Voting Information Project Specification Documentation

Release 5.1.2

Feb 12, 2018

Contents

Welcome to the Voting Information Project's (VIP) *open XML* and *CSV* format specification. This data format provides an easy way to produce data that lets developers take a voter's address, compare it to street segments, and determine that voter's precinct (or precinct split). Knowing a voter's precinct allows information disseminators (such as Google) to provide voters with their official polling locations (and early voting sites), ballots (including both candidates and referenda), local election administrations, and election officials.

To see a changelog of all of the updates, please see the GitHub repository.

CHAPTER 1

XML Documentation

1.1 XML Specification

1.1.1 Getting Started

The actual election information specifies collections of elements, some containing links between each other. The entire set of tags must be encapsulated in a root object named vip_object. See the sample xml file and xsd file for more details.

Each top-level tag is a container for other fields, described in their own section. The only required top-level tags are the *source object* and the *election object*, each of which must be present exactly once. All other top-level tags can be repeated an unlimited number of times, or not included at all; order of top-level tags does not matter. Each top-level tag is required to have a single attribute, "id", which is required to be unique in a data file. The id attribute for the state object should be the state's FIPS number and this is strongly recommended. The id attributes are not required to remain constant for the same piece of semantic data across multiple productions of the feed (e.g. candidate Michael Smith, running for dogcatcher in Iowa, is not required to have the same candidate id attribute each time the state of Iowa publishes data).

In general, subtag data can appear a maximum of one time within each top-level tag object and in any order. Exceptions are noted below.

For the data itself, the special characters &, <, and > need to be encoded as &, <, and >, respectively.

Naming convention

While many of the Voting Information Project's data processes are managed by software, the quality of the entire system relies on human intervention, especially for error reporting and quality control. For this reason, VIP files should follow a naming convention that describes the contents of each individual feed file in an accessible way.

The file containing the VIP feed should be named using the following convention:

```
vipfeed-${FIPS}-${ELECTION_DATE}-${STATE}[-${LOCAL}].{xml|zip}
```

An explanation of each of the segments of the file naming convention above are as follows:

- `${FIPS}` - The FIPS code for the jurisdiction.
- `${ELECTION_DATE}` - The date of the election in ISO 8601 format.
- `${STATE}` - The full state name (e.g. Alaska, Arkansas, etc...) and not the abbreviation. If there are spaces in the state name, they should be substituted with underscores (e.g. New York -> New_York).
- `${LOCAL}` (optional) - This additional identifier should be used if the file contains data from a specific jurisdiction. As with `${STATE}` above, all spaces should be substituted with underscores. For example, if the data contained in the file only covers Maricopa County, AZ for the November 6, 2012 election, the file name would be `vipfeed-04013-2012-11-06-Arizona-Maricopa_County.xml`.
- `{xml|zip}` - If the file is an uncompressed XML document, the extension should be `.xml`. If the file is zipped, the file extension should end with `.zip`.

For a final example, `vipfeed-19-2012-11-06-Iowa.zip` denotes Iowa's (**NB:** the FIPS code for IA is 19) feed for the Nov 6, 2012 election that has been compressed.

1.1.2 Best Practices

There are many different ways to generate a valid feed. We strongly encourage reviewing and adhering to the guidelines described in the following best practices sections:

XML Data Best Practices

Following is a series of best practice for data collection and XML file creation and suggestions about best practices of formatting data within your VIP XML file.

Element Identifiers

Most elements within the VIP feed require unique identifiers, `xs:ID` data types, and in order to maintain uniqueness and provide context for the identifiers, the best practice is to use Hungarian-Style notation for identifiers.

ID values should follow Hungarian-Style notation, were the identifier prefix implicitly names the data element. Below is a list of preferred prefixes by element (e.g. par00001 for a `Party id`):

Element	Prefix
BallotMeasureContest	bmc
BallotMeasureSelection	bms
BallotStyle	bs
Candidate	can
CandidateContest	cc
CandidateSelection	cs
ContactInformation	ci
Election	ele
ElectionAdministration	ea
ElectoralDistrict	ed
HoursOpen	hours
Locality	loc
Office	off
OrderedContest	oc
Party	par
PartyContest	pc
PartySelection	ps
Person	per
PollingLocation	pl
RetentionContest	rc
Source	src
State	st
StreetSegment	ss

File Structure

All XML files should be encoded UFT-8 and line breaks should be LF (`\n`) as opposed to CR LF (`\r\n`).

For consistency across files and to aid human readability all indentation of elements should be an indent of two spaces and tabs should not be used. Each child node of an element should also be indented an additional two spaces.

General Data Structure

All data that are presented to end users of the data (i.e. contest names, referendum text, polling location names, street names, proper names), where possible, should be converted to Title Case to aid readability.

All data should be trimmed to remove leading and trailing white space.

Optional elements without values should be omitted from XML feed.

Specific Data Types

Elements with a data type of `xs:integer` must contain a valid whole number greater than zero.

Elements with a data type of `xs:anyURI` should be entered as a fully qualified domain name (e.g. https://www.votinginfoproject.org/)

Elements with a data type of `xs:dateTime` should be entered in ISO-8601 format.

Elements with a data type of `xs:boolean` should either have a value of `true` or `false`

Elements with a data type of `xs:language` should contain a two character, lower-case, value corresponding to the ISO 639 standard.

Elements that have enumerations which include an `other` should have a corresponding value assigned to `OtherType` within the containing element. For example:

```
<BallotMeasureContest id="bm390616670907">
   <BallotSelectionId>bms390616670907</BallotSelectionId>
   <ElectoralDistrictId>ed3906177703103</ElectoralDistrictId>
   <Name>Proposed Tax Levy School District</Name>
   <SequenceOrder>34</SequenceOrder>
   <FullText>
     <Text language="en">An additional tax for the benefit of the Lockland Local␣
↪School District, County of Hamilton,
     Ohio, for the purpose of CURRENT EXPENSES at a rate not exceeding eleven and two-
↪tenths (11.2) mills for each
     one dollar of valuation, which amounts to one dollar and twelve cents ($1.12)␣
↪for each one hundred dollars of
     valuation, for a continuing period of time, commencing in 2015, first due in␣
↪calendar year 2016.</Text>
   </FullText>
   <SummaryText>
     <Text language="en">4 Proposed Tax Levy</Text>
   </SummaryText>
   <Type>other</Type>
   <OtherType>bond</OtherType>
</BallotMeasureContest>
```

Specific Data Elements

Street Segments: Valid street segment records should not contain leading zeros in `xs:integer` fields and should have a `Zip` value of `00000` if a value is unknown.

External Identifiers: External identifiers with an enumeration of `fips` should contain valid FIPS code values as defined by the U.S. Census Bureau. External identifiers with an enumeration of `ocd-id` should contain a valid Open Civic Data Division Identifier.

For long text fields (e.g. `FullText` in `BallotMeasureContest`) the XML line break (`
`) should be used to enforce line break styling.

In all fields the characters <, >, and `&` should be encoded `<`, `>`, and `&` respectively.

These sections outline several recommendations for feed layout, element identifiers, and enumeration selection that will make it easier to generate and troubleshoot your VIP feeds.

1.1.3 Elements & Enumerations

Single-page Format

XML Elements & Enumerations (Single Page)

Elements

State

The State object includes state-wide election information. The ID attribute is recommended to be the state's FIPS code, along with the prefix "st".

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ElectionAdministrationId	`xs:IDREF`	Optional	Single	Links to the state's election administration object.	If the field is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifier for the state that relates to another dataset (e.g. OCD-ID).	If the element is invalid or not present, then the implementation is required to ignore it.
Name	`xs:string`	**Required**	Single	Specifiers the name of a state, such as Alabama.	If the field is invalid, then the implementation is required to ignore it.
PollingLocationIds	`xs:IDREFS`	Optional	Single	Specifies a link to the state's *polling locations*. If early vote centers or ballot drop locations are state-wide (e.g., anyone in the state can use them), they can be specified here, but you are encouraged to only use the *Precinct* element.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<State id="st51">
   <ElectionAdministrationId>ea40133</ElectionAdministrationId>
   <ExternalIdentifiers>
     <ExternalIdentifier>
       <Type>ocd-id</Type>
       <Value>ocd-division/country:us/state:va</Value>
     </ExternalIdentifier>
   </ExternalIdentifiers>
   <Name>Virginia</Name>
</State>
```

Office

`Office` represents the office associated with a contest or district (e.g. Alderman, Mayor, School Board, et al).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ContactInformation	*ContactInformation*	Optional	Repeats	Specifies the contact information for the office and/or individual holding the office.	If the element is invalid or not present, then the implementation is required to ignore it.
Description	*InternationalizedText*	Optional	Single	A brief description of the office and its purpose.	If the element is invalid or not present, then the implementation is required to ignore it.
ElectoralDistrictId	`xs:IDREF`	**Required**	Single	Links to the *ElectoralDistrict* element associated with the office.	If the field is invalid or not present, the implementation is required to ignore the `Office` element containing it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifiers that link this office to other related datasets (e.g. campaign finance systems, OCD IDs, et al.).	If the element is invalid or not present, then the implementation is required to ignore it.
FilingDeadline	`xs:date`	Optional	Single	Specifies the date and time when a candidate must have filed for the contest for the office.	If the field is invalid or not present, then the implementation is required to ignore it.
IsPartisan	`xs:boolean`	Optional	Single	Indicates whether the office is partisan.	If the field is invalid or not present, then the implementation is required to ignore it.
Name	*InternationalizedText*	**Required**	Single	The name of the office.	If the field is invalid or not present, the implementation is required to ignore the `Office` element containing it.
OfficeHolderPersonIds	`xs:IDREFS`	Optional	Single	Links to the *Person* element(s) that hold additional information about the current office holder(s).	If the field is invalid or not present, then the implementation is required to ignore it.
Term	*Term*	Optional	Single	Defines the term the office can be held.	If the element is invalid or not present, then the implementation is required to ignore it.

Term

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Type	*OfficeTermType*	Optional	Single	Specifies the type of office term (see *OfficeTermType* for valid values).	If the field is invalid or not present, the implementation is required to ignore the `Office` element containing it.
StartDate	`xs:date`	Optional	Single	Specifies the start date for the current term of the office.	If the field is invalid or not present, then the implementation is required to ignore it.
EndDate	`xs:date`	Optional	Single	Specifies the end date for the current term of the office.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<Office id="off0000">
  <ElectoralDistrictId>ed60129</ElectoralDistrictId>
  <FilingDeadline>2013-01-01</FilingDeadline>
  <IsPartisan>false</IsPartisan>
  <Name>
    <Text language="en">Governor</Text>
  </Name>
  <Term>
    <Type>full-term</Type>
  </Term>
</Office>
```

