5 EML Core Components

387
388
The core schema contains elements and data types that are used throughout the e-voting schemas.

389
390
To help message schema diagrams fit on the page, these elements and data types are not expanded each time they appear in other diagrams.

391
The following schema components are defined in the EML core.

Elements	Complex Data Types	Simple Data Types
Accepted	AffiliationIdentifierStructure	ConfirmationReferenceType
Affiliation	AffiliationStructure	CountingAlgorithmType
AffiliationIdentifier	AgentIdentifierStructure	DateType
Agent	AgentStructure	EmailType
AgentIdentifier	AreaStructure	ErrorCodeType
Area	AuditInformationStructure	GenderType
AuditInformation	AuthorityIdentifierStructure	LanguageType
AuthorityIdentifier	BallotIdentifierRangeStructure	MessageTypeType
BallotIdentifier	BallotIdentifierStructure	SealUsageType
BallotIdentifierRange	CandidateIdentifierStructure	ShortCodeType
Candidate	CandidateStructure	TelephoneNumberType
CandidateIdentifier	ComplexDateRangeStructure	VotingChannelType
ContactDetails	ContactDetailsStructure	VotingMethodType
ContestIdentifier	ContestIdentifierStructure	VotingValueType
CountingAlgorithm	DocumentIdentifierStructure	YesNoType
DocumentIdentifier	ElectionGroupStructure	
ElectionIdentifier	ElectionIdentifierStructure	
ElectionStatement	EmailStructure	
EventIdentifier	EMLStructure	
EventQualifier	EventIdentifierStructure	
Gender	EventQualifierStructure	
Logo	IncomingGenericCommunicationStructure	
ManagingAuthority	InternalGenericCommunicationStructure	
MaxVotes	LogoStructure	
MessageType	ManagingAuthorityStructure	
MinVotes	MessagesStructure	
NominatingOfficer	NominatingOfficerStructure	
NumberInSequence	OutgoingGenericCommunicationStructure	
NumberOfPositions	PeriodStructure	
Period	PictureDataStructure	
PersonName	PollingDistrictStructure	
PollingDistrict	PollingPlaceStructure	

Elements	Complex Data Types	Simple Data Types
PollingPlace	PositionStructure	
Position	ProcessingUnitStructure	
PreviousElectoralAddress	ProposalIdentifierStructure	
Profile	ProposalStructure	
Proposal	ProposerStructure	
ProposalIdentifier	ProxyStructure	
Proposer	ReferendumOptionIdentifierStructure	
Proxy	ReportingUnitIdentifierStructure	
ReferendumOptionIdentifier	ResponsibleOfficerStructure	
ReportingUnitIdentifier	ScrutinyRequirementStructure	
ResponsibleOfficer	SealStructure	
ScrutinyRequirement	SimpleDateRangeStructure	
Seal	TelephoneStructure	
SequenceNumber	VoterIdentificationStructure	
TransactionId	VoterInformationStructure	
VoterName	VTokenStructure	
VotingChannel	VTokenQualifiedStructure	
VotingMethod		
VToken		
VTokenQualified		

392   ## 5.1 Simple Data Types

393   The simple data types are included here with their base data types and any restrictions applied.

394   ### 5.1.1 ConfirmationReferenceType

395   `xs:token`.

396   The reference generated once the confirmation of a vote has been completed.

397   ### 5.1.2 CountingAlgorithmType

398   `xs:token`

399   The method of counting used for more complex forms of election.

400   ### 5.1.3 DateType

401   Union of `xs:date` and `xs:dateTime`

402   There are several possible dates associated with an election. Some of these can be either just a
403   date or have a time associated with them. These can use this data type.

404   ### 5.1.4 EmailType

405   `xs:token` with restrictions.

406   Restrictions: `xs:maxLength`: 129
407   `xs:pattern`: [^@]+@[^@]+

408 This type is a simple definition of an email address, pending a more complete description that is
409 widely accepted in industry and government. It allows any characters except the @ symbol,
410 followed by an @ symbol and another set of characters excluding this symbol.

411 5.1.5 ErrorCodeType

412 `xs:token`

413 One of a pre-defined set of error codes as described in the section "Error Messages".

414 5.1.6 GenderType

415 `xs:token` with restrictions.

416 Restrictions: `xs:enumeration`: male, female, unknown

417 The gender of a voter or candidate. Options are male, female or unknown (unknown is not
418 allowed in all contexts).

419 5.1.7 LanguageType

420 `xs:language`

421 Declaration of the type of language used in the election.

422 5.1.8 MessageTypeType

423 `xs:NMTOKEN`

424 This is the alphanumeric type of the message (e.g. 440 or 350a). This may be required for audit
425 purposes.

426 5.1.9 SealUsageType

427 `xs:NMTOKEN` with restrictions.

428 Restrictions: `xs:enumeration`: receiver, sender

429 Indicates whether a device logging a seal was the sender or receiver of the seal.

430 5.1.10 ShortCodeType

431 `xs:NMTOKEN`

432 This identifies an aspect of the election (such as a contest or candidate) when voting using SMS
433 or other voting mechanisms where a short identifier is required.

434 5.1.11 TelephoneNumberType

435 `xs:token` with restrictions.

436 Restrictions: `xs:maxLength`: 35
437   xs:minLength: 1
438   xs:pattern: \+?[0-9\(\)\-\s]{1,35}

439 Since this must allow for various styles of international telephone number, the pattern has been
440 kept simple. This allows an optional plus sign, then between 1 and 35 characters with a
441 combination of digits, brackets, the dash symbol and white space. If a more complete definition
442 becomes widely accepted in industry and government, this will be adopted.

443 5.1.12 VotingChannelType

444 `xs:token` with restrictions.

445 Restrictions: `xs:enumeration`: SMS, WAP, digitalTV, internet, kiosk, polling, postal,
446 telephone, other

447 This type exists to hold the possible enumerations for the channel through which a vote is cast.

448 SMS is the Short Message Service (text message). WAP is the Wireless Access Protocol.

449 If `other` is used, it is assumed that those managing the election will have a common
450 understanding of the channel in use.

5.1.13 VotingMethodType

452 `xs:token` with restrictions.

453 Restrictions: `xs:enumeration`: AMS, FPP, OPV, SPV, STV, approval, block, partylist,
454 supplementaryvote, other

455 The `VotingMethod` type holds the enumerated values for the type of election (such as *first past*
456 *the post* or *single transferable vote*). The meanings of the acronyms are:

457 • `AMS` – Additional Member System

458 • `FPP` - First Past the Post

459 • `OPV` - Optional Preferential Voting

460 • `SPV` - Single Preferential Vote

461 • `STV` - Single Transferable Vote

5.1.14 VotingValueType

463 `xs:positiveInteger`.

464 Indicates a value assigned when voting for a candidate or referendum option. This might be a
465 weight or preference order depending on the election type.

5.1.15 YesNoType

467 `xs:token` with restrictions.

468 Restrictions: `xs:enumeration`: no, yes

469 This is a simple enumeration of `yes` and `no` and is used for elements and attributes that can only
470 take these binary values.

5.2 Complex Data Types

472 The choice between defining an element or a data type for a reusable message component is a
473 significant design issue. It is widely accepted as good practice to use element declarations when
474 there is good reason to always refer to an element by the same name and there is no expectation
475 of a need to derive new definitions. In all other cases, data type declarations are preferable. The
476 term *schema component* is used to refer to elements and data types collectively.

477 When defining a complete mark-up language, limiting the use of elements and types can restrict
478 further development of the language. For that reason, both data types and elements are defined
479 in EML. Only where an element is an example of a primitive or derived data type defined in XML
480 Schema part 2 is no explicit data type defined within EML.

481 In use, it is expected that, for example:

482 • A voting token will always have an element name `VToken` and so will use the element
483 name;

| 484 | • A logo or a map have similar definitions, so both use the `PictureDataStructure`. |
485	There is no `PictureData` element.
486	• Within voter identification, some elements will usually need to be made mandatory and so
487	a schema will specify a new element based on the `VoterIdentificationStructure`
488	data type.

489   5.2.1 AffiliationIdentifierStructure

490
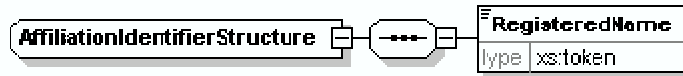
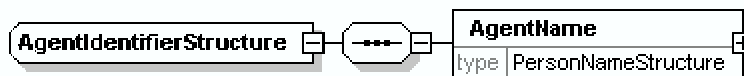| Element	Attribute	Type	Use	Comment
AffiliationIdentifierStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
	ShortCode	ShortCodeType	optional	
	ExpectedConfirmationReference	ConfirmationReferenceType	optional	

491   This data type is used to identify an affiliation, such as a political party. The identifier indicates the
492   official name and ID of the organization. It supports use of a short code for voting systems such
493   as SMS, and an expected confirmation reference for security systems that require this.

494   5.2.2 AffiliationStructure

495
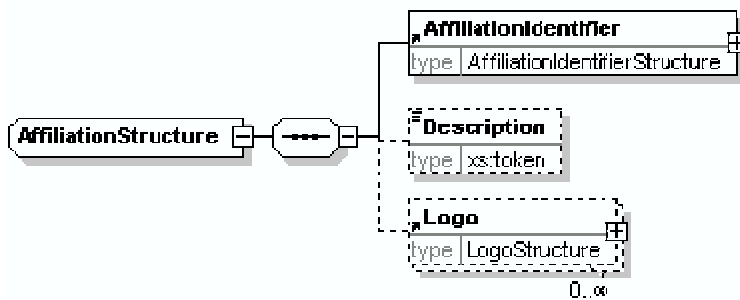
496   `AffiliationStructure` data type indicates membership of some organization such as a
497   political party. The description will normally be used to indicate the name usually associated with
498   the organization, and so is the value that will usually be shown on a ballot. An organization may
499   indicate several logos, each with a rôle. For example, one rôle might indicate that the logo should
500   be used on a ballot paper. Each logo can be identified by a URL or sent as a Base64 encoded
501   binary value. In the latter case, the format of the logo (BMP, TIFF, PNG, GIF or JPEG) must be
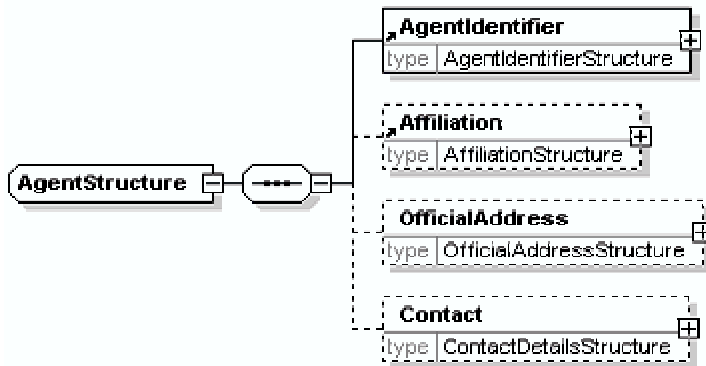502   indicated.

