform a recount. In this case a second set of results is recorded in the results data, and @recount is equal to "Y".

<p>diagram</p>							
<p>children</p>	<p>question_dec_votes election_dec_votes</p>						
<p>used by</p>	<p>question</p>						
<p>attributes</p>	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>recount</td> <td>optional</td> <td>"Y" if a second set of recount data is provided.</td> </tr> </tbody> </table>	Name	Use	Description	recount	optional	"Y" if a second set of recount data is provided.
Name	Use	Description					
recount	optional	"Y" if a second set of recount data is provided.					
<p>source</p>	<pre> <xs:element name="referendum_votes"> <xs:complexType> <xs:sequence> <xs:element ref="question_dec_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="election_dec_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="recount" default="N"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="N"/> <xs:enumeration value="Y"/> </xs:restriction> </xs:simpleType> </xs:attribute> </xs:complexType> </xs:element> </pre>						

Element `poll_votes`

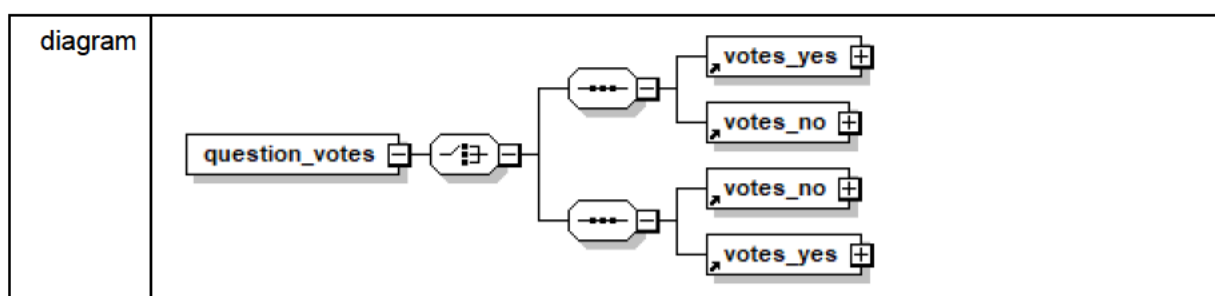
The final published results provide the totals of Yes/No and informal votes.

Returning Officers may be required to perform a recount. In this case a second set of results is recorded in the results data, and `@recount` is equal to "Y".

diagram							
children	question_dec_votes election_dec_votes						
used by	question						
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>recount</td> <td>optional</td> <td>"Y" if a second set of recount data is provided.</td> </tr> </tbody> </table>	Name	Use	Description	recount	optional	"Y" if a second set of recount data is provided.
Name	Use	Description					
recount	optional	"Y" if a second set of recount data is provided.					
source	<pre> <xs:element name="poll_votes"> <xs:complexType> <xs:sequence> <xs:element ref="question_dec_votes" minOccurs="0" maxOccurs="unbounded"/> <xs:element ref="election_dec_votes" minOccurs="0"/> </xs:sequence> <xs:attribute name="recount" default="N"> <xs:simpleType> <xs:restriction base="xs:NMTOKEN"> <xs:enumeration value="N"/> <xs:enumeration value="Y"/> </xs:restriction> </xs:simpleType> </xs:attribute> </xs:complexType> </xs:element> </pre>						

Element `question_votes`

A set of Yes/No votes. Both "Yes" votes and "No" votes must be specified. If the vote count is zero, then the data must explicitly show "0" votes `<cc_votes>0</cc_votes>`.



children	votes_yes votes_no
used by	polling_place_poll polling_place_ref
attributes	
source	<pre> <xs:element name="question_votes"> <xs:complexType> <xs:choice> <xs:sequence> <xs:element ref="votes_yes"/> <xs:element ref="votes_no"/> </xs:sequence> <xs:sequence> <xs:element ref="votes_no"/> <xs:element ref="votes_yes"/> </xs:sequence> </xs:choice> </xs:complexType> </xs:element> </pre>

Element votes_yes

The number of “Yes” votes in a referendum or poll. In the LGE 2012, FP votes for referenda and polls will not be reported on election night and element //fp_votes will not be used in the XML data.

diagram	
children	fp_votes cc_votes
used by	question_votes
attributes	
source	<pre> <xs:element name="votes_yes"> <xs:complexType> <xs:sequence> <xs:element ref="fp_votes" minOccurs="0"/> <xs:element ref="cc_votes" minOccurs="0"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element votes_no