Election

The Election object represents an Election Day, which usually consists of many individual contests and/or referenda. A feed must contain **exactly one** Election object. All relationships in the feed (e.g., street segment to precinct to polling location) are assumed to relate only to the Election specified by this object. It is permissible, and recommended, to combine unrelated contests (e.g., a special election and a general election) that occur on the same day into one feed with one Election object.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Date	`xs:date`	**Required**	Single	Specifies when the election is being held. The *Date* is considered to be in the timezone local to the state holding the election.	If the field is invalid, then the implementation is required to ignore the `Election` element containing it.
ElectionType	*InternationalizedText*	Optional	Single	Specifies the highest controlling authority for election (e.g., federal, state, county, city, town, etc.)	If the element is invalid or not present, then the implementation is required to ignore it.
StateId	`xs:IDREF`	**Required**	Single	Specifies a link to the *State* element where the election is being held.	If the field is invalid, then the implementation is required to ignore the `Election` element containing it.
IsStatewide	`xs:boolean`	Optional	Single	Indicates whether the election is statewide.	If the field is not present or invalid, the implementation is required to default to "yes".
Name	*InternationalizedText*	Optional	Single	The name for the election (**NB:** while optional, this element is highly recommended).	If the element is invalid or not present, then the implementation is required to ignore it.
RegistrationInfo	*InternationalizedText*	Optional	Single	Specifies information about registration for this election either as text or a URI.	If the element is invalid or not present, then the implementation is required to ignore it.
AbsenteeBallotInfo	*InternationalizedText*	Optional	Single	Specifies information about requesting absentee ballots either as text or a URI	If the element is invalid or not present, then the implementation is required to ignore it.
ResultsUri	`xs:anyURI`	Optional	Single	Contains a URI where results for the election may be found	If the field is invalid or not present, then the implementation is required to ignore it.
PollingHours	*Ho*	Op-	Sin-	Contains the hours (in local time) that Election Day polling loca-	If the element

```
<Election id="ele30000">
  <AbsenteeRequestDeadline>2013-10-30</AbsenteeRequestDeadline>
  <Date>2013-11-05</Date>
  <ElectionType>
    <Text language="en">General</Text>
    <Text language="es">Generales</Text>
  </ElectionType>
  <HasElectionDayRegistration>false</HasElectionDayRegistration>
  <HoursOpenId>hours0001</HoursOpenId>
  <IsStatewide>true</IsStatewide>
  <Name>
    <Text language="en">2013 Statewide General</Text>
  </Name>
  <RegistrationDeadline>2013-10-15</RegistrationDeadline>
  <ResultsUri>http://www.sbe.virginia.gov/ElectionResults.html</ResultsUri>
  <StateId>st51</StateId>
</Election>
```

BallotSelectionBase

A base model for all ballot selection types: *BallotMeasureSelection*, *CandidateSelection*, and *PartySelection*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
SequenceOrder	xs:integer	Optional	Single	The order in which a selection can be listed on the ballot or in results. This is the default ordering, and can be overridden by *OrderedBallotSlectionIds* in *OrderedContest*.	If the field is invalid or not present, then the implementation is required to ignore it.

Source

The Source object represents the organization that is publishing the information. This object is the only required object in the feed file, and only one source object is allowed to be present.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Name	xs:string	**Required**	Single	Specifies the name of the organization that is providing the information.	If the field is invalid, then the implementation is required to ignore the `Source` element containing it.
VipId	xs:string	**Required**	Single	Specifies the ID of the organization. VIP uses FIPS codes for this ID.	If the field is invalid, then the implementation is required to ignore the `Source` element containing it.
DateTime	xs:dateTime	**Required**	Single	Specifies the date and time of the feed production. The date/time is considered to be in the timezone local to the organization.	If the field is invalid, then the implementation is required to ignore it.
Description	*InternationalizedText*	Optional	Single	Specifies both the nature of the organization providing the data and what data is in the feed.	If the element is invalid or not present, then the implementation is required to ignore it.
OrganizationUri	xs:string	Optional	Single	Specifies a URI to the home page of the organization publishing the data.	If the field is invalid or not present, then the implementation is required to ignore it.
FeedContactId	xs:IDREF	Optional	Single	Reference to the *Person* who will respond to inquiries about the information contained within the file.	If the field is invalid or not present, then the implementation is required to ignore it.
TouUri	xs:anyURI	Optional	Single	Specifies the website where the Terms of Use for the information in this file can be found.	If the field is invalid or not present, then the implementation is required to ignore it.
Version	xs:string	**Required**	Single	Specifies the version of the data	If the field is invalid, then the implementation is required to ignore it.

```
<Source id="src1">
   <DateTime>2013-10-24T14:25:28</DateTime>
   <Description>
      <Text language="en">SBE is the official source for Virginia data</Text>
   </Description>
   <Name>State Board of Elections, Commonwealth of Virginia</Name>
   <OrganizationUri>http://www.sbe.virginia.gov/</OrganizationUri>
   <VipId>51</VipId>
   <Version>5.0</Version>
</Source>
```

BallotMeasureContest

The BallotMeasureContest provides information about a ballot measure before the voters, including summary statements on each side. Extends *ContestBase*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ConStatement	*InternationalizedText*	Optional	Single	Specifies a statement in opposition to the referendum. It does not necessarily appear on the ballot.	If the element is invalid or not present, then the implementation is required to ignore it.
EffectOfAbstain	*InternationalizedText*	Optional	Single	Specifies what effect abstaining (i.e. not voting) on this proposition will have (i.e. whether abstaining is considered a vote against it).	If the element is invalid or not present, then the implementation is required to ignore it.
FullText	*InternationalizedText*	Optional	Single	Specifies the full text of the referendum as it appears on the ballot.	If the element is invalid or not present, then the implementation is required to ignore it.
InfoUri	`xs:anyURI`	Optional	Single	Specifies a URI that links to additional information about the referendum.	If the field is invalid or not present, then the implementation is required to ignore it.
PassageThreshold	*InternationalizedText*	Optional	Single	Specifies the threshold of votes that the referendum needs in order to pass. The default is a simple majority (i.e. 50% plus one vote). Other common thresholds are "three-fifths" and "two-thirds". If there are competing initiatives, information about their effect on the passage of the BallotMeasureContest would go here.	If the element is invalid or not present, then the implementation is required to ignore it.
ProStatement	*InternationalizedText*	Optional	Single	Specifies a statement in favor of the referendum. It does not necessarily appear on the ballot.	If the element is invalid or not present, then the implementation is required to ignore it.
SummaryText	*InternationalizedText*	Optional	Single	Specifies a short summary of the referendum that is on the ballot, below the title, but above the text.	If the element is invalid or not present, then the implementation is required to ignore it.
Type	*BallotMeasureType*	Optional	Single	Specifies the particular type of ballot measure. Must be one of the valid *BallotMeasureType* options.	If the field is invalid or not present, then the implementation is required to ignore it.
OtherType	`xs:string`	Optional	Single	Allows for cataloging a new *BallotMeasureType* option, when Type is specified as "other."	If the field is invalid or not present, then the

```
<BallotMeasureContest id="bmc30001">
   <BallotSelectionIds>bms30001a bms30001b</BallotSelectionIds>
   <BallotTitle>
      <Text language="en">State of the State</Text>
      <Text language="es">Estado del Estado.</Text>
   </BallotTitle>
   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
   <Name>Referendum on Virginia</Name>
   <ConStatement label="bmc30001con">
      <Text language="en">This is no good.</Text>
      <Text language="es">Esto no es bueno.</Text>
   </ConStatement>
   <EffectOfAbstain label="bmc30001abs">
      <Text language="en">Nothing will happen.</Text>
      <Text language="es">Nada pasará.</Text>
   </EffectOfAbstain>
   <ProStatement label="bmc30001pro">
      <Text language="en">Everything will be great.</Text>
      <Text language="es">Todo va a estar bien.</Text>
   </ProStatement>
   <Type>referendum</Type>
</BallotMeasureContest>
```

ContestBase

A base model for all Contest types: *BallotMeasureContest*, *CandidateContest*, *PartyContest*, and *RetentionContest* (NB: the latter because it extends *BallotMeasureContest*).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Abbreviation	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
BallotSelectionIds	xs:IDREFS	Optional	Single	References a set of BallotSelections, which could be of any selection type that extends *BallotSelectionBase*.	If the field is invalid or not present, then the implementation should ignore it.
BallotSubTitle	*InternationalizedText*	Optional	Single	Subtitle of the contest as it appears on the ballot.	If the element is invalid or not present, then the implementation should ignore it.
BallotTitle	*InternationalizedText*	Optional	Single	Title of the contest as it appears on the ballot.	If the element is invalid or not present, then the implementation should ignore it.
ElectoralDistrictId	xs:IDREF	**Required**	Single	References an *ElectoralDistrict* element that represents the geographical scope of the contest.	If the field is invalid, then the implementation should ignore it.
ElectorateSpecification	*InternationalizedText*	Optional	Single	Specifies any changes to the eligible electorate for this contest past the usual, "all registered voters" electorate. This subtag will most often be used for primaries and local elections. In primaries, voters may have to be registered as a specific party to vote, or there may be special rules for which ballot a voter can pull. In some local elections, non-citizens can vote.	If the element is invalid or not present, then the implementation should ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifiers for a contest that links to another source of information.	If the element is invalid or not present, then the implementation should ignore it.
HasRotation	xs:boolean	Optional	Single	Indicates whether the selections in the contest are rotated.	If the field is invalid or not present, then the implementation should ignore it.
Name	xs:string	Optional	Single	Name of the contest, not necessarily how it appears on the ballot (NB: BallotTitle should be used for this purpose).	If the field is invalid or not present, then the implementation should ignore it.