503   5.2.3 AgentIdentifierStructure

504

505

Element	Attribute	Type	Use	Comment
AgentIdenttifierStructure	Id	xs:NMTOKEN	optional	

| | DisplayOrder | xs:positiveInteger | optional | |

506 The agent identifier contains a name and ID. The data type for the name is localized using the
507 EML externals schema.

5.2.4 AgentStructure

509

Element	Attribute	Type	Use	Comment
AgentStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
	Role	xs:token	optional	

510 A candidate in an election can have one or more agents, each agent having a specific rôle,
511 identified by the Role attribute. For example, an agent may be allowed access to the count, but
512 not to amend details of the candidate.

513 The agent has an identifier, comprising a name and ID, and an affiliation. He or she also has an
514 official address and a standard set of contact details.

5.2.5 AreaStructure

516 The AreaStructure is an extension of xs:token to add the following attributes:

Element	Attribute	Type	Use	Comment
AreaStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
	Type	xs:token	optional	

517 This data type is used to define elements defining the geographical area covered by a contest.
518 The Type attribute is used to indicate the type of area, such as "county".

519

5.2.6 AuditInformationStructure
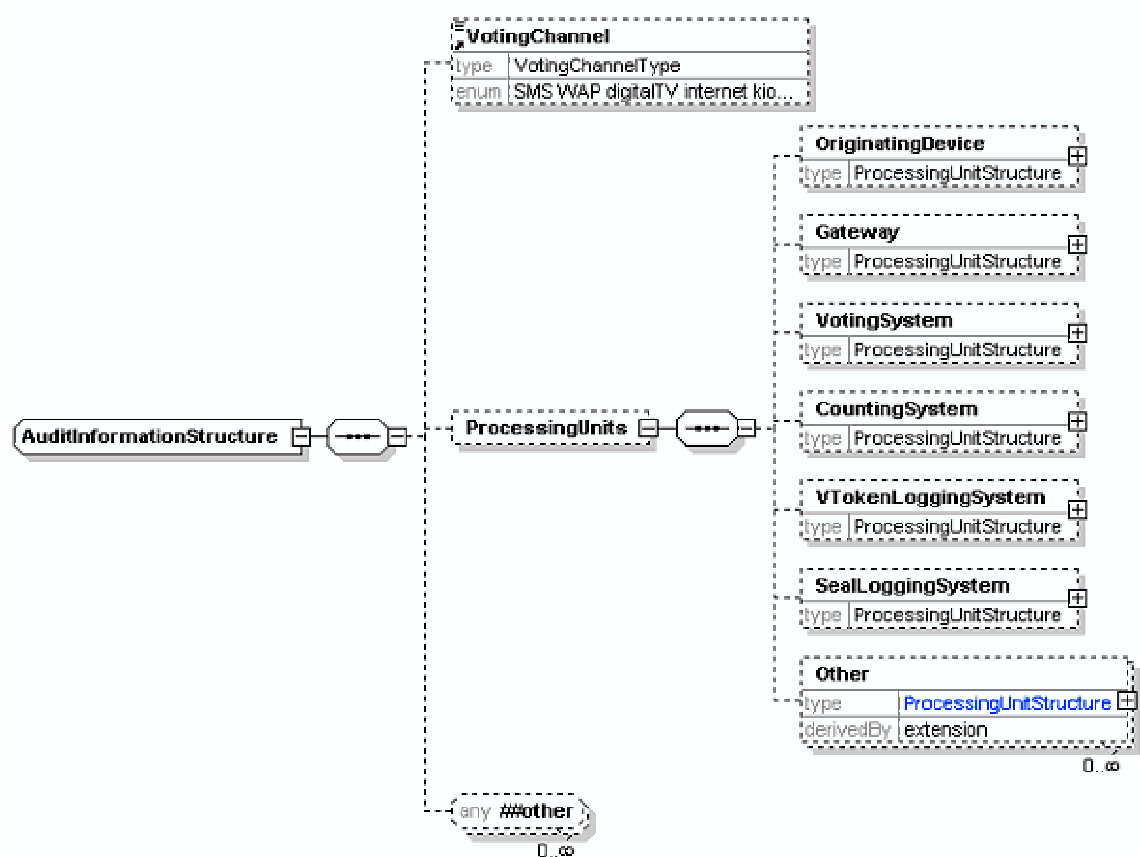
520

Element	Attribute	Type	Use	Comment
Other	Role	xs:token (restricted)	required	Standard attribute for a ProcessingUnitStructure
	Type	xs:token	required	Additional attribute for this element

521 The AuditInformationStructure is used to define an element to provide information for audit
522 purposes. It allows the voting channel in use to be described, with the identities of those devices
523 that have participated in the message being sent. Each device has an attribute to describe its rôle
524 (see ProcessingUnitStructure).

525 Where a device does not fit any of the categories here, it can be described as Other with the
526 addition of a Type attribute.

5.2.7 AuthorityIdentifierStructure

528 The AuthorityIdentifierStructure is an extension of xs:token to add the following
529 attributes:

Element	Attribute	Type	Use	Comment
AuthorityIdentifierStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	

530 This data type defines information to identify an election authority. This may include a system ID
531 and text description.

5.2.8 BallotIdentifierStructure
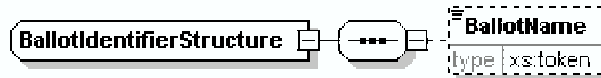
Element	Attribute	Type	Use	Comment
BallotIdentifierStructure	Id	xs:NMTOKEN	required	
	DisplayOrder	xs:positiveInteger	optional	

This data type is used to define an element that is an identifier for a ballot. This will usually use the Id attribute as the identifier, but might use a name to indicate a set of identical ballots. Elements using this data type will usually only be used for paper ballots.

5.2.9 BallotIdentifierRangeStructure
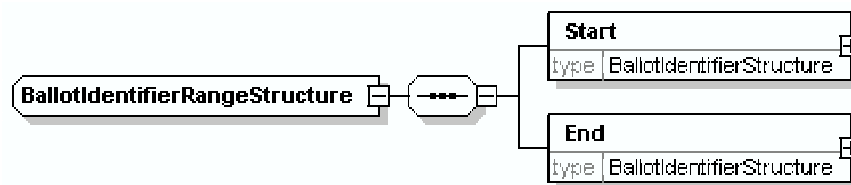
Element	Attribute	Type	Use	Comment
BallotIdentifierRangeStructure	Colour	xs:token	optional	

This data type is used to define an element that identifies a range of ballots. This might be used, for example, to assign ranges of ballot identifiers to different reporting units for a contest. It is unlikely that the ballot name would be used when defining range, the Id attribute being used instead. Elements using this data type will usually only be used for paper ballots.

5.2.10 CandidateIdentifierRangeStructure
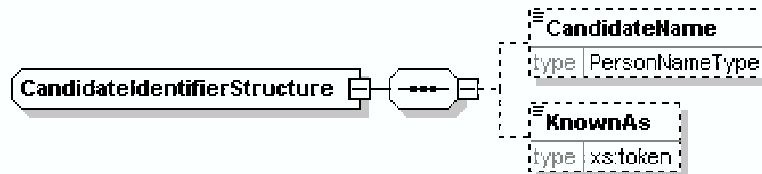
Element	Attribute	Type	Use	Comment
CandidateIdentifierStructure	Id	xs:NMTOKEN	required	
	DisplayOrder	xs:positiveInteger	optional	
	ShortCode	ShortCodeType	optional	
	ExpectedConfirmationReference	ConfirmationReferenceType	optional	

556   The candidate identifier indicates a system ID for the candidate and the candidate's name as it
557   will appear in a ballot. Sometimes an additional line is required on the ballot to help identify the
558   candidate. This will use the `KnownAs` element of the candidate identifier. A short code can also be
559   included, either for SMS voting or where the security mechanism in place requires it. An
560   `ExpectedConfirmationReference` attribute also allows for security mechanisms where the
561   confirmation reference may be different for each combination of voter and candidate.