The number of “No” votes in a referendum or poll. In the LGE 2012, FP votes for referenda and polls will not be reported on election night and element //fp_votes will not be used in the XML data.

diagram	
---------	--

children	fp_votes cc_votes
used by	question_votes
attributes	
source	<pre> <xs:element name="votes_no"> <xs:complexType> <xs:sequence> <xs:element ref="fp_votes" minOccurs="0"/> <xs:element ref="cc_votes" minOccurs="0"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element question_dec_votes

A set of Yes/No declaration votes.

diagram	
children	dec_votes_yes dec_votes_no
used by	referendum_votes
attributes	
source	<pre> <xs:element name="question_dec_votes"> <xs:complexType> <xs:sequence minOccurs="0" maxOccurs="unbounded"> <xs:element ref="dec_votes_yes"/> <xs:element ref="dec_votes_no"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

Element dec_votes_yes

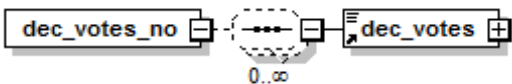
The number of “Yes” declaration votes in a referendum or poll.

diagram	
children	dec_votes
used by	question_dec_votes
attributes	

source	<pre> <xs:element name="dec_votes_yes"> <xs:complexType> <xs:sequence minOccurs="0" maxOccurs="unbounded"> <xs:element ref="dec_votes"/> </xs:sequence> </xs:complexType> </xs:element> </pre>
--------	--

Element dec_votes_no

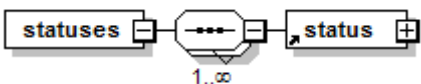
The number of “No” declaration votes in a referendum or poll.

diagram	
children	dec_votes
used by	question_dec_votes
attributes	
source	<pre> <xs:element name="dec_votes_no"> <xs:complexType> <xs:sequence minOccurs="0" maxOccurs="unbounded"> <xs:element ref="dec_votes"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

Status List

Element statuses

This element is a wrapper for a list of status values that lists each status, and the date/time at which the status was achieved. That is, it records the relevant timestamp when a status was *first* set. Note that subsequent XML data feeds may show the same status value for a contest but the timestamp may be later than that shown in the status list.

diagram	
children	status
used by	mayoral_election council_election question
attributes	

source	<pre> <xs:element name="statuses"> <xs:complexType> <xs:sequence maxOccurs="unbounded"> <xs:element ref="status"/> </xs:sequence> </xs:complexType> </xs:element> </pre>
---------------	--

Element status

This element records the relevant timestamp when a a status was *first* set. Note that subsequent XML data feeds may show the same status value for a contest but the timestamp may be later than that shown in the status list.

The status value will be one of the allowed values for a mayoral or council contest, or for a referendum or poll question.

diagram															
children	None														
used by	statuses														
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Use</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>timestamp</td> <td>required</td> <td>Date and time the status value was first achieved.</td> </tr> <tr> <td>value</td> <td>required</td> <td>Any allowable status value for mayoral or council election, or for referendum/poll question</td> </tr> <tr> <td>status-text</td> <td>optional</td> <td>Can be used specific text for the given value of @status. This defines the text that will appear in VTR when the user rolls the mouse over the displayed status value.</td> </tr> </tbody> </table>			Name	Use	Description	timestamp	required	Date and time the status value was first achieved.	value	required	Any allowable status value for mayoral or council election, or for referendum/poll question	status-text	optional	Can be used specific text for the given value of @status. This defines the text that will appear in VTR when the user rolls the mouse over the displayed status value.
Name	Use	Description													
timestamp	required	Date and time the status value was first achieved.													
value	required	Any allowable status value for mayoral or council election, or for referendum/poll question													
status-text	optional	Can be used specific text for the given value of @status. This defines the text that will appear in VTR when the user rolls the mouse over the displayed status value.													
source	<pre> <xs:element name="status"> <xs:complexType> <xs:attribute name="value" type="xs:anySimpleType" use="required"/> <xs:attribute name="timestamp" type="xs:anySimpleType" use="required"/> <xs:attribute name="status-text" type="xs:anySimpleType"/> </xs:complexType> </xs:element> </pre>														