BallotStyle

A container for the contests/measures on the ballot.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ImageUri	`xs:anyURI`	Optional	Single	Specifies a URI that returns an image of the sample ballot.	If the field is invalid or not present, then the implementation is required to ignore it.
OrderedContestIds	`xs:IDREFS`	Optional	Single	Reference to a set of :ref:‘single-xml-ordered-contest‘s	If the field is invalid or not present, then the implementation is required to ignore it.
PartyIds	`xs:IDREFS`	Optional	Single	Reference to a set of :ref:‘single-xml-party‘s.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<BallotStyle id="bs00000">
   <OrderedContestIds>oc20003 oc20004 oc20005 oc20025 oc20355 oc20449</
↪OrderedContestIds>
</BallotStyle>
```

Person

`Person` defines information about a person. The person may be a candidate, election administrator, or elected official. These elements reference `Person`:

- *Candidate*
- *ElectionAdministration*
- *Office*

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ContactInformation	*ContactInformation*	Optional	Repeats	Specifies contact information for the person.	If the element is invalid or not present, then the implementation is required to ignore it.
DateOfBirth	`xs:date`	Optional	Single	Represents the individual's date of birth.	If the field is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Identifiers for this person.	If the element is invalid or not present, then the implementation is required to ignore it.
FirstName	`xs:string`	Optional	Single	Represents an individual's first name.	If the field is invalid or not present, then the implementation is required to ignore it.
FullName	*InternationalizedText*	Optional	Single	Specifies a person's full name (**NB:** this information is *InternationalizedText* because it sometimes appears on ballots in multiple languages).	If the element is invalid or not present, then the implementation is required to ignore it.
Gender	`xs:string`	Optional	Single	Specifies a person's gender.	If the field is invalid or not present, then the implementation is required to ignore it.
LastName	`xs:string`	Optional	Single	Represents an individual's last name.	If the field is invalid or not present, then the implementation is required to ignore it.
MiddleName	`xs:string`	Optional	Repeats	Represents any number of names between an individual's first and last names (e.g. John **Ronald Reuel** Tolkien).	If the field is invalid or not present, then the implementation is required to ignore it.
Nickname	`xs:string`	Optional	Single	Represents an individual's nickname.	If the field is invalid or not present, then the implementation is required to ignore it.
PartyId	`xs:IDREF`	Optional	Single	Refers to the associated *Party*. This information is intended to be used by feed consumers to help them disambiguate the person's identity, but not to be presented as part of any ballot information. For that see *Candidate* **PartyId**.	If the field is invalid or not present, then the implementation is required to ignore it.
Prefix	`xs:string`	Optional	Single	Specifies a prefix associated with a person (e.g. Dr.).	If the field is invalid or not present, then the implementation is required to ignore it.
Profession	*InternationalizedText*	Optional	Single	Specifies a person's profession (**NB:** this information is *InternationalizedText* because it sometimes appears on ballots in multiple languages).	If the element is invalid or not present, then the implementation is required to ignore it.

```
<Person id="per50001">
   <ContactInformation label="ci60002">
     <Email>rwashburne@albemarle.org</Email>
     <Phone>4349724173</Phone>
   </ContactInformation>
   <FirstName>RICHARD</FirstName>
   <LastName>WASHBURNE</LastName>
   <MiddleName>J.</MiddleName>
   <Nickname>JAKE</Nickname>
   <Title>
     <Text language="en">General Registrar Physical</Text>
   </Title>
</Person>
```

ExternalIdentifiers

The `ExternalIdentifiers` element allows VIP data to connect with external datasets (e.g. candidates with campaign finance datasets, electoral geographies with OCD-IDs that allow for greater connectivity with additional datasets, etc...). Examples for `ExternalIdentifiers` can be found on the objects that support them:

- *Candidate*
- Any element that extends *ContestBase*
- *ElectoralDistrict*
- *Locality*
- *Office*
- *Party*
- *Precinct*
- *State*

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ExternalIdentifier	*ExternalIdentifier*	**Required**	Repeats	Defines the identifier and the type of identifier it is (see *ExternalIdentifier* for complete information).	At least one valid *ExternalIdentifier* must be present for `ExternalIdentifiers` to be valid. If no valid *ExternalIdentifier* is present, the implementation is required to ignore the `ExternalIdentifiers` element.

ExternalIdentifier

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Type	*IdentifierType*	**Required**	Single	Specifies the type of identifier. Must be one of the valid types as defined by *IdentifierType*.	If the field is invalid or not present, the implementation is required to ignore the `ElectionIdentifier` containing it.
OtherType	`xs:string`	Optional	Single	Allows for cataloging an `ExternalIdentifier` type that falls outside the options listed in *IdentifierType*. `Type` should be set to "other" when using this field.	If the field is invalid or not present, then the implementation is required to ignore it.
Value	`xs:string`	**Required**	Single	Specifies the identifier.	If the field is invalid or not present, the implementation is required to ignore the `ElectionIdentifier` containing it.

```
<ExternalIdentifiers>
   <ExternalIdentifier>
      <Type>ocd-id</Type>
      <Value>ocd-division/country:us/state:nc/county:durham</Value>
   </ExternalIdentifier>
   <ExternalIdentifier>
      <Type>FIPS</Type>
      <Value>37063</Value>
   </ExternalIdentifier>
   <ExternalIdentifier>
      <Type>OTHER</Type>
      <OtherType>GNIS</OtherType>
      <Value>1008550</Value>
   </ExternalIdentifier>
   <external_identifer>
      <Type>OTHER</Type>
      <OtherType>census</OtherType>
      <Value>99063</Value>
   </ExternalIdentifier>
</ExternalIdentifiers>
```

ContactInformation

For defining contact information about objects such as persons, boards of authorities, organizations, etc. ContactInformation is always a sub-element of another object (e.g. *ElectionAdministration*, *Office*, *Person*, *Source*). ContactInformation has an optional attribute `label`, which allows the feed to refer back to the original label for the information (e.g. if the contact information came from a CSV, `label` may refer to a row ID).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
AddressLine	`xs:string`	Optional	Repeats	The "location" portion of a mailing address. *See usage note.*	If the field is invalid or not present, then the implementation is required to ignore it.
Directions	*InternationalizedText*	Optional	Single	Specifies further instructions for locating this entity.	If the element is invalid or not present, then the implementation is required to ignore it.
Email	`xs:string`	Optional	Repeats	An email address for the contact.	If the field is invalid or not present, then the implementation is required to ignore it.
Fax	`xs:string`	Optional	Repeats	A fax line for the contact.	If the field is invalid or not present, then the implementation is required to ignore it.
Hours **[deprecated]**	*InternationalizedText*	Optional	Single	Contains the hours (in local time) that the location is open *(NB: this element is deprecated in favor of the more structured :ref:'single-xml-hours-open' element. It is strongly encouraged that data providers move toward contributing hours in this format).*	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	`xs:IDREF`	Optional	Single	References an *HoursOpen* element, which lists the hours of operation for a location.	If the field is invalid or not present, then the implementation is required to ignore it.
LatLng	*LatLng*	Optional	Single	Specifies the latitude and longitude of this entity.	If the element is invalid or not present, then the implementation is required to ignore it.
Name	`xs:string`	Optional	Single	The name of the location or contact. *See usage note.*	If the field is invalid or not present, then the implementation is required to ignore it.
Phone	`xs:string`	Optional	Repeats	A phone number for the contact.	If the field is invalid or not present, then the implementation is required to ignore it.
Uri	`xs:anyURI`	Optional	Repeats	An informational URI for the contact or location.	If the field is invalid or not present, then the implementation is required to ignore it.

`Name` and `AddressLine` Usage Note

The `Name` and `AddressLine` fields should be chosen so that a display or mailing address can be constructed programmatically by joining the `Name` and `AddressLine` fields together. For example, for the following address:

```
Department of Elections
1 Dr. Carlton B. Goodlett Place, Room 48
San Francisco, CA 94102
```

The name could be "Department of Elections" and the first address line could be "1 Dr. Carlton B. Goodlett Place, Room 48."

However, VIP does not yet support the representation of mailing addresses whose "name" portion spans more than one line, for example:

```
California Secretary of State
Elections Division
1500 11th Street
Sacramento, CA 95814
```

For addresses like the above, we recommend choosing a name like, "California Secretary of State, Elections Division" with "1500 11th Street" as the first address line. This would result in a programmatically constructed address like the following:

```
California Secretary of State, Elections Division
1500 11th Street
Sacramento, CA 95814
```

```
<ContactInformation label="ci10861a">
   <AddressLine>1600 Pennsylvania Ave</AddressLine>
   <AddressLine>Washington, DC 20006</AddressLine>
   <Email>president@whitehouse.gov</Email>
   <Phone>202-456-1111</Phone>
   <Phone annotation="TDD">202-456-6213</Phone>
   <Uri>http://www.whitehouse.gov</Uri>
</ContactInformation>
```

CandidateSelection

CandidateSelection extends *BallotSelectionBase* and represents a ballot selection for a candidate contest.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
CandidateIds	xs:IDREFS	Optional	Single	References a set of *Candidate* elements. The number of candidates that can be references is unbounded in cases where the ballot selection is for a ticket (e.g. "President/Vice President", "Governor/Lt Governor").	If the field is invalid or not present, then the implementation is required to ignore it.
EndorsementPartyIds	xs:IDREFS	Optional	Single	References a set of *Party* elements, which signifies one or more endorsing parties for the candidate(s).	If the field is invalid or not present, then the implementation is required to ignore it.
IsWriteIn	xs:boolean	Optional	Single	Signifies if the particular ballot selection allows for write-in candidates. If true, one or more write-in candidates are allowed for this contest.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<CandidateSelection id="cs10861">
   <CandidateIds>can10861a can10861b</CandidateIds>
   <EndorsementPartyIds>par0001</EndorsementPartyIds>
</CandidateSelection>
```