562   **5.2.11 CandidateStructure**

563
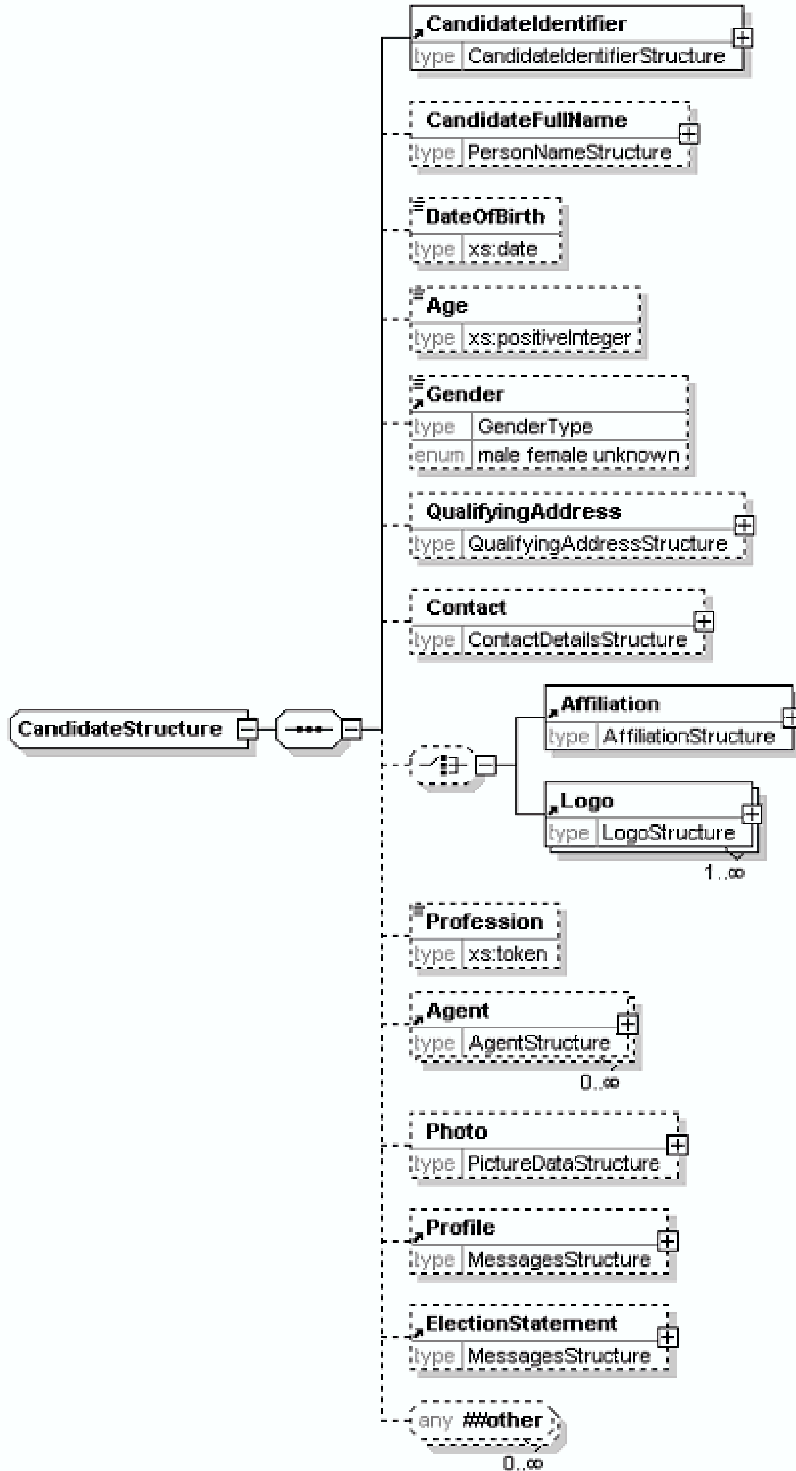
Element	Attribute	Type	Use	Comment
CandidateStructure	Independent	YesNoType	optional	
	DisplayOrder	xs:positiveInteger	optional	

564 The candidate description includes all the information required about the candidate. In different
565 messages, the amount of information is reduced, either by restricting the information in EML or as
566 part of a localization.

567 The candidate has an identifier. The full name of the candidate may also be provided, and
568 whether the candidate is an independent. This is supplied as an attribute rather than affiliation as
569 certain election types treat independents differently from other candidates, even though they may
570 define an affiliation.

571 The candidate profile describes the candidate. The election statement describes the opinions of
572 the candidate. Optionally, a photo may be included, either as a link or as Base64 encoded binary.

### 573 5.2.12 ComplexDateRangeStructure
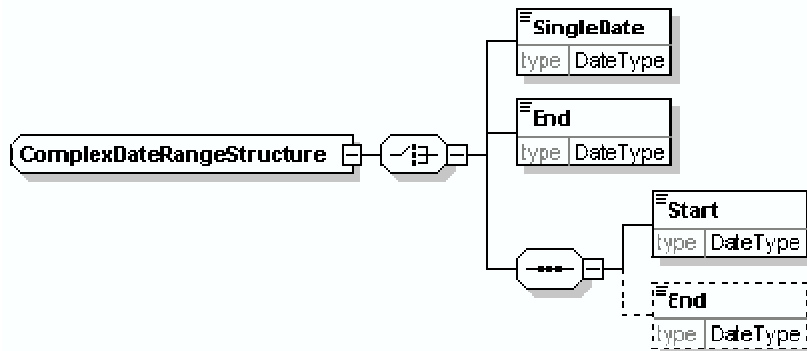
574

Element	Attribute	Type	Use	Comment
ComplexDateRangeStructure	Type	xs:token	required	

575 This data type is used to describe ranges of dates or dates and times. Each date can be a single
576 date, a start date, an end date or include both start and end dates.

577 The Type attribute is used to indicate the purpose of the date (e.g. "deadline for nominations"). It
578 is likely that this will be removed before release of EML version 4 and applied to elements instead
579 as an extension of this data type.

580

5.2.13 ContactDetailsStructure
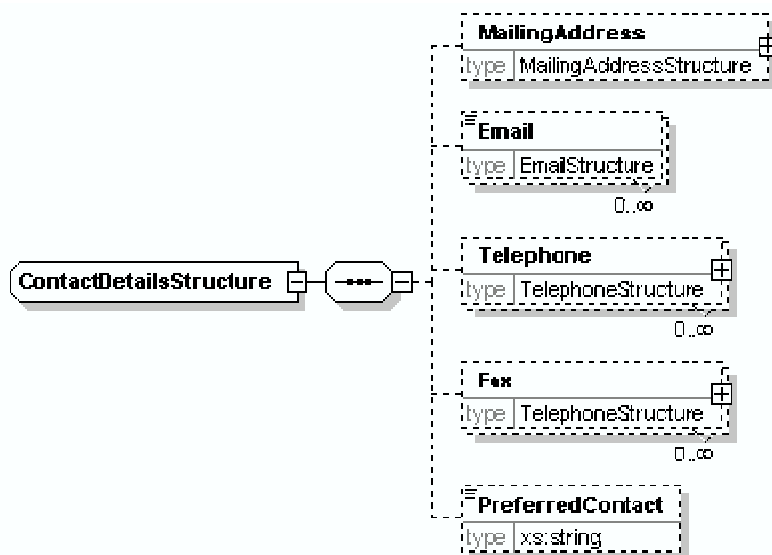
581

Element	Attribute	Type	Use	Comment
ContactDetailsStructure	DisplayOrder	xs:positiveInteger	optional	

582 This data type is used in many places throughout the EML schemas. The mailing address uses
583 whatever format is defined in the EML externals schema document. Where several addresses or
584 numbers can be given (for example, email addresses), there is a facility to indicate whichever is
585 preferred. The overall preferred method of contact can also be provided by placing an XPath to
586 the preferred method in the `PreferredContact` element.

5.2.14 ContestIdentifierStructure

587
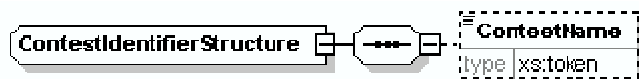
588

Element	Attribute	Type	Use	Comment
ContestIdentifierStructure	Id	xs:NMTOKEN	required	
	DisplayOrder	xs:positiveInteger	optional	
	ShortCode	ShortCodeType	optional	

589 This data type is used to define an element that is an identifier for a contest. It holds a name and
590 ID. A short code can also be included, for example, for SMS voting.

5.2.15 DocumentIdentifierStructure

591

592 The `DocumentIdentifierStructure` is an extension of `xs:token` to add the following
593 attribute:

Element	Attribute	Type	Use	Comment
DocumentIdentifierStructure	Href	xs:anyURI	required	

594

595 This allows identification of external documents relating to an event, election or contest. The
596 document can have a name and URL.

5.2.16 ElectionGroupStructure

598 The `ElectionGroupStructure` is an extension of `xs:token` to add the following attribute:

Element	Attribute	Type	Use	Comment
DocumentIdentifierStructure	Id	xs:token	required	

599 The election group is used to group a number of elections together. This could be required, for
600 example, under the additional member system, where two elections are held, the result of one
601 influencing the result of the other. It could also be used at a company AGM, where proposals
602 might be grouped for display purposes.

5.2.17 ElectionIdentifierStructure
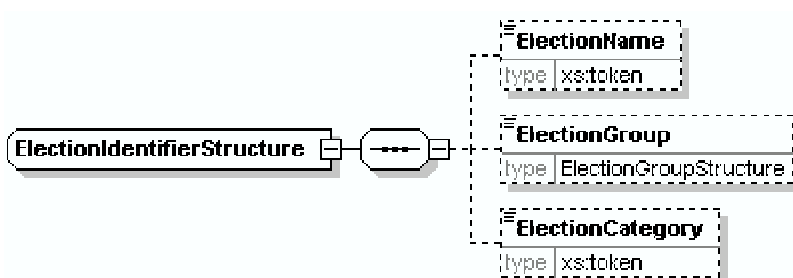
604

Element	Attribute	Type	Use	Comment
ElectionIdentifierStructure	Id	xs:NMTOKEN	required	
	DisplayOrder	xs:positiveInteger	optional	
	ShortCode	ShortCodeType	optional	

605 The election identifier is used wherever the election needs to be specified. There is an `Id`
606 attribute, which can often be used on its own to identify the election. In other cases, particularly
607 where the content of a message is to be displayed, the election name can also be provided. The
608 election group is used to group a number of elections together as described above.

609 The election category is used in messages where several elections are included in the message,
610 but may be treated differently under localisation rules. Each election that requires different
611 treatment will be given a category unique within that election event, allowing a Schematron
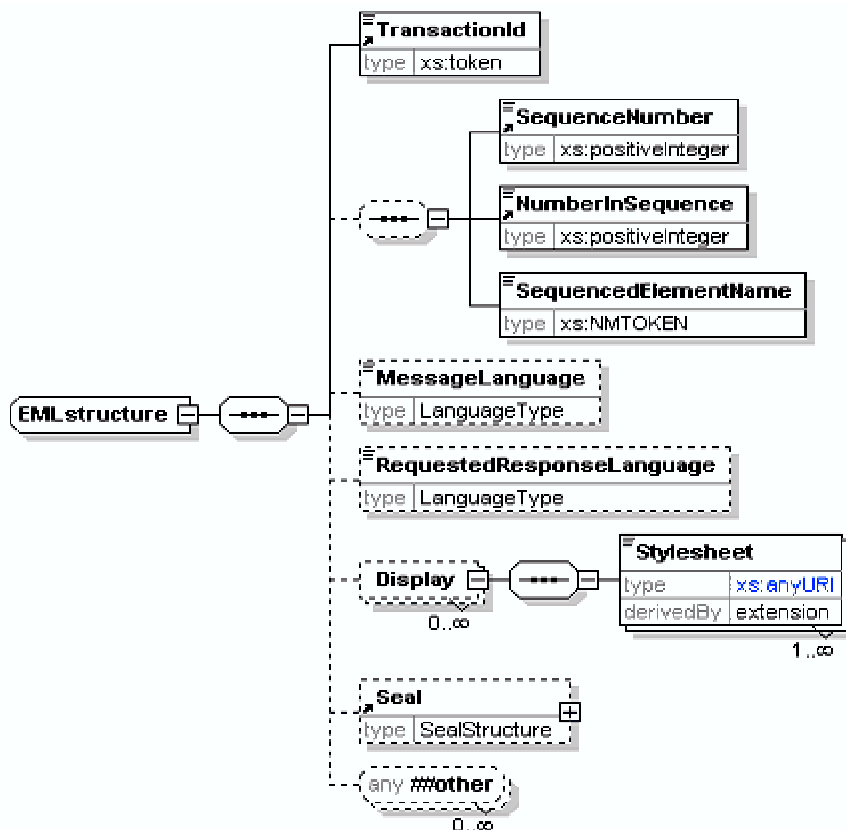612 processor to distinguish between the elections.

5.2.18 EmailStructure

614   The `EmailStructure` is an extension of the `EmailType` to add the following attribute:

Element	Attribute	Type	Use	Comment
EmailStructure	Preferred	YesNoType	optional	

615   The `Preferred` attribute is used to distinguish which of several email addresses to use.

5.2.19 EMLstructure

616

617

Element	Attribute	Type	Use	Comment
EMLstructure	Id	MessageTypeType	required	
	SchemaVersion	xs:NMTOKEN	requried	
	ShortCode	ShortCodeType	optional	
Stylesheet	Type	xs:token	required	

618   The EML element defined by this data type forms the root element of all EML documents. The
619   transaction ID is used to group messages together, for example, when they are split using the
620   message splitting mechanism. This mechanism is implemented using the next three elements.
621   The optional message language indicates the language of the message using ISO 639 three
622   letter language codes, while the requested response language can be used to indicate the
623   preferred language for a response. This element is used in messages from the voter or candidate
624   to the election organizers.

625   The display element allows the definition of stylesheets to display the message. Multiple
626   stylesheets can be declared. When displaying on the web, the first is likely to be an XSLT

627 stylesheet, while the second might describe a CSS stylesheet to be incorporated as well. The
628 `Type` attribute of the `Stylesheet` element should contain a media types as defined in RFC 2046
629 Pt 2 [1] using the list of media types defined by IANA [2], for example, text/xsl. The final element
630 defined is the seal, which is used to seal the complete message.

5.2.20 EventIdentifierStructure
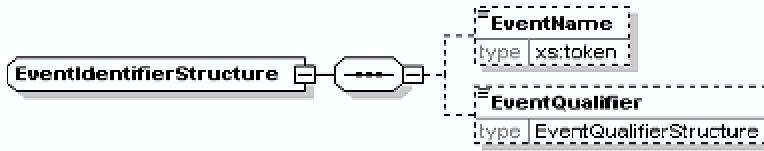
632

Element	Attribute	Type	Use	Comment
EventIdentifierStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	

633 The event identifier is used wherever the election event needs to be specified. There is an `Id`
634 attribute, which can often be used on its own to identify the event. In other cases, particularly
635 where the content of a message is to be displayed, the event name can also be provided. The
636 event qualifier is used to further identify the event.

5.2.21 EventQualifierStructure

638 The `EventQualifierStructure` is an extension of `xs:token` to add the following attribute:

Element	Attribute	Type	Use	Comment
EventQualifierStructure	Id	xs:NMTOKEN	optional	

639 The event qualifier is used to further identify the event. For example, there might be "County
640 Elections" covering an entire country, but the events are organized at a county level, so the event
641 qualifier would identify the county.

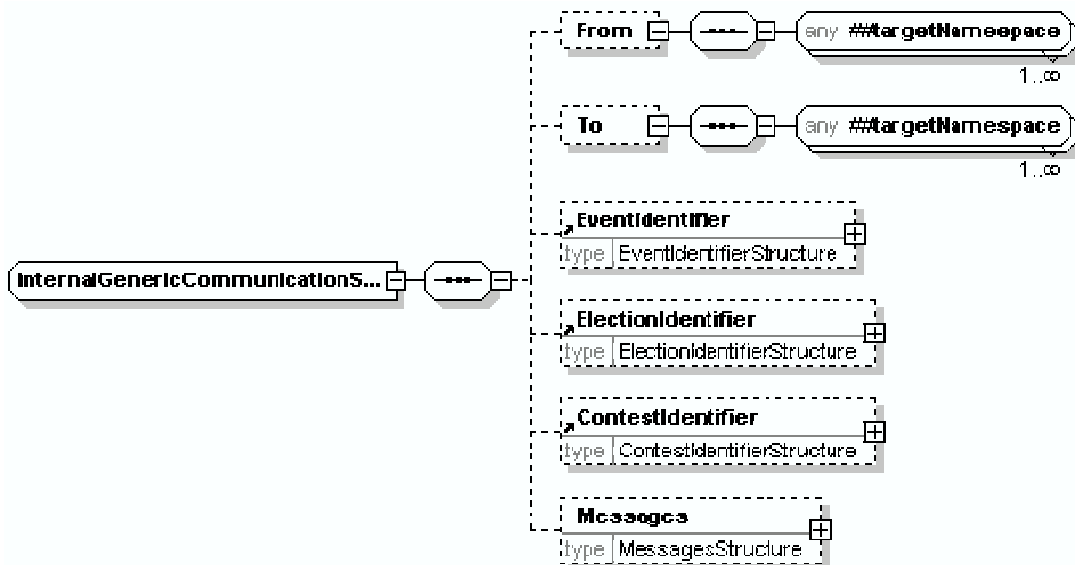# 5.2.22 IncomingGenericCommunicationStructure
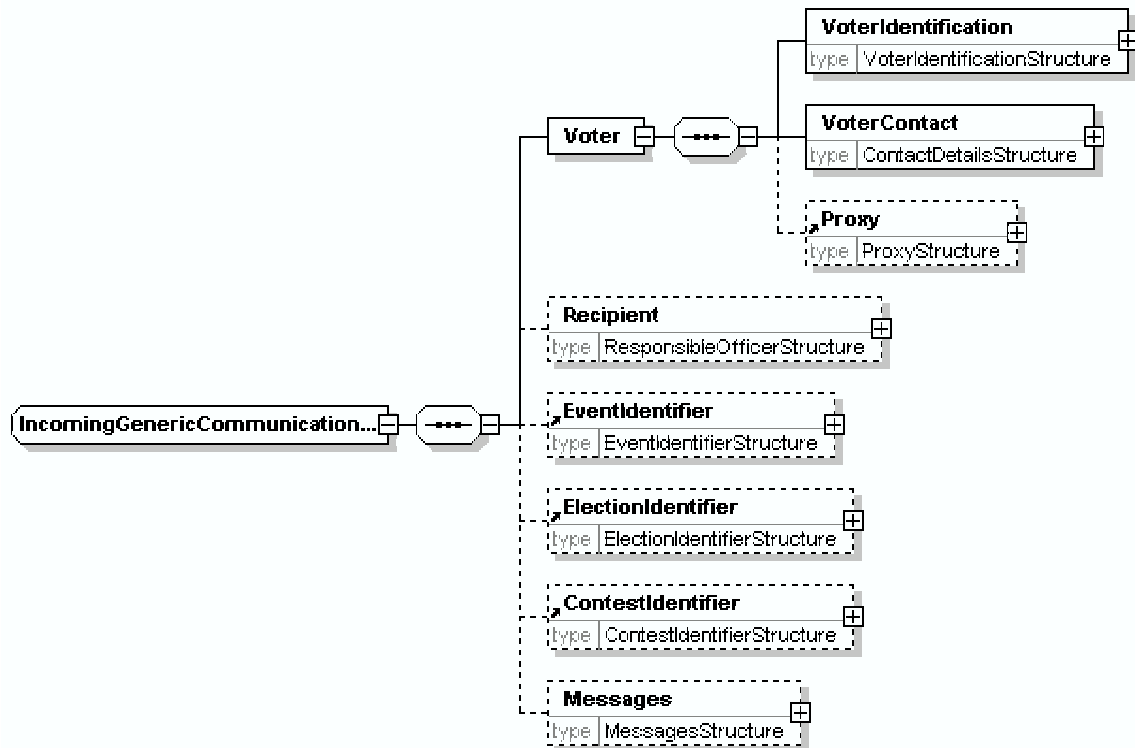
643
644   This data type provides a common structure for incoming communications. Individual message
645   types, such as that used for selecting a preferred voting channel (schema 360b) are based on
646   extensions of this type.

647 ## 5.2.23 InternalGenericCommunicationStructure

648

649 This data type provides a common structure for communications between entities involved in the
650 organization of an election. Individual message types are based on extensions of this type. The
651 sender and recipient can use any elements defined within EML.

652 ## 5.2.24 LogoStructure

653 The `LogoStructure` is an extension of the `PictureDataStructure` to add one attribute:

Element	Attribute	Type	Use	Comment
LogoStructure	Id	xs:NMTOKEN	optional	Standard attribute for a PictureDataStructure
	DisplayOrder	xs:positiveInteger	optional	Standard attribute for a PictureDataStructure
	Role	xs:token	optional	Additional attribute for this element

654 This element extends the picture data structure by adding an attribute to define the rôle of the
655 logo. This can be used to indicate the purpose of the logo (for example, it is to appear on a
656 ballot).

657   **5.2.25 ManagingAuthorityStructure**
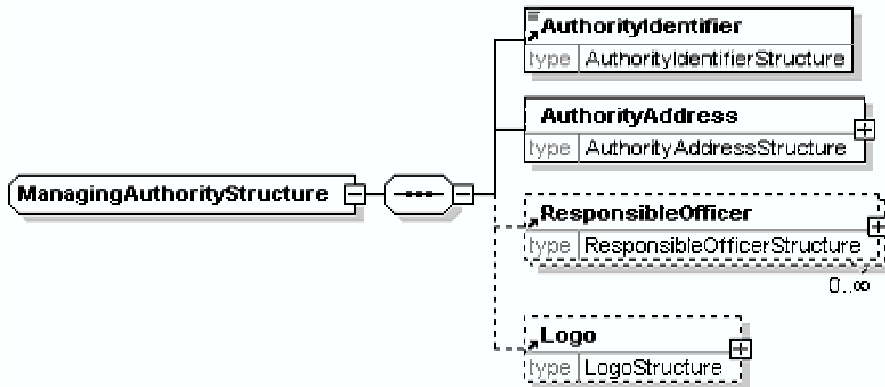
658

659   The managing authority is the body responsible for an election event, election, contest or
660   reporting unit. In most cases, not all of these will be required, but sometimes more than one is
661   necessary. For example, an election using the additional member system might be organized on
662   a regional basis, whilst local authorities organise their local election events. In this case, the
663   region becomes the managing authority for the contest, whilst the local authority is the managing
664   authority for the event. There will also be an authority responsible for the overall conduct of the
665   election, although this information might not be required.