LatLng

The latitude and longitude of a polling location in WGS 84 format. Both latitude and longitude values are measured in decimal degrees.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Latitude	xs:float	**Required**	Single	The latitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
Longitude	xs:float	**Required**	Single	The longitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
Source	xs:string	Optional	Single	The system used to perform the lookup from location name to lat/lng. For example, this could be the name of a geocoding service.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<PollingLocation id="pl81274">
   <AddressLine>ALBEMARLE HIGH SCHOOL</AddressLine>
   <AddressLine>2775 Hydraulic Rd</AddressLine>
   <AddressLine>Charlottesville, VA 229018917</AddressLine>
   <HoursOpenId>hours0001</HoursOpenId>
   <LatLng>
     <Latitude>38.0754627</Latitude>
     <Longitude>-78.5014875</Longitude>
     <Source>Google Maps</Source>
   </LatLng>
</PollingLocation>
```

Candidate

The Candidate object represents a candidate in a contest. If a candidate is running in multiple contests, each contest **must** have its own Candidate object. Candidate objects may **not** be reused between Contests.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
BallotName	*InternationalizedText*	**Required**	Single	The candidate's name as it will be displayed on the official ballot (e.g. "Ken T. Cuccinelli II").	If the element is invalid or not present, then the implementation is required to ignore the Candidate element containing it.
ContactInformation	*ContactInformation*	Optional	Single	Contact and physical address information for this Candidate and/or their campaign (see *ContactInformation*).	If the element is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Another identifier for a candidate that links to another source of information (e.g. a campaign committee ID that links to a campaign finance system).	If the element is invalid or not present, then the implementation is required to ignore it.
FileDate	`xs:date`	Optional	Single	Date when the candidate filed for the contest.	If the field is invalid or not present, then the implementation is required to ignore it.
IsIncumbent	`xs:boolean`	Optional	Single	Indicates whether the candidate is the incumbent for the office associated with the contest.	If the field is invalid or not present, then the implementation is required to ignore it.
IsTopTicket	`xs:boolean`	Optional	Single	Indicates whether the candidate is the top of a ticket that includes multiple candidates.	If the field is invalid or not present, then the implementation is required to ignore it.
PartyId	`xs:IDREF`	Optional	Single	Reference to a *Party* element with additional information about the candidate's affiliated party. This is the party affiliation that is intended to be presented as part of ballot information.	If the field is invalid or not present, then the implementation is required to ignore it.
PersonId	`xs:IDREF`	Optional	Single	Reference to a *Person* element with additional information about the candidate.	If the field is invalid or not present, then the implementation is required to ignore it.
PostElectionStatus	*CandidatePostElectionStatus*	Optional	Single	Final status of the candidate (e.g. winner, withdrawn, etc. . .).	If the field is invalid or not present, then the implementation is required to ignore it.
PreElectionStatus	*CandidatePreElectionStatus*	Optional	Single	Registration status of the candidate (e.g. filed, qualified, etc. . .).	If the field is invalid or not present, then the implementation is required to ignore it.

```
<Candidate id="can10961">
   <BallotName>
     <Text language="en">Ken T. Cuccinelli II</Text>
   </BallotName>
   <PartyId>par0001</PartyId>
   <PersonId>per10961</PersonId>
</Candidate>
```

PollingLocation

The PollingLocation object represents a site where voters cast or drop off ballots.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
AddressLine	`xs:string`	**Required**	Repeats	Represents the various parts of an address to a polling location.	At least one valid `AddressLine` must be present for `PollingLocation` to be valid. If no valid `AddressLine` is present, the implementation is required to ignore the `PollingLocation` element containing it.
Directions	*InternationalizedText*	Optional	Single	Specifies further instructions for locating the polling location.	If the element is invalid or not present, then the implementation is required to ignore it.
Hours **[deprecated]**	*InternationalizedText*	Optional	Single	Contains the hours (in local time) that the polling location is open (**NB:** this element is deprecated in favor of the more structured *HoursOpen* element. It is strongly encouraged that data providers move toward contributing hours in this format).	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	`xs:IDREF`	Optional	Single	Links to an *HoursOpen* element, which is a schedule of dates and hours during which the polling location is available.	If the field is invalid or not present, then the implementation is required to ignore it.
IsDropBox	`xs:boolean`	Optional	Single	Indicates if this polling location is a drop box.	If the field is invalid or not present, then the implementation is required to ignore it.
IsEarlyVoting	`xs:boolean`	Optional	Single	Indicates if this polling location is an early vote site.	If the field is invalid or not present, then the implementation is required to ignore it.
LatLng	*LatLng*	Optional	Single	Specifies the latitude and longitude of this polling location.	If the element is invalid or not present, then the implementation is required to ignore it.
Name	`xs:string`	Optional	Single	Name of the polling location.	If the field is invalid or not present, then the implementation is required to ignore it.
PhotoUri	`xs:anyURI`	Optional	Single	Contains a link to an image of the polling location.	If the field is invalid or not present, then the implementation is required to ignore it.

Locality

The Locality object represents the jurisdiction below the *State* (e.g. county).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ElectionAdministrationId	xs:IDREF	Optional	Single	Links to the locality's *ElectionAdministration* object.	If the field is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Another identifier for a locality that links to another dataset (e.g. OCD-ID)	If the element is invalid or not present, then the implementation is required to ignore it.
Name	xs:string	**Required**	Single	Specifies the name of a locality.	If the field is not present or invalid, the implementation is required to ignore the Locality element containing it.
PollingLocationIds	xs:IDREFS	Optional	Single	Specifies a link to a set of the locality's :ref:'polling locations <single-xml-polling-location>'s. If early vote centers or ballot drop locations are locality-wide, they should be specified here.	If the field is invalid or not present, the implementation is required to ignore it. However, the implementation should still check to see if there are any polling locations associated with this locality's state.
StateId	xs:IDREF	**Required**	Single	References the locality's *State*.	If the field is invalid or not present, the implementation is required to ignore the Locality element containing.
Type	*DistrictType*	Optional	Single	Defines the kind of locality (e.g. county, town, et al.), which is one of the various *DistrictType enumerations*.	If the field is invalid or not present, then the implementation is required to ignore it.
OtherType	xs:string	Optional	Single	Allows for defining a type of locality that falls outside the options listed in *DistrictType*.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<Locality id="loc70001">
  <ElectionAdministrationId>ea40001</ElectionAdministrationId>
  <ExternalIdentifiers>
    <ExternalIdentifier>
      <Type>ocd-id</Type>
      <Value>ocd-division/country:us/state:va/county:albemarle</Value>
    </ExternalIdentifier>
  </ExternalIdentifiers>
  <Name>ALBEMARLE COUNTY</Name>
  <StateId>st51</StateId>
  <Type>county</Type>
</Locality>
```

Precinct

The Precinct object represents a precinct, which is contained within a Locality. While the id attribute does not have to be static across feeds for one election, the combination of *Source.VipId*, *Locality.Name*, *Precinct.Ward*, *Precinct.Name*, and *Precinct.Number* should remain constant across feeds for one election (NB: not all of the fields just mentioned are required – omitting those non-required fields is fine).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
BallotStyleId	xs:IDREF	Optional	Single	Links to the *BallotStyle*, which a person who lives in this precinct will vote.	If the field is invalid or not present, then the implementation is required to ignore it.
ElectoralDistrictIds	xs:IDREFS	Optional	Single	Links to the :ref:'single-xml-electoral-district's (e.g., congressional district, state house district, school board district) to which the entire precinct/precinct split belongs. **Highly Recommended** if candidate information is to be provided.	If the field is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifier for the precinct that relates to another dataset (e.g. OCD-ID).	If the element is invalid or not present, then the implementation is required to ignore it.
IsMailOnly	xs:boolean	Optional	Single	Determines if the precinct runs mail-only elections.	If the field is missing or invalid, the implementation is required to assume *IsMailOnly* is false.
LocalityId	xs:IDREF	**Required**	Single	Links to the *Locality* that comprises the precinct.	If the field is invalid or not present, the implementation is required to ignore the precinct element containing it.
Name	xs:string	**Required**	Single	Specifies the precinct's name (or number if no name exists).	If the field is invalid or not present, the implementation is required to ignore the precinct element containing it.
Number	xs:string	Optional	Single	Specifies the precinct's number (e.g., 32 or 32A – alpha characters are legal). Should be used if the *Name* field is populated by a name and not a number.	If the field is invalid or not present, then the implementation is required to ignore it.
PollingLocationIds	xs:IDREFS	Optional	Single	Specifies a link to the precinct's *PollingLocation* object(s).	If the field is invalid or not present, then the implementation is required to ignore it.
PrecinctSplitName	xs:string	Optional	Single	If this field is empty, then this *Precinct* object represents a full precinct. If this field is present, then this *Precinct* object represents one portion of a split precinct. Each *Precinct* object that represents one portion of a split precinct **must** have the same *Name* value, but different *PrecinctSplitName* values. See the *sample_feed.xml* file for examples.	If the field is invalid or not present, then the implementation is required to ignore it.
Ward	xs:string	Optional	Single	Specifies the ward the precinct is contained within.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<Precinct id="pre90111">
   <BallotStyleId>bs00010</BallotStyleId>
   <ElectoralDistrictIds>ed60129 ed60311 ed60054</ElectoralDistrictIds>
   <IsMailOnly>false</IsMailOnly>
   <LocalityId>loc70001</LocalityId>
   <Name>203 - GEORGETOWN</Name>
   <Number>0203</Number>
   <PollingLocationIds>pl81274</PollingLocationIds>
</Precinct>
<!--
  Precinct split. Name and PollingLocationIds are the same but
  PrecinctSplitName is present, the ElectoralDistrictIds are different,
  and the BallotStyleId is different.
-->
<Precinct id="pre90348sp0000">
  <BallotStyleId>bs00002</BallotStyleId>
  <ElectoralDistrictIds>ed60129 ed60054 ed60150</ElectoralDistrictIds>
  <IsMailOnly>false</IsMailOnly>
  <LocalityId>loc70001</LocalityId>
  <Name>201 - JACK JOUETT</Name>
  <Number>0201</Number>
  <PollingLocationIds>pl00000 pl81273 pl81662</PollingLocationIds>
  <PrecinctSplitName>0000</PrecinctSplitName>
</Precinct>
<Precinct id="pre90348sp0001">
  <BallotStyleId>bs00015</BallotStyleId>
  <ElectoralDistrictIds>ed60129 ed60054 ed60267</ElectoralDistrictIds>
  <IsMailOnly>false</IsMailOnly>
  <LocalityId>loc70001</LocalityId>
  <Name>201 - JACK JOUETT</Name>
  <Number>0201</Number>
  <PollingLocationIds>pl00000 pl81273 pl81662</PollingLocationIds>
  <PrecinctSplitName>0001</PrecinctSplitName>
</Precinct>
```

OrderedContest

`OrderedContest` encapsulates links to the information that comprises a contest and potential ballot selections. `OrderedContest` elements can be collected within a *BallotStyle* to accurate depict exactly what will show up on a particular ballot in the proper order.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ContestId	`xs:IDREF`	**Required**	Single	Links to elements that extend *ContestBase*.	If the field is invalid or not present, the implementation is required to ignore the `OrderedContest` element containing it.
OrderedBallotSelectionIds	`xs:IDREFS`	Optional	Single	Links to elements that extend *BallotSelectionBase*.	If the field is invalid or not present, the implementation is required to ignore it. If an `OrderedBallotSelectionIds` element is not present, the presumed order of the selection will be the order of *BallotSelectionBase*-extended elements referenced by the underlying *ContestBase*-extended elements.