666   The managing authority indicates the authority name, address, Id, any logo that might be required
667   for display during the election and a list of responsible officers.

668   **5.2.26 MessageStructure**
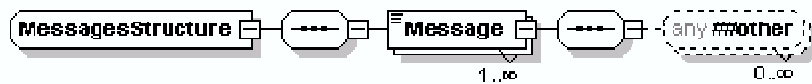
669

Element	Attribute	Type	Use	Comment
MessagesStructure	DisplayOrder	xs:positiveInteger	optional	
Message	Format	xs:topken	optional	
	Type	xs:token	optional	
	Lang	LanguageType	optional	

670   The Message element is of 'mixed' type, so can have both text and element content. The
671   intention is that it should have one or the other. The Message element has three attributes: Lang
672   is used to indicate the language of the message using ISO 639 three letter language codes,
673   Format indicates the format of element content using the media types definition from RFC 2046
674   Pt 2 [1] and the list of media types defined by IANA [2], for example, text/html, and Type indicates
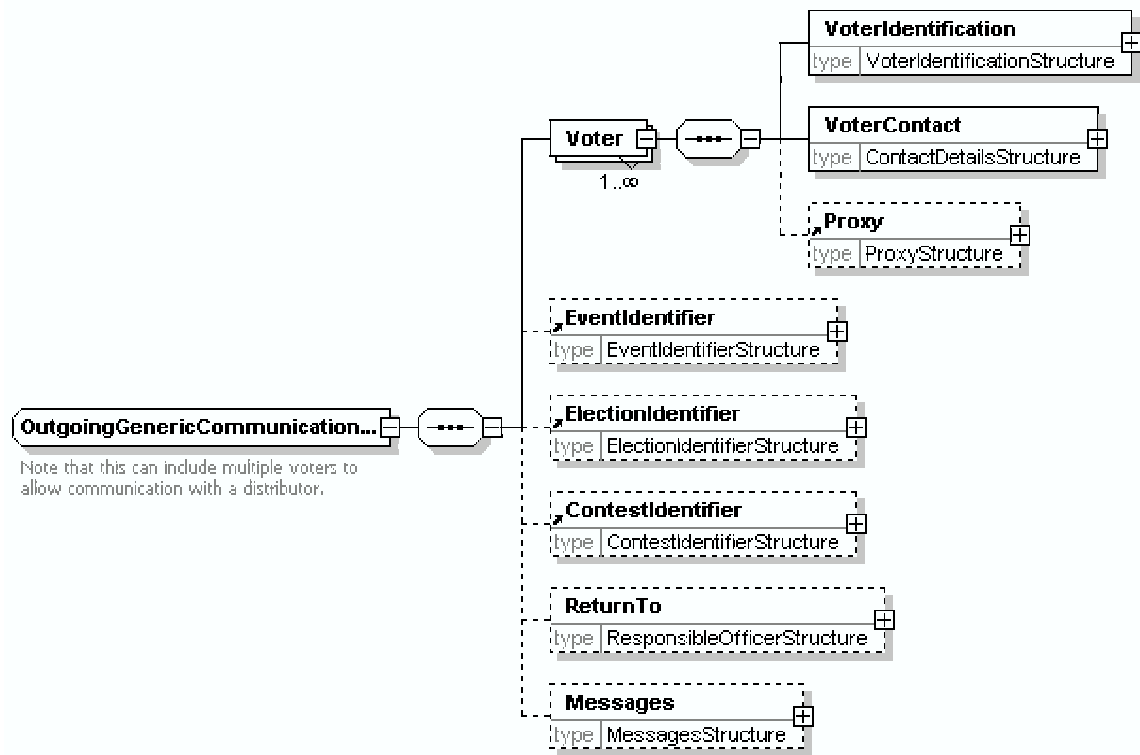675   the purpose of the message.

676   **5.2.27 NominatingOfficerStructure**
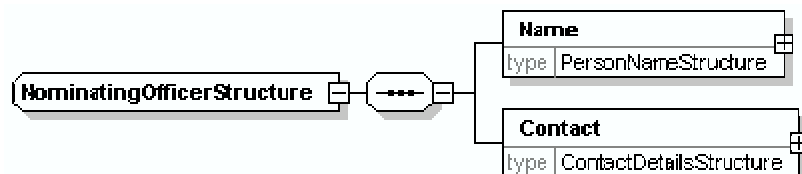
677

678   The nominating officer is the person nominating a party in an election run under, for example, the
679   party list system. The data type includes a name and contact information.

**5.2.28 OutgoingGenericCommunicationStructure**

681

682 This data type provides a common structure for communications from electoral service organisers
683 to voters. Multiple voters can be identified to allow printing of messages. Individual message
684 types, such as that used for offering voting channel options (360a) are based on extensions of
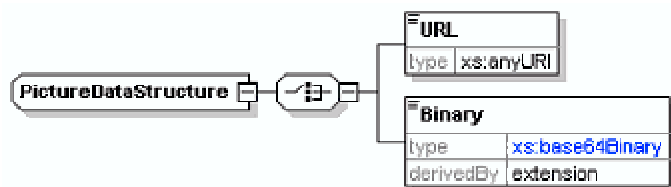685 this type.

686 ## 5.2.29 PeriodStructure
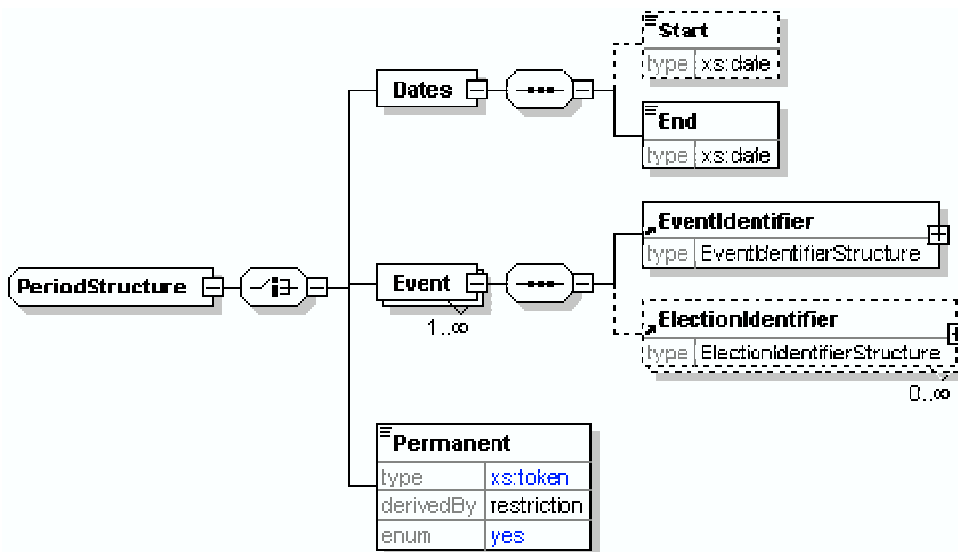
687

688 This element can be used when appointing a proxy or registering to vote using a specific channel
689 (e.g. postal). It allows this registration to be for a period of time, for specific election events (and
690 possibly elections within those events) or permanently.
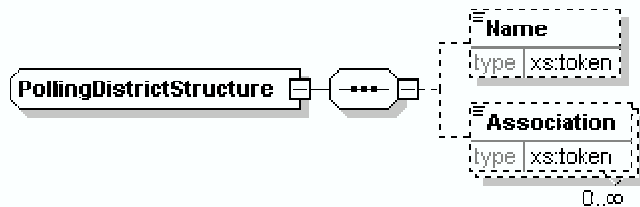
691 ## 5.2.30 PictureDataStructure

692

Element	Attribute	Type	Use	Comment
PictureDataStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
Binary	Format	xs:NMTOKEN (restricted)	required	

693 Where a picture (logo, map, photo) is provided, it may be given as either a link or as Base64
694 encoded binary data. In the latter case, the format of the logo (bmp, gif, jpeg, png or tiff) must be
695 indicated using the `Format` attribute of the `Binary` element.
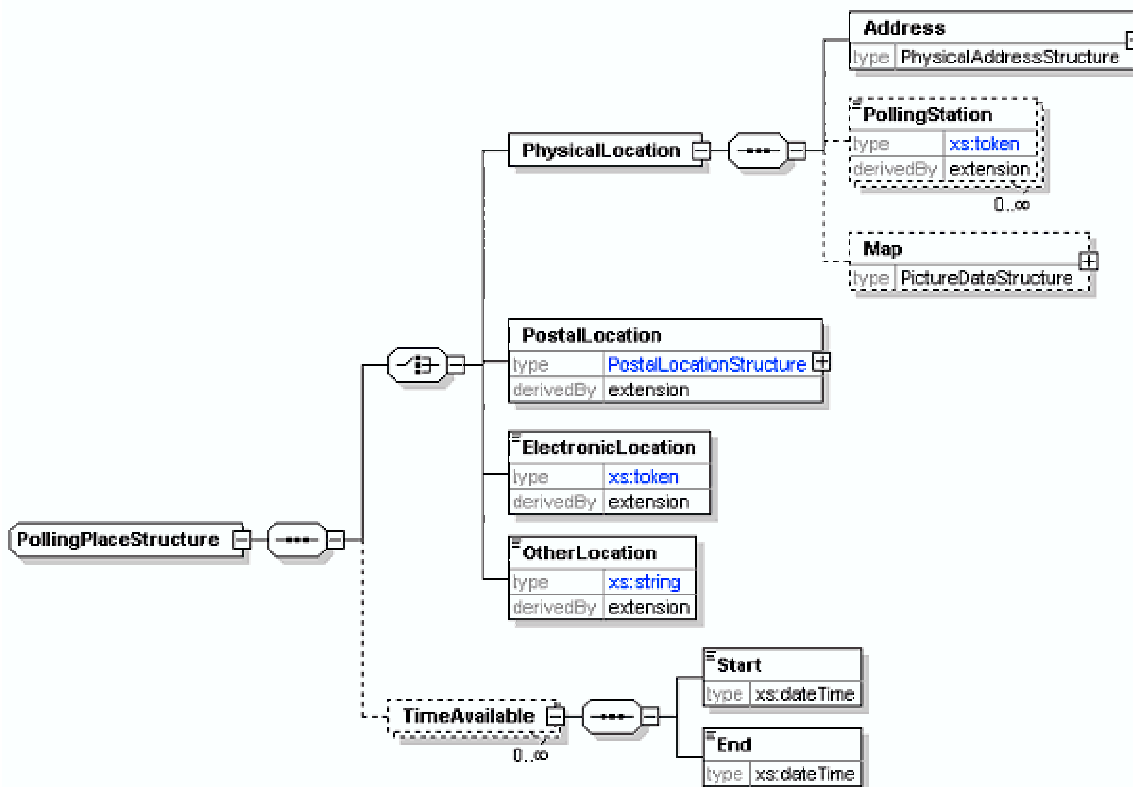
696 ## 5.2.31 PollingDistrictStructure

697

Element	Attribute	Type	Use	Comment
PollingDistrictStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	

698 The polling district indicates where a voter is registered to vote. The polling district can have a
699 name and an Id attribute. It can also be associated with other terms such as a constituency. This
700 is done through the Association element, which has Type attribute and may have an Id
701 attribute as well as a text value.

702 ## 5.2.32 PollingPlaceStructure

703

Element	Attribute	Type	Use	Comment
PollingPlaceStructure	Channel	VotingChannelType	required	
	DisplayOrder	xs:positiveInteger	optional	
PhysicalLocation	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
PostalLocation	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
ElectronicLocation	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
OtherLocation	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
PollingStation	Id	xs:NMTOKEN	optional	

704 In general, a polling place will be either a physical location (for paper or kiosk voting), a postal
705 address (for postal votes) or an electronic location (for Internet, SMS, telephone and other
706 electronic means of voting). However, it is possible that none of these types will meet every need,
707 and so an `OtherLocation` element has been included. Each of these locations must indicate the
708 channel for which it is to be used. If a single location supports multiple channels, it must be
709 included multiple times.

710 A physical location has an address. Sometimes, several polling stations will be at the same
711 address, so a polling station can be defined by name and/or Id within the address. Access to an
712 external map can also be provided as a URI or Base64 encoded binary data.

713 An electronic location must indicate its address (e.g. phone number, URL).

714 An optional `TimeAvailable` element is also provided. In most cases, this is not required as the
715 time a location is available is the same as the time the channel is available. However, there are
716 circumstances, such as the use of mobile polling stations, where this is not the case.

717 ## 5.2.33 PositionStructure

718 The `PositionStructure` is an extension of `xs:token` to add the following attributes:

Element	Attribute	Type	Use	Comment
PositionStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	

719 The element defined by this type indicates the position (e.g. President) for which an election is
720 being held. It has a text description and an optional ID.

721 ## 5.2.34 ProcessingUnitStructure
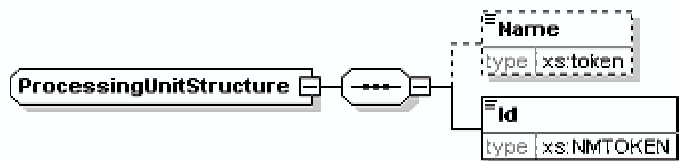
722

Element	Attribute	Type	Use	Comment
ProcessingUnitStructure	Role	xs:token (restricted)	required	

723 A processing unit is a physical system used in the election process. It is identified as part of audit
724 information by its ID (which might be an IP address) and optional name.

725 Each processing unit has an attribute to describe its rôle. The rôle can be "sender", "receiver",
726 "previous sender" or "next receiver". The latter two are used when there is a gateway involved.
727 For example, a 440 (cast vote) message might have an `OriginatingDevice` as its original
728 sender, a gateway as sender and voting system as receiver.
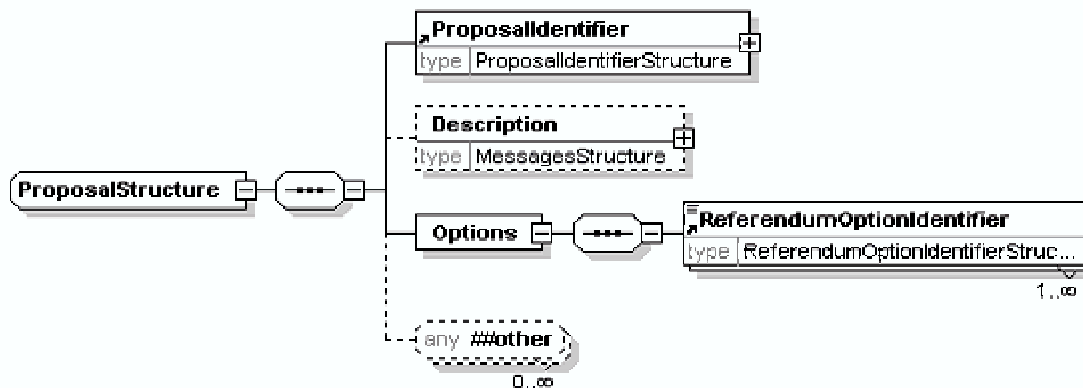
729 ## 5.2.35 ProposalIdentifierStructure
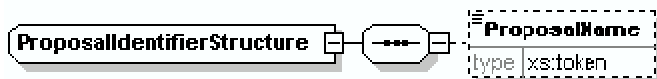
730

Element	Attribute	Type	Use	Comment
ProposalIdentifierStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
	ShortCode	ShortCodeType	optional	
	ExpectedConfirmationReference	ConfirmationReferenceType	optional	

731 A proposal is used in a referendum. At a basic level, it is a piece of text with the options ('yes' and
732 'no', 'for' and 'against' etc) to be voted on.

733 The proposal identifier indicates a system ID for the proposal. A short code can also be included,
734 either for SMS voting or where the security mechanism in place requires it. An
735 `ExpectedConfirmationReference` attribute also allows for security mechanisms where the
736 confirmation reference may be different for each combination of voter and candidate.

737 ## 5.2.36 ProposalStructure

738

Element	Attribute	Type	Use	Comment
ProposalStructure	Type	xs:token	optional	

739 The proposal identifier provides a name and ID. The description is used to provide the information
740 that will be displayed to the voter to indicate the aim of the proposal. The options are then used to
741 indicate how the voter may vote.

742 The Type attribute allows for referenda where there are different kinds of proposal, for example,
743 'initiative' or 'referendum'.

744 ## 5.2.37 ProposerStructure
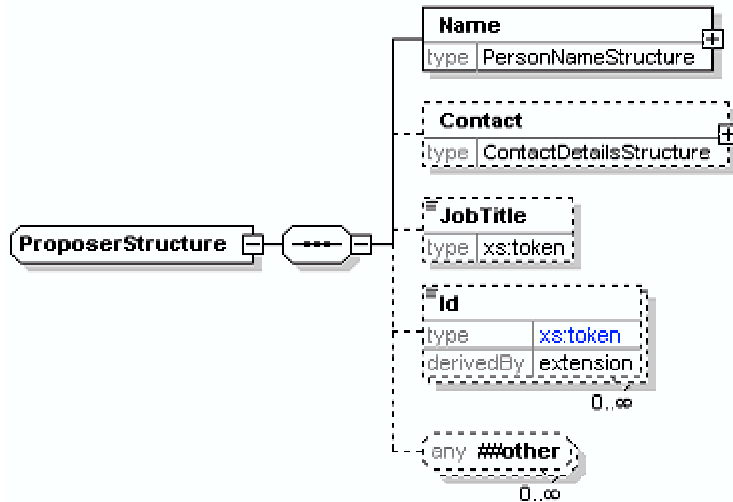
745

Element	Attribute	Type	Use	Comment
ProposerStructure	Category	xs:token (restricted)	optional	

746 A proposer proposes, seconds or endorses a candidate or referendum proposal. A proposer can
747 have a category, which indicates one of "primary", "secondary" or "other". A name is always
748 required, and additional information might be needed.

749 ## 5.2.38 ProxyStructure
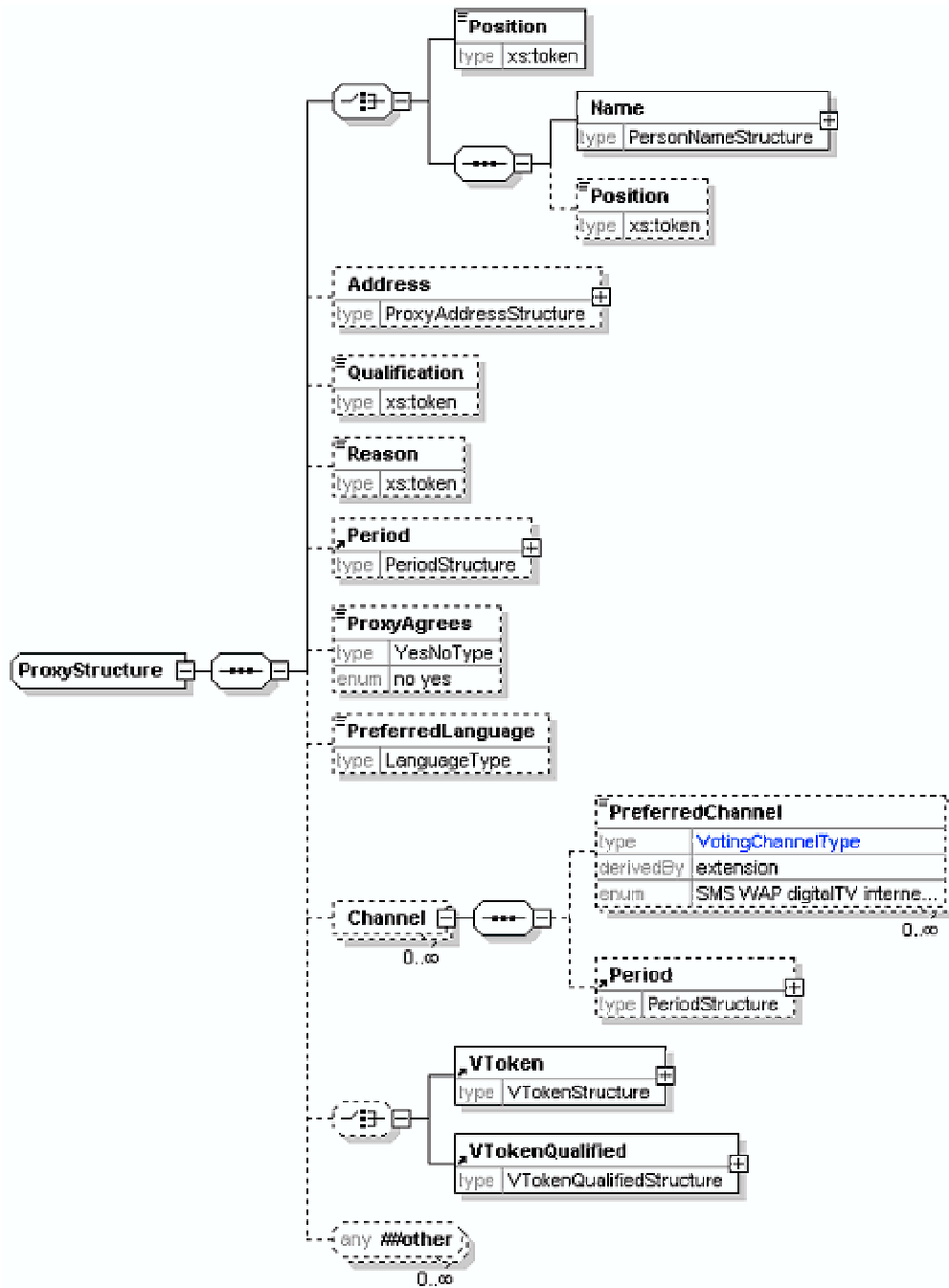
750

Element	Attribute	Type	Use	Comment
ProxyStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
PreferredChannel	Fixed	YesNoType	optional	

751 In many elections, a voter may appoint a proxy to vote on his or her behalf. That proxy may be
752 identified by position (for example, appointing the chairman as proxy at a company AGM), or by
753 name (for example, appointing your spouse as proxy for a public election), or both.