```
<OrderedContest id="oc20003abc">
   <ContestId>cc20003</ContestId>
   <OrderedBallotSelectionIds>cs10961 cs10962 cs10963</OrderedBallotSelectionIds>
</OrderedContest>
```

PartyContest

An extension of *ContestBase* which describes a contest in which the possible ballot selections are of type *PartySelection*. These could include contests in which straight-party selections are allowed, or party-list contests (although these are more common outside of the United States).

ElectionAdministration

The Election Administration represents an institution for serving a locality's (or state's) election functions.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
AbsenteeUri	xs:anyURI	Optional	Single	Specifies the web address for information on absentee voting.	If the field is invalid or not present, then the implementation is required to ignore it.
AmIRegisteredUri	xs:anyURI	Optional	Single	Specifies the web address for information on whether an individual is registered.	If the field is invalid or not present, then the implementation is required to ignore it.
Department	*Department*	**Required**	Repeats	Describes the administrative body for a particular voter service.	There must be at least one valid *Department* in each *ElectionAdministration* element. If no valid *Department* objects are present, the implementation is required to ignore the *ElectionAdministration* object that contains it/them.
ElectionsUri	xs:anyURI	Optional	Single	Specifies web address the administration's website.	If the field is invalid or not present, then the implementation is required to ignore it.
RegistrationUri	xs:anyURI	Optional	Single	Specifies web address for information on registering to vote.	If the field is invalid or not present, then the implementation is required to ignore it.
RulesUri	xs:anyURI	Optional	Single	Specifies a URI for the election rules and laws (if any) for the jurisdiction of the administration.	If the field is invalid or not present, then the implementation is required to ignore it.
WhatIsOnMyBallotUri	xs:anyURI	Optional	Single	Specifies web address for information on what is on an individual's ballot.	If the field is invalid or not present, then the implementation is required to ignore it.
WhereDoIVoteUri	xs:anyURI	Optional	Single	The Specifies web address for information on where an individual votes based on their address.	If the field is invalid or not present, then the implementation is required to ignore it.

Department

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ContactInformation	*ContactInformation*	Optional	Single	Contact and physical address information for the election administration body (see *ContactInformation*).	If the element is invalid or not present, then the implementation is required to ignore it.
ElectionOfficialPersonId	xs:IDREF	Optional	Single	The individual to contact at the election administration office. The specified person should be the *election official*.	If the field is invalid or not present, then the implementation is required to ignore it.
VoterService	*VoterService*	Optional	Repeats	The types of services and appropriate contact individual available to voters.	If the element is invalid or not present, then the implementation is required to ignore it.

VoterService

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ContactInformation	*ContactInformation*	Optional	Single	The contact for a particular voter service.	If the element is invalid or not present, then the implementation is required to ignore it.
Description	*InternationalizedText*	Optional	Single	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
ElectionOfficialPersonId	xs:IDREF	Optional	Single	The *authority* for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
Type	*VoterServiceType*	Optional	Single	The type of *voter service*.	If the field is invalid or not present, then the implementation is required to ignore it.
OtherType	xs:string	Optional	Single	If Type is "other", OtherType allows for cataloging another type of voter service.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<ElectionAdministration id="ea40133">
   <AbsenteeUri>http://www.sbe.virginia.gov/absenteevoting.html</AbsenteeUri>
   <AmIRegisteredUri>https://www.vote.virginia.gov/</AmIRegisteredUri>
   <Department>
     <ContactInformation label="ci60000">
       <AddressLine>Washington Building First Floor</AddressLine>
       <AddressLine>1100 Bank Street</AddressLine>
       <AddressLine>Richmond, VA 23219</AddressLine>
       <Name>State Board of Elections</Name>
     </ContactInformation>
   </Department>
   <ElectionsUri>http://www.sbe.virginia.gov/</ElectionsUri>
   <RegistrationUri>https://www.vote.virginia.gov/</RegistrationUri>
   <RulesUri>http://www.sbe.virginia.gov/</RulesUri>
```

```
   <WhatIsOnMyBallotUri>https://www.vote.virginia.gov/</WhatIsOnMyBallotUri>
   <WhereDoIVoteUri>https://www.vote.virginia.gov/</WhereDoIVoteUri>
</ElectionAdministration>
```

RetentionContest

`RetentionContest` extends *BallotMeasureContest* and represents a contest where a candidate is retained in a position (e.g. a judge).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
CandidateId	`xs:IDREF`	**Required**	Single	Links to the *Candidate* being retained.	If the field is invalid or not present, the implementation is required to ignore the `RetentionContest` element containing it.
OfficeId	`xs:IDREF`	Optional	Single	Links to the information about the office.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<RetentionContest id="rc40001">
   <BallotSelectionIds>rc40001a rc40001b</BallotSelectionIds>
   <BallotTitle>
      <Text language="en">Retention of Supreme Court Justice</Text>
      <Text language="es">La retención de juez de la Corte Suprema</Text>
   </BallotTitle>
   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
   <Name>Judicial Retention, Supreme Court</Name>
   <CandidateId>can14444</CandidateId>
   <OfficeId>off20006</OfficeId>
</RetentionContest>
```

StreetSegment

A Street Segment objection represents a portion of a street and the links to the precinct that this geography (i.e., segment) is contained within. The start address house number must be less than the end address house number unless the segment consists of only one address, in which case these values are equal.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
AddressDirection	xs:string	Optional	Single	Specifies the (inter-)cardinal direction of the entire address. An example is "NE" for the address "100 E Capitol St NE."	If the field is invalid or not present, then the implementation is required to ignore it.
City	xs:string	**Required**	Single	The city specifies the city or town of the address.	If the field is invalid, then the implementation is required to ignore it.
IncludesAllAddresses	xs:boolean	Optional	Single	Specifies if the segment covers every address on this street. If this is *true*, then the values of **StartHouseNumber** and **EndHouseNumber** should be ignored. The value of **OddEvenBoth** must be *both*.	If the field is invalid or not present, then the implementation is required to ignore it.
IncludesAllStreets	xs:boolean	Optional	Single	Specifies if the segment covers every street in this city. If this is *true*, then the values of **OddEvenBoth**, **StartHouseNumber**, **EndHouseNumber**, **StreetName**, and **Zip** should be ignored.	If the field is invalid or not present, then the implementation is required to ignore it.
OddEvenBoth	*OebEnum*	Optional	Single	Specifies whether the odd side of the street (in terms of house numbers), the even side, or both are in included in the street segment.	If the field is not present or invalid, the implementation is required to ignore the StreetSegment containing it.
PrecinctId	xs:IDREF	Optional	Single	References the *Precinct* that contains the entire street segment.	If the field is not present or invalid, the implementation is required to ignore the StreetSegment element containing it.
StartHouseNumber	xs:integer	Optional	Single	The house number at which the street segment starts. This value is necessary for the street segment to make any sense. Unless **IncludesAllAddresses** or **IncludesAllStreets** are true, this value must be less than or equal to **EndHouseNumber**. If **IncludesAllAddresses** or **IncludesAllStreets** are true, this value is ignored.	Unless **IncludesAllAddresses** or **IncludesAllStreets** are true, if the field is not present or invalid, the implementation is required to ignore the StreetSegment element containing it. If the **StartHouseNumber** is greater than the **EndHouseNumber**, the implementation should ignore the element containing them.
EndHouseNumber	xs:integer	Optional	Single	The house number at which the street segment ends. This value is necessary for the street segment to make any sense. Unless **IncludesAllAddresses** or **IncludesAllStreets** are true, it must be greater than or equal to **StartHouseNumber**. If **IncludesAllAddresses** or **IncludesAllStreets** are true, this value is ignored.	Unless **IncludesAllAddresses** or **IncludesAllStreets** are true, if the field is not present or invalid, the implementation is required to ignore the StreetSegment element containing it. If the **EndHouseNumber** is less than the **StartHouseNumber**, the implementation should ignore the element containing it.
State	xs:string	**Required**	Single	Specifies the two-letter state abbreviation of the address.	If the field is invalid, then the implementation is required to ignore it.
StreetDirection	xs:string	Optional	Single	Specifies the (inter-)cardinal direction of the street address (e.g., the "E" in "100 E Capitol St NE").	If the field is invalid or not present, then the implementation is required to ignore it.
StreetName	xs:string	Optional	Single	Represents the name of the street for the address. A special wildcard, "*", denotes every street in the given city/town. It optionally may contain street direction, street suffix or address direction (e.g., both "Capitol" and "E Capitol St NE" are acceptable for the address "100 E Capitol St NE"), however this is not preferred. Preferred is street name alone (e.g.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<StreetSegment id="ss999999">
   <City>Charlottesville</City>
   <IncludesAllAddresses>true</IncludesAllAddresses>
   <OddEvenBoth>both</OddEvenBoth>
   <PrecinctId>pre99999</PrecinctId>
   <State>VA</State>
   <StreetName>CHAPEL HILL</StreetName>
   <StreetSuffix>RD</StreetSuffix>
   <Zip>22901</Zip>
</StreetSegment>
<StreetSegment id="ss309904">
   <City>GREENWOOD</City>
   <OddEvenBoth>both</OddEvenBoth>
   <PrecinctId>pre92145</PrecinctId>
   <StartHouseNumber>1</StartHouseNumber>
   <EndHouseNumber>201</EndHouseNumber>
   <State>VA</State>
   <StreetName>MISTY MOUNTAIN</StreetName>
   <StreetSuffix>RD</StreetSuffix>
   <Zip>22943</Zip>
</StreetSegment>
```

CandidateContest

CandidateContest extends *ContestBase* and represents a contest among candidates.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
NumberElected	xs:integer	Optional	Single	Number of candidates that are elected in the contest (i.e. "N" of N-of-M).	If the field is invalid or not present, then the implementation is required to ignore it.
OfficeIds	xs:IDREFS	Optional	Single	References a set of *Office* elements, if available, which give additional information about the offices. **Note:** the order of the office IDs **must** be in the same order as the candidates listed in *BallotSelectionIds*. E.g., if the various *BallotSelectionIds* reference *CandidateSelection* elements which reference the candidate for President first and Vice-President second, the *OfficeIds* should reference the office of President first and the office of Vice-President second.	If the field is invalid or not present, then the implementation is required to ignore it.
PrimaryPartyIds	xs:IDREFS	Optional	Single	References *Party* elements, if the contest is related to a particular party.	If the field is invalid or not present, then the implementation is required to ignore it.
VotesAllowed	xs:integer	Optional	Single	Maximum number of votes/write-ins per voter in this contest.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<CandidateContest id="cc20003">
   <BallotSelectionIds>cs10961 cs10962 cs10963</BallotSelectionIds>
   <BallotTitle>
     <Text language="en">Governor of Virginia</Text>
   </BallotTitle>
   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
   <Name>Governor</Name>
   <NumberElected>1</NumberElected>
   <OfficeId>off0000</OfficeId>
   <VotesAllowed>1</VotesAllowed>
</CandidateContest>
```

Party

This element describes a political party and the metadata associated with them. These can also include "dummy" parties to indicate a type of contest (e.g., a Voter Nominated *CandidateContest* can use the **PrimaryPartyIds** field and a dummy Party object to indicate that the contest is a "Top-Two" primary).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Abbreviation	`xs:string`	Optional	Single	An abbreviation for the party name.	If the field is invalid or not present, then the implementation is required to ignore it.
Color	*HtmlColorString*	Optional	Single	The preferred display color for the party, for use in maps and other displays.	If the element is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifiers that link this party to other related data sets (e.g. a campaign finance system, etc).	If the element is invalid or not present, then the implementation is required to ignore it.
IsWriteIn	`xs:boolean`	Optional	Single	Signals if this political party is one that is officially recognized by a local, state, or federal organization, or is a "write-in" in jurisdictions which allow candidates to free-form enter their political affiliation. If this field is not present then it is assumed to be false.	If the field is invalid or not present, then the implementation is required to ignore it.
LogoUri	`xs:anyURI`	Optional	Single	Web address of a logo to use in displays.	If the field is invalid or not present, then the implementation is required to ignore it.
Name	*InternationalizedText*	Optional	Single	The name of the party.	If the element is invalid or not present, then the implementation is required to ignore it.

HtmlColorString

A restricted string pattern for a six-character hex code representing an HTML color string. The pattern is:

`[0-9a-f]{6}`

```
<Party id="par0001">
  <Abbreviation>REP</Abbreviation>
  <Color>e91d0e</Color>
  <Name>
    <Text language="en">Republican</Text>
  </Name>
</Party>
```

InternationalizedText

`InternationalizedText` allows for support of multiple languages for a string. `InternationalizedText` has an optional attribute `label`, which allows the feed to refer back to the original label for the information (e.g. if the contact information came from a CSV, `label` may refer to a row ID). Examples of `InternationalizedText` can be seen in:

- Any element that extends *ContestBase*
- Any element that extends *BallotSelectionBase*
- *Candidate*
- *ContactInformation*
- *Election*
- *ElectionAdministration*
- *Office*
- *Party*
- *Person*
- *PollingLocation*
- *Source*

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Text	*LanguageString*	**Required**	Repeats	Contains the translations of a particular string of text.	At least one valid `Text` must be present for `InternationalizedText` to be valid. If no valid `Text` is present, the implementation is required to ignore the `InternationalizedText` element.

LanguageString

`LanguageString` extends xs:string and can contain text from any language. `LanguageString` has one required attribute, `language`, that must contain the 2-character language code for the type of language `LanguageString` contains.

```
<BallotTitle>
   <Text language="en">Retention of Supreme Court Justice</Text>
   <Text language="es">La retención de juez de la Corte Suprema</Text>
</BallotTitle>
```

BallotMeasureSelection

Represents the possible selection (e.g. yes/no, recall/do not recall, et al) for a *BallotMeasureContest* that would appear on the ballot. BallotMeasureSelection extends *BallotSelectionBase*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Selection	*InternationalizedText*	**Required**	Single	Selection text for a *BallotMeasureContest*	If the element is invalid or not present, the implementation is required to ignore the BallotMeasureSelection containing it.

```
<BallotMeasureSelection id="bms30001a">
   <Selection label="bms30001at">
      <Text language="en">Yes</Text>
      <Text language="es">Sí</Text>
   </Selection>
</BallotMeasureSelection>
<BallotMeasureSelection id="bms30001b">
   <Selection label="bms30001bt">
      <Text language="en">No</Text>
      <Text language="es">No</Text>
   </Selection>
</BallotMeasureSelection>
```

HoursOpen

A structured way of describing the days and hours that a place such as a *Office* or *PollingLocation* is open, or that an event such as an *Election* is happening. The range of days indicated by the *StartDate* and *EndDate* in each *Schedule* element should not overlap with peer *Schedule* elements. For example, it is invalid to specify a schedule from 10/01/2016 to 10/31/2016 and also specify a schedule from 10/10/2016 to 10/11/2016 within the same *HoursOpen* element.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Schedule	*Schedule*	**Required**	Repeats	Defines a block of days and hours that a place will be open.	At least one valid *Schedule* must be present for `HoursOpen` to be valid. If no valid *Schedule* is present, the implementation is required to ignore the `HoursOpen` element.

Schedule

A sub-portion of the schedule. This describes a range of days, along with one or more set of open and close times for those days, as well as the options describing whether or not appointments are necessary or possible.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Hours	*Hours*	Optional	Repeats	Blocks of hours in the date range in which the place is open.	If the element is invalid or not present, then the implementation is required to ignore it.
IsOnlyByAppointment	`xs:boolean`	Optional	Single	If true, the place is only open during the specified time window with an appointment.	If the field is invalid or not present, then the implementation is required to ignore it.
IsOrByAppointment	`xs:boolean`	Optional	Single	If true, the place is open during the hours specified time window and may also be open with an appointment.	If the field is invalid or not present, then the implementation is required to ignore it.
IsSubjectToChange	`xs:boolean`	Optional	Single	If true, the place should be open during the specified time window, but may be subject to change. People should contact prior to arrival to confirm hours are still accurate.	If the field is invalid or not present, then the implementation is required to ignore it.
StartDate	`xs:date`	Optional	Single	The date at which this collection of start and end times and options begin.	If the field is invalid or not present, then the implementation is required to ignore it.
EndDate	`xs:date`	Optional	Single	The date at which this collection of start and end times and options end.	If the field is invalid or not present, then the implementation is required to ignore it.

Hours

The open and close time for this place. All times must be fully specified, including a timezone offset from UTC.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
StartTime	*TimeWithZone*	Optional	Single	The time at which this place opens.	If the element is invalid or not present, then the implementation is required to ignore it.
EndTime	*TimeWithZone*	Optional	Single	The time at which this place closes.	If the element is invalid or not present, then the implementation is required to ignore it.

TimeWithZone

A string pattern restricting the value to a time with an included offset from UTC. The pattern is

```
(([01][0-9]|2[0-3]):[0-5][0-9]:[0-5][0-9]|(24:00:00))(Z|[+-]((0[0-9]|1[0-3]):[0-5][0-9]|14
```

```
<HoursOpen id="hours0001">
  <Schedule>
    <Hours>
      <StartTime>06:00:00-05:00</StartTime>
      <EndTime>12:00:00-05:00</EndTime>
```

```
    </Hours>
    <Hours>
      <StartTime>13:00:00-05:00</StartTime>
      <EndTime>19:00:00-05:00</EndTime>
    </Hours>
    <StartDate>2013-11-05</StartDate>
    <EndDate>2013-11-05</EndDate>
  </Schedule>
</HoursOpen>
```

ElectoralDistrict

The `ElectoralDistrict` object represents the geographic area in which contests are held. Examples of `ElectoralDistrict` include: "the state of Maryland", "Virginia's 5th Congressional District", or "Union School District". The geographic area that comprises a `ElectoralDistrict` is defined by which precincts link to the `ElectoralDistrict`.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifiers that link to external datasets (e.g. OCD-IDs)	If the element is invalid or not present, then the implementation is required to ignore it.
Name	`xs:string`	**Required**	Single	Specifies the electoral area's name.	If the field is invalid or not present, then the implementation is required to ignore the `ElectoralDistrict` object containing it.
Number	`xs:integer`	Optional	Single	Specifies the district number of the district (e.g. 34, in the case of the 34th State Senate District). If a number is not applicable, instead of leaving the field blank, leave this field out of the object; empty strings are not valid for xs:integer fields.	If the field is invalid or not present, then the implementation is required to ignore it.
Type	*DistrictType*	**Required**	Single	Specifies the type of electoral area.	If the field is invalid or not present, then the implementation is required to ignore the `ElectoralDistrict` object containing it.
OtherType	`xs:string`	Optional	Single	Allows for cataloging a new *DistrictType* option when `Type` is specified as "other".	If the field is invalid or not present, then the implementation is required to ignore it.

```
<ElectoralDistrict id="ed60129">
   <ExternalIdentifiers>
     <ExternalIdentifier>
       <Type>ocd-id</Type>
       <Value>ocd-division/country:us/state:va</Value>
     </ExternalIdentifier>
     <ExternalIdentifier>
       <Type>fips</Type>
```

```
        <Value>51</Value>
      </ExternalIdentifier>
    </ExternalIdentifiers>
    <Name>Commonwealth of Virginia</Name>
    <Type>state</Type>
</ElectoralDistrict>
```

PartySelection

This element extends *BallotSelectionBase* to support contests in which the selections can be groups of one or more parties.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
PartyIds	xs:IDREFS	**Required**	Single	One or more *Party* IDs which collectively represent a ballot selection.	If one or more parties referenced are invalid or not present, the implementation is required to ignore the PartySelection containing it.

Enumerations

IdentifierType

Tag	Description
fips	Federal Information Processing Standards codes for states, counties, and cities.
local-level	An identifier generated or used by local governments or organizations.
national-level	An identifier generated or used by national organizations.
ocd-id	An Open Civic Data Division Identifier.
state-level	An identifier generated or used by state governments or organizations.
other	Any identifier which doesn't fall into any of the above categories.

CandidatePostElectionStatus

Tag	Description
advanced-to-runoff	For contests in which the top *N* candidates advance to the next round.
projected-winner	A candidate is expected to win, but official results are not yet complete.
winner	The candidate has officially won.
withdrawn	The candidate has withdrawn from the contest.

CandidatePreElectionStatus

Tag	Description
filed	The candidate has filed for office but not yet been qualified.
qualified	The candidate has qualified for the contest.
withdrawn	The candidate has withdrawn from the contest (but may still be on the ballot).
write-in	

VoterServiceType

Tag	Description
absentee-ballots	This department handles the dispatch, tracking, and return of absentee ballots.
overseas-voting	The department for overseas, military, and other outside-the-U.S. voters.
polling-places	This deparment handles the selection and management of polling places.
voter-registration	The deparment that manages voter registration.
other	Any other service not covered by the above descriptions.

DistrictType

Enumeration describing the set of possible jurisdiction and district types. Please use the enumeration value which most accurately reflects the type of district or jurisdiction in your state or county. For example, "town" and "township" may mean different things – or not be defined at all – in your state, so please use the definition which best matches your local meaning.

Tag	Description
borough	A borough
city	A city.
city-council	A specific seat/jurisdiction for a city, town, or village council.
congressional	A United States congressional district.
county	A county.
county-council	A county council district, either in its entirety or for a specific seat.
judicial	A judicial district.
municipality	A civil division which is not a town, city, village, or county.
national	The United States.
school	A school district.
special	A special-purpose district that exist separate from general-purpose districts.
state	A state, district, commonwealth, or U.S. territory.
state-house	The lower house of a state legislature.
state-senate	The upper house of a state legislature.
town	A town.
township	A township, which may be different than a town. See the Wikipedia article.
utility	A non-water public or municipal utility district.
village	A village district.
ward	A ward.
water	A water district.
other	Any district not described above. Use the *OtherType* field to describe it.

OfficeTermType

Tag	Description
full-term	This election is for an office for which the existing term has been completed.
unexpired-term	This election is for an office for which the original term is not yet complete.

OebEnum

Tag	Description
both	Both even and odd addresses within the range.
even	Only even-numbered addresses within the range.
odd	Only odd-numbered addresses within the range.

BallotMeasureType

A list of the various types of ballot measures. States may have different legal definitions of each type; Wikipedia has more details about each type. These values are to help states with multiple types of non-candidate-based contests distinguish between each type; as such, the definitions in this table are simple guidelines. Ultimately it is up to the state or local election official to choose the value which best describes the ballot measure(s) in their jurisdiction.

Tag	Description
ballot-measure	A catch-all for generic types of non-candidate-based contests.
initiative	These are usually citizen-driven measures to be placed on the ballot. These could include both statutory changes and constitutional amendments.
referendum	These could include measures to repeal existing acts of legislation, legislative referrals, and legislatively-referred state constitutional amendments.
other	Anything that does not fall into the above categories.

VoteVariation

Note that the descriptions below describe what the enumeration names stand for in the context of the VIP spec, rather than provide general definitions of the election terms that the names correspond to. For example, even though there are majority voting methods that are not "1-of-m" (e.g. ranked choice voting), we constrain "majority" to 1-of-m. We do this to eliminate any source of ambiguity when a single enumeration value needs to be assigned to a contest.

Tag	Description
1-of-m	A method where each voter can select up to one option.
approval	Approval voting, where each voter can select as many options as desired.
borda	Borda count, where each voter can rank the options, and the rankings are assigned point values.
cumulative	Cumulative voting, where each voter can distribute their vote to up to *N* options.
majority	A 1-of-m method where the winner needs more than 50% of the vote to be elected.
n-of-m	A method where each voter can select up to *N* options.
plurality	A 1-of-m method where the option with the most votes is elected, regardless of whether the option has more than 50% of the vote.
proportional	A proportional representation method (other than STV), which is any system that elects winners in proportion to the total vote.
range	Range voting, where each voter can select a score for each option.
rcv	Ranked choice voting (RCV), where each voter can rank the options, and the ballots are counted in rounds. Also known as instant-runoff voting (IRV) and the single transferable vote (STV).
supermajority	A 1-of-m method where the winner needs more than some predetermined fraction of the vote to be elected, where the fraction is more than 50% (e.g. three-fifths or two-thirds).
other	Used when the vote variation type is not included in this enumeration.

Elements (Separate Pages)

BallotMeasureContest

The BallotMeasureContest provides information about a ballot measure before the voters, including summary statements on each side. Extends *ContestBase*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ConStatement	*InternationalizedText*	Optional	Single	Specifies a statement in opposition to the referendum. It does not necessarily appear on the ballot.	If the element is invalid or not present, then the implementation is required to ignore it.
EffectOfAbstain	*InternationalizedText*	Optional	Single	Specifies what effect abstaining (i.e. not voting) on this proposition will have (i.e. whether abstaining is considered a vote against it).	If the element is invalid or not present, then the implementation is required to ignore it.
FullText	*InternationalizedText*	Optional	Single	Specifies the full text of the referendum as it appears on the ballot.	If the element is invalid or not present, then the implementation is required to ignore it.
InfoUri	`xs:anyURI`	Optional	Single	Specifies a URI that links to additional information about the referendum.	If the field is invalid or not present, then the implementation is required to ignore it.
PassageThreshold	*InternationalizedText*	Optional	Single	Specifies the threshold of votes that the referendum needs in order to pass. The default is a simple majority (i.e. 50% plus one vote). Other common thresholds are "three-fifths" and "two-thirds". If there are competing initiatives, information about their effect on the passage of the BallotMeasureContest would go here.	If the element is invalid or not present, then the implementation is required to ignore it.
ProStatement	*InternationalizedText*	Optional	Single	Specifies a statement in favor of the referendum. It does not necessarily appear on the ballot.	If the element is invalid or not present, then the implementation is required to ignore it.
SummaryText	*InternationalizedText*	Optional	Single	Specifies a short summary of the referendum that is on the ballot, below the title, but above the text.	If the element is invalid or not present, then the implementation is required to ignore it.
Type	*BallotMeasureType*	Optional	Single	Specifies the particular type of ballot measure. Must be one of the valid *BallotMeasureType* options.	If the field is invalid or not present, then the implementation is required to ignore it.
OtherType	`xs:string`	Optional	Single	Allows for cataloging a new *BallotMeasureType* option, when Type is specified as "other."	If the field is invalid or not present, then the

```
<BallotMeasureContest id="bmc30001">
   <BallotSelectionIds>bms30001a bms30001b</BallotSelectionIds>
   <BallotTitle>
      <Text language="en">State of the State</Text>
      <Text language="es">Estado del Estado.</Text>
   </BallotTitle>
   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
   <Name>Referendum on Virginia</Name>
   <ConStatement label="bmc30001con">
      <Text language="en">This is no good.</Text>
      <Text language="es">Esto no es bueno.</Text>
   </ConStatement>
   <EffectOfAbstain label="bmc30001abs">
      <Text language="en">Nothing will happen.</Text>
      <Text language="es">Nada pasará.</Text>
   </EffectOfAbstain>
   <ProStatement label="bmc30001pro">
      <Text language="en">Everything will be great.</Text>
      <Text language="es">Todo va a estar bien.</Text>
   </ProStatement>
   <Type>referendum</Type>
</BallotMeasureContest>
```

BallotMeasureSelection

Represents the possible selection (e.g. yes/no, recall/do not recall, et al) for a *BallotMeasureContest* that would appear on the ballot. BallotMeasureSelection extends *BallotSelectionBase*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Selection	*InternationalizedText*	**Required**	Single	Selection text for a *BallotMeasureContest*	If the element is invalid or not present, the implementation is required to ignore the BallotMeasureSelection containing it.

```
<BallotMeasureSelection id="bms30001a">
   <Selection label="bms30001at">
      <Text language="en">Yes</Text>
      <Text language="es">Sí</Text>
   </Selection>
</BallotMeasureSelection>
<BallotMeasureSelection id="bms30001b">
   <Selection label="bms30001bt">
      <Text language="en">No</Text>
      <Text language="es">No</Text>
   </Selection>
</BallotMeasureSelection>
```

BallotSelectionBase

A base model for all ballot selection types: *BallotMeasureSelection*, *CandidateSelection*, and *PartySelection*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
SequenceOrder	xs:integer	Optional	Single	The order in which a selection can be listed on the ballot or in results. This is the default ordering, and can be overridden by *OrderedBallotSlectionIds* in *OrderedContest*.	If the field is invalid or not present, then the implementation is required to ignore it.

BallotStyle

A container for the contests/measures on the ballot.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ImageUri	xs:anyURI	Optional	Single	Specifies a URI that returns an image of the sample ballot.	If the field is invalid or not present, then the implementation is required to ignore it.
OrderedContestIds	xs:IDREFS	Optional	Single	Reference to a set of :ref:‘multi-xml-ordered-contest‘s	If the field is invalid or not present, then the implementation is required to ignore it.
PartyIds	xs:IDREFS	Optional	Single	Reference to a set of :ref:‘multi-xml-party‘s.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<BallotStyle id="bs00000">
   <OrderedContestIds>oc20003 oc20004 oc20005 oc20025 oc20355 oc20449</
↪OrderedContestIds>
</BallotStyle>
```

Candidate

The Candidate object represents a candidate in a contest. If a candidate is running in multiple contests, each contest **must** have its own Candidate object. Candidate objects may **not** be reused between Contests.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
BallotName	*InternationalizedText*	**Required**	Single	The candidate's name as it will be displayed on the official ballot (e.g. "Ken T. Cuccinelli II").	If the element is invalid or not present, then the implementation is required to ignore the Candidate element containing it.
ContactInformation	*ContactInformation*	Optional	Single	Contact and physical address information for this Candidate and/or their campaign (see *ContactInformation*).	If the element is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Another identifier for a candidate that links to another source of information (e.g. a campaign committee ID that links to a campaign finance system).	If the element is invalid or not present, then the implementation is required to ignore it.
FileDate	`xs:date`	Optional	Single	Date when the candidate filed for the contest.	If the field is invalid or not present, then the implementation is required to ignore it.
IsIncumbent	`xs:boolean`	Optional	Single	Indicates whether the candidate is the incumbent for the office associated with the contest.	If the field is invalid or not present, then the implementation is required to ignore it.
IsTopTicket	`xs:boolean`	Optional	Single	Indicates whether the candidate is the top of a ticket that includes multiple candidates.	If the field is invalid or not present, then the implementation is required to ignore it.
PartyId	`xs:IDREF`	Optional	Single	Reference to a *Party* element with additional information about the candidate's affiliated party. This is the party affiliation that is intended to be presented as part of ballot information.	If the field is invalid or not present, then the implementation is required to ignore it.
PersonId	`xs:IDREF`	Optional	Single	Reference to a *Person* element with additional information about the candidate.	If the field is invalid or not present, then the implementation is required to ignore it.
PostElectionStatus	*CandidatePostElectionStatus*	Optional	Single	Final status of the candidate (e.g. winner, withdrawn, etc. . .).	If the field is invalid or not present, then the implementation is required to ignore it.
PreElectionStatus	*CandidatePreElectionStatus*	Optional	Single	Registration status of the candidate (e.g. filed, qualified, etc. . .).	If the field is invalid or not present, then the implementation is required to ignore it.

```
<Candidate id="can10961">
   <BallotName>
     <Text language="en">Ken T. Cuccinelli II</Text>
   </BallotName>
   <PartyId>par0001</PartyId>
   <PersonId>per10961</PersonId>
</Candidate>
```

CandidateContest

CandidateContest extends *ContestBase* and represents a contest among candidates.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
NumberElected	xs:integer	Optional	Single	Number of candidates that are elected in the contest (i.e. "N" of N-of-M).	If the field is invalid or not present, then the implementation is required to ignore it.
OfficeIds	xs:IDREFS	Optional	Single	References a set of *Office* elements, if available, which give additional information about the offices. **Note:** the order of the office IDs **must** be in the same order as the candidates listed in *BallotSelectionIds*. E.g., if the various *BallotSelectionIds* reference *CandidateSelection* elements which reference the candidate for President first and Vice-President second, the *OfficeIds* should reference the office of President first and the office of Vice-President second.	If the field is invalid or not present, then the implementation is required to ignore it.
PrimaryPartyIds	xs:IDREFS	Optional	Single	References *Party* elements, if the contest is related to a particular party.	If the field is invalid or not present, then the implementation is required to ignore it.
VotesAllowed	xs:integer	Optional	Single	Maximum number of votes/write-ins per voter in this contest.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<CandidateContest id="cc20003">
   <BallotSelectionIds>cs10961 cs10962 cs10963</BallotSelectionIds>
   <BallotTitle>
     <Text language="en">Governor of Virginia</Text>
   </BallotTitle>
   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
   <Name>Governor</Name>
   <NumberElected>1</NumberElected>
   <OfficeId>off0000</OfficeId>
   <VotesAllowed>1</VotesAllowed>
</CandidateContest>
```

CandidateSelection

CandidateSelection extends *BallotSelectionBase* and represents a ballot selection for a candidate contest.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
CandidateIds	xs:IDREFS	Optional	Single	References a set of *Candidate* elements. The number of candidates that can be references is unbounded in cases where the ballot selection is for a ticket (e.g. "President/Vice President", "Governor/Lt Governor").	If the field is invalid or not present, then the implementation is required to ignore it.
EndorsementPartyIds	xs:IDREFS	Optional	Single	References a set of *Party* elements, which signifies one or more endorsing parties for the candidate(s).	If the field is invalid or not present, then the implementation is required to ignore it.
IsWriteIn	xs:boolean	Optional	Single	Signifies if the particular ballot selection allows for write-in candidates. If true, one or more write-in candidates are allowed for this contest.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<CandidateSelection id="cs10861">
   <CandidateIds>can10861a can10861b</CandidateIds>
   <EndorsementPartyIds>par0001</EndorsementPartyIds>
</CandidateSelection>
```

ContactInformation

For defining contact information about objects such as persons, boards of authorities, organizations, etc. ContactInformation is always a sub-element of another object (e.g. *ElectionAdministration*, *Office*, *Person*, *Source*). ContactInformation has an optional attribute `label`, which allows the feed to refer back to the original label for the information (e.g. if the contact information came from a CSV, `label` may refer to a row ID).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
AddressLine	`xs:string`	Optional	Repeats	The "location" portion of a mailing address. *See usage note.*	If the field is invalid or not present, then the implementation is required to ignore it.
Directions	*InternationalizedText*	Optional	Single	Specifies further instructions for locating this entity.	If the element is invalid or not present, then the implementation is required to ignore it.
Email	`xs:string`	Optional	Repeats	An email address for the contact.	If the field is invalid or not present, then the implementation is required to ignore it.
Fax	`xs:string`	Optional	Repeats	A fax line for the contact.	If the field is invalid or not present, then the implementation is required to ignore it.
Hours **[deprecated]**	*InternationalizedText*	Optional	Single	Contains the hours (in local time) that the location is open *(NB: this element is deprecated in favor of the more structured :ref:'multi-xml-hours-open' element. It is strongly encouraged that data providers move toward contributing hours in this format).*	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	`xs:IDREF`	Optional	Single	References an *HoursOpen* element, which lists the hours of operation for a location.	If the field is invalid or not present, then the implementation is required to ignore it.
LatLng	*LatLng*	Optional	Single	Specifies the latitude and longitude of this entity.	If the element is invalid or not present, then the implementation is required to ignore it.
Name	`xs:string`	Optional	Single	The name of the location or contact. *See usage note.*	If the field is invalid or not present, then the implementation is required to ignore it.
Phone	`xs:string`	Optional	Repeats	A phone number for the contact.	If the field is invalid or not present, then the implementation is required to ignore it.
Uri	`xs:anyURI`	Optional	Repeats	An informational URI for the contact or location.	If the field is invalid or not present, then the implementation is required to ignore it.

`Name` and `AddressLine` Usage Note

The `Name` and `AddressLine` fields should be chosen so that a display or mailing address can be constructed programmatically by joining the `Name` and `AddressLine` fields together. For example, for the following address:

```
Department of Elections
1 Dr. Carlton B. Goodlett Place, Room 48
San Francisco, CA 94102
```

The name could be "Department of Elections" and the first address line could be "1 Dr. Carlton B. Goodlett Place, Room 48."

However, VIP does not yet support the representation of mailing addresses whose "name" portion spans more than one line, for example:

```
California Secretary of State
Elections Division
1500 11th Street
Sacramento, CA 95814
```

For addresses like the above, we recommend choosing a name like, "California Secretary of State, Elections Division" with "1500 11th Street" as the first address line. This would result in a programmatically constructed address like the following:

```
California Secretary of State, Elections Division
1500 11th Street
Sacramento, CA 95814
```

```
<ContactInformation label="ci10861a">
   <AddressLine>1600 Pennsylvania Ave</AddressLine>
   <AddressLine>Washington, DC 20006</AddressLine>
   <Email>president@whitehouse.gov</Email>
   <Phone>202-456-1111</Phone>
   <Phone annotation="TDD">202-456-6213</Phone>
   <Uri>http://www.whitehouse.gov</Uri>
</ContactInformation>
```

ContestBase

A base model for all Contest types: *BallotMeasureContest*, *CandidateContest*, *PartyContest*, and *RetentionContest* (NB: the latter because it extends *BallotMeasureContest*).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Abbreviation	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
BallotSelectionIds	xs:IDREFS	Optional	Single	References a set of BallotSelections, which could be of any selection type that extends *BallotSelectionBase*.	If the field is invalid or not present, then the implementation should ignore it.
BallotSubTitle	*InternationalizedText*	Optional	Single	Subtitle of the contest as it appears on the ballot.	If the element is invalid or not present, then the implementation should ignore it.
BallotTitle	*InternationalizedText*	Optional	Single	Title of the contest as it appears on the ballot.	If the element is invalid or not present, then the implementation should ignore it.
ElectoralDistrictId	xs:IDREF	**Required**	Single	References an *ElectoralDistrict* element that represents the geographical scope of the contest.	If the field is invalid, then the implementation should ignore it.
ElectorateSpecification	*InternationalizedText*	Optional	Single	Specifies any changes to the eligible electorate for this contest past the usual, "all registered voters" electorate. This subtag will most often be used for primaries and local elections. In primaries, voters may have to be registered as a specific party to vote, or there may be special rules for which ballot a voter can pull. In some local elections, non-citizens can vote.	If the element is invalid or not present, then the implementation should ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifiers for a contest that links to another source of information.	If the element is invalid or not present, then the implementation should ignore it.
HasRotation	xs:boolean	Optional	Single	Indicates whether the selections in the contest are rotated.	If the field is invalid or not present, then the implementation should ignore it.
Name	xs:string	Optional	Single	Name of the contest, not necessarily how it appears on the ballot (NB: BallotTitle should be used for this purpose).	If the field is invalid or not present, then the implementation should ignore it.

Election

The Election object represents an Election Day, which usually consists of many individual contests and/or referenda. A feed must contain **exactly one** Election object. All relationships in the feed (e.g., street segment to precinct to polling location) are assumed to relate only to the Election specified by this object. It is permissible, and recommended, to combine unrelated contests (e.g., a special election and a general election) that occur on the same day into one feed with one Election object.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Date	`xs:date`	**Required**	Single	Specifies when the election is being held. The *Date* is considered to be in the timezone local to the state holding the election.	If the field is invalid, then the implementation is required to ignore the `Election` element containing it.
ElectionType	*InternationalizedText*	Optional	Single	Specifies the highest controlling authority for election (e.g., federal, state, county, city, town, etc.)	If the element is invalid or not present, then the implementation is required to ignore it.
StateId	`xs:IDREF`	**Required**	Single	Specifies a link to the *State* element where the election is being held.	If the field is invalid, then the implementation is required to ignore the `Election` element containing it.
IsStatewide	`xs:boolean`	Optional	Single	Indicates whether the election is statewide.	If the field is not present or invalid, the implementation is required to default to "yes".
Name	*InternationalizedText*	Optional	Single	The name for the election (**NB:** while optional, this element is highly recommended).	If the element is invalid or not present, then the implementation is required to ignore it.
RegistrationInfo	*InternationalizedText*	Optional	Single	Specifies information about registration for this election either as text or a URI.	If the element is invalid or not present, then the implementation is required to ignore it.
AbsenteeBallotInfo	*InternationalizedText*	Optional	Single	Specifies information about requesting absentee ballots either as text or a URI	If the element is invalid or not present, then the implementation is required to ignore it.
ResultsUri	`xs:anyURI`	Optional	Single	Contains a URI where results for the election may be found	If the field is invalid or not present, then the implementation is required to ignore it.
PollingHours	*Ho*	Op-	Sin-	Contains the hours (in local time) that Election Day polling loca-	If the element

```
<Election id="ele30000">
  <AbsenteeRequestDeadline>2013-10-30</AbsenteeRequestDeadline>
  <Date>2013-11-05</Date>
  <ElectionType>
    <Text language="en">General</Text>
    <Text language="es">Generales</Text>
  </ElectionType>
  <HasElectionDayRegistration>false</HasElectionDayRegistration>
  <HoursOpenId>hours0001</HoursOpenId>
  <IsStatewide>true</IsStatewide>
  <Name>
    <Text language="en">2013 Statewide General</Text>
  </Name>
  <RegistrationDeadline>2013-10-15</RegistrationDeadline>
  <ResultsUri>http://www.sbe.virginia.gov/ElectionResults.html</ResultsUri>
  <StateId>st51</StateId>
</Election>
```

ElectionAdministration

The Election Administration represents an institution for serving a locality's (or state's) election functions.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
AbsenteeUri