754 In some elections, the proxy must, for example, be a family member. This is indicated using the
755 `Qualification` element, while a reason for appointing a proxy can be indicated using the
756 `Reason` element.

757 A proxy can be permanent (i.e. appointed until revoked), appointed for one or more election
758 events (and individual elections within each event) or for a period of time. A proxy can also list his
759 or her preferred voting channels. These are listed in order of preference for a given period (which
760 may be specific election events, a date range or permanent), so that information can be sent
761 regarding the most appropriate voting channel at any election. The channel may be fixed, for
762 example, if registering to vote by a specific channel prevents voting by other means.

763 A proxy may also have a voting token, indicating the right to vote, or a qualified voting token,
764 indicating that there is a question over their right to vote.

765   5.2.39 ReferendumOptionIdentifierStructure

766 The `ReferendumOptionIdentifierStructure` is an extension of `xs:token` to add the
767 following attributes:

Element	Attribute	Type	Use	Comment
ReferendumOptionIdentifierStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
	ShortCode	ShortCodeType	optional	
	ExpectedConfirmationReference	ConfirmationReferenceType	optional	

768 A referendum option is used to indicate the possible answers to a referendum question, such as
769 "yes" and "no" or "for" and "against".

770 The referendum option identifier has a text description and can have a system ID. A short code
771 can also be included, either for SMS voting or where the security mechanism in place requires it.
772 An `ExpectedConfirmationReference` attribute also allows for security mechanisms where the
773 confirmation reference may be different for each combination of voter and option.

774   5.2.40 ReportingUnitIdentifierStructure

775 The `ReportingUnitIdentifierStructure` is an extension of `xs:token` to add the following
776 attributes:

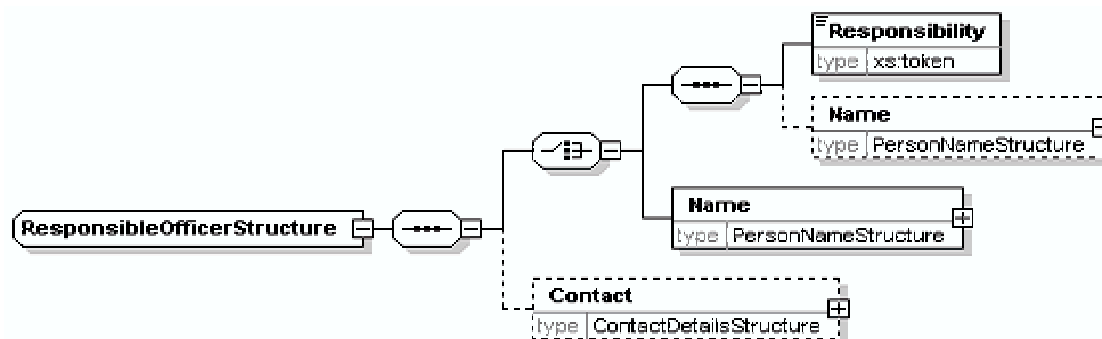| Element	Attribute	Type	Use	Comment
ReportingUnitIdentifierStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	

777 A reporting unit is an entity that reports partial information relating to a contest (votes or the
778 results of a count) without having the full set of information required to generate a result. This will
779 happen when votes from several independently managed areas must be amalgamated to
780 produce a result.

781 The reporting unit identifier structure defines a string with an optional `Id`.

5.2.41 ResponsibleOfficerStructure

783

Element	Attribute	Type	Use	Comment
ResponsibleOfficerStructure	Id	xs:NMTOKEN	optional	

784 A responsible officer is someone who has some sort of rôle to play in the organization of an
785 election. Each responsible officer has a name and/or responsibility (such as 'returning officer')
786 and optional contact information. Local rules will usually indicate the values allowed in the
787 `Responsibility` element.

5.2.42 ScrutinyRequirementStructure

789 The `ScrutinyRequirementStructure` is an extension of `xs:token` to add the following
790 attribute:

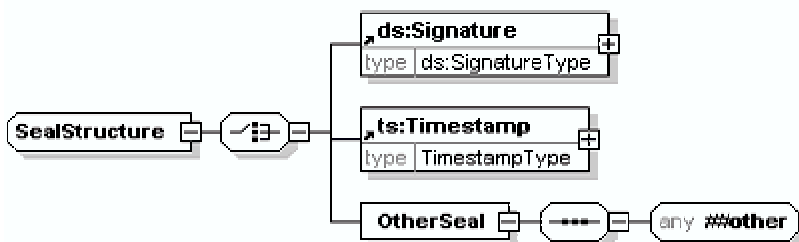| Element	Attribute	Type	Use	Comment
ScrutinyRequirementStructure	Type	xs:token	required	

791 A scrutiny requirement has two parts, a `Type` attribute and a text value. The `Type` specifies a
792 condition that a candidate must meet, such as an age or membership requirement or the payment
793 of a fee. The text describes how that condition has been met. For example:

794
795
796
797

```
<ScrutinyRequirement Type="dateofbirth">8 June
1955</ScrutinyRequirement>
```

5.2.43 SealStructure

799

Element	Attribute	Type	Use	Comment
OtherSeal	Type	xs:token	required	

800 The seal is used to protect information such as a vote, voting token or complete message. The
801 seal provides the means of proving that no alterations have been made to a message or
802 individual parts of a message such as a vote or collection of votes, from when they were originally

803 created by the voter. The seal may also be used to authenticate the identity of the system that
804 collected a vote, and provide proof of the time at which the vote was cast.

805 If a message is to be divided, each part must be separately sealed to protect the integrity of the
806 data. For example, if votes in several elections are entered on a single ballot, and these votes are
807 being counted in separate locations, each vote must be separately sealed.

808 A seal may be any structure which provides the required integrity characteristics, including an
809 XML signature [1] or a time-stamp.

810 The XML signature created by the voting system provides integrity and authentication of the
811 identity of the system that collected the vote. The time-stamp provides integrity of the vote and
812 proof of the time that the vote was cast.

813 ## 5.2.44 SimpleDateRangeStructure

814
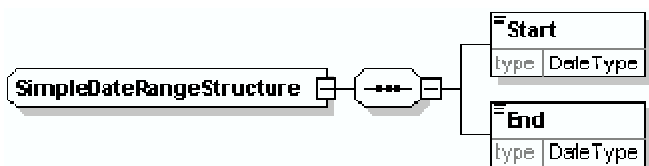
815 This data type is used to describe ranges of dates or dates and times.

816 ## 5.2.45 TelephoneStructure
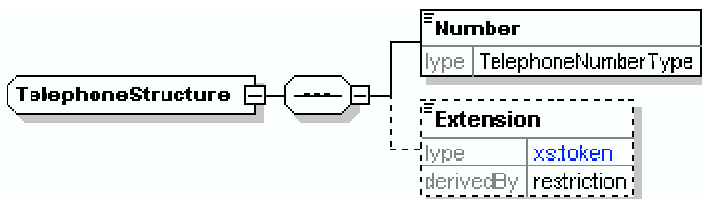
817

Element	Attribute	Type	Use	Comment
TelephoneStructure	Preferred	YesNoType	optional	
	Mobile	YesNoType	optional	

818 This is an extension of the `TelephoneType` and adds an `Extension` element and the two
819 attributes `Preferred` and `Mobile` of `YesNoType`. The `Preferred` attribute indicates which of
820 several phone numbers or fax numbers is preferred.
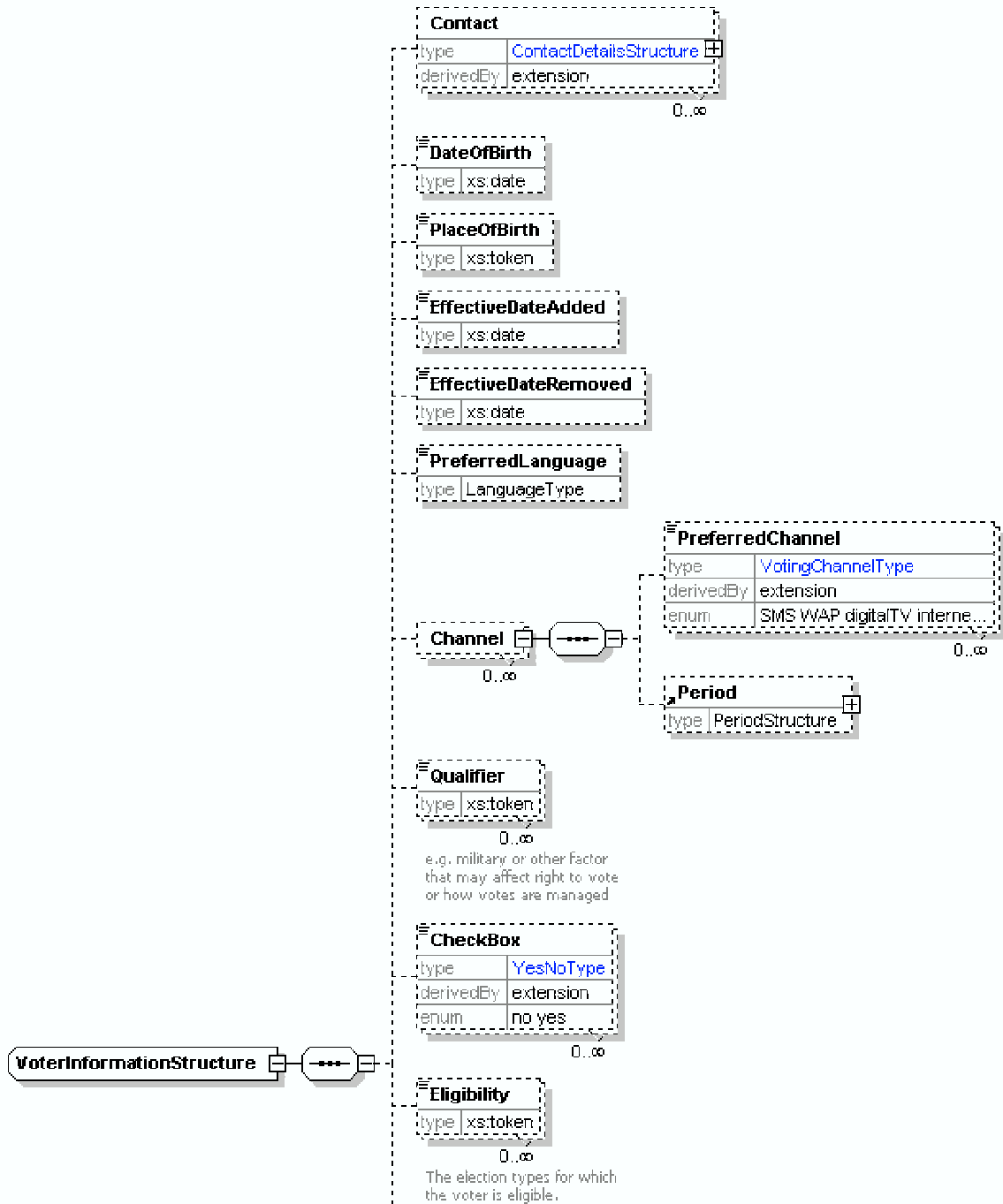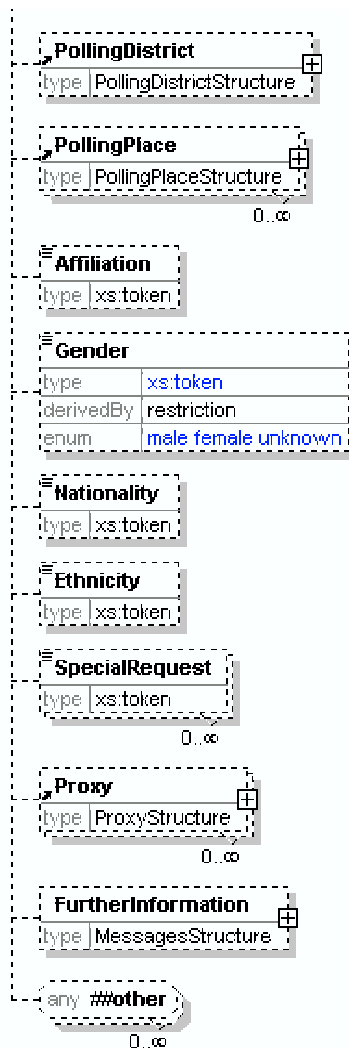
5.2.46 VoterIdentificationStructure
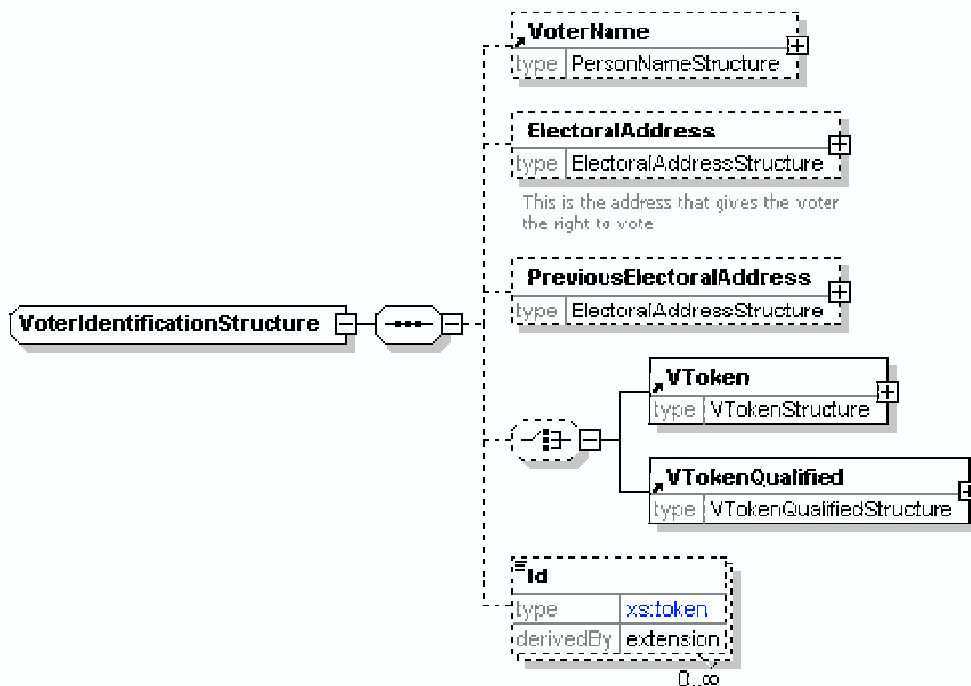
822

Element	Attribute	Type	Use	Comment
VoterIdentificationStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
Id	Type	xs:token	required	

823 An element defined by this data type is used wherever identification of a voter is required. It
824 contains the voter's name and electoral address (the address that gives them the right to vote in a
825 specific contest), the voting token (either normal or qualified) and a number of identifiers (such as
826 an electoral registration number). It may also include a previous electoral address if this is
827 required (for example, because a voter has not been at his or her current address for more than a
828 predefined period).

829

5.2.47 VoterInformationStructure

832

Element	Attribute	Type	Use	Comment
VoterInformationStructure	Id	xs:NMTOKEN	optional	
	DisplayOrder	xs:positiveInteger	optional	
ContactDetailsStructure	DisplayOrder	xs:positiveInteger	optional	standard attribute for this data type
	ElectionId	xs:NMTOKEN	optional	additional attribute
PreferredChannel	Fixed	YesNoType	optional	
Checkbox	Type	xs:token	required	

833 This contains more information about the voter. It contains all the information that would typically
834 be included on an electoral register other than that used for identification of the voter. In many
835 cases, it will be restricted to only include the information required in a specific message type.

836 A voter can list his or her preferred voting channels. These are listed in order of preference for a
837 given period (which may be specific election events, a date range or permanent), so that
838 information can be sent regarding the most appropriate voting channel at any election. The
839 channel may be fixed, for example, if registering to vote by a specific channel prevents voting by
840 other means.

841 The `Qualifier` element is used to hold information that might affect a voter's right to vote or how
842 the voting process is managed. Suitable enumerations for this are likely to be added as part of
843 localisation. The `CheckBox` element with its `Type` attribute allows binary information such as
844 whether the voter's entry on the electoral register can be sold, or whether the voter wants to
845 participate in the count. The eligibility indicates what election types a voter is eligible to participate
846 in.

847 Special requests are requests from the voter, for example, for wheelchair access to a polling
848 station.

5.2.48 VTokenStructure

850

Element	Attribute	Type	Use	Comment
Component	Type	xs:NMTOKEN	required	

851 The voting token contains the information required to authenticate the voter's right to vote in a
852 specific election or contest. A voting token can consist of a continuous string of encoded or
853 encrypted data, alternatively it may be constructed from several data components that a user may
854 input at various stages during the voting process (such as PIN, password and other coded data
855 elements). The totality of the voting token data proves that a person with the right to vote in the
856 specific election has cast the vote.

857 Depending on the type of election, the voter may need to cast their votes anonymously, thus not
858 providing a link to the voter's true identity. In this case the voting token data will not identify the
859 actual person casting the vote; it just proves that the vote was cast by a person with the right to
860 do so. Election rules may require a link to be maintained between a vote and a voter, in which
861 case a link is maintained between the voting token data and the voter's identity.

862 The components of the voting token are identified by a `Type` attribute and may contain text or
863 markup from any namespace depending on the token type. The content could be defined further
864 in separate schemas for specific types of token.

5.2.49 VTokenQualifiedStructure
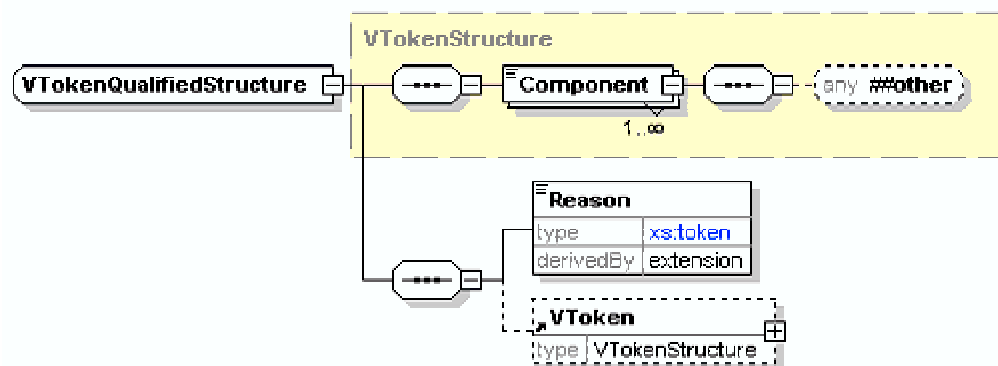
866

Element	Attribute	Type	Use	Comment
Reason	Type	xs:token	required	

867 There are occasions when a normal voting token cannot be used. For example, if a voter is
868 challenged, or an election officer claims the voter has already voted. In these circumstances a
869 qualified voting token can be used and treated appropriately by the election system according to
870 the election rules. For example, challenged votes might be ignored unless there were sufficient to

871   alter the result of the election, in which case each vote would be investigated and counted if
872   deemed correct to do so.

873   The `VTokenQualifiedStructure` is therefore an extension of the `VTokenStructure` to add
874   the additional information required. This additional information comprises a reason for
875   qualification (as a `Reason` element with a `Type` attribute and textual description) and possibly an
876   original `VToken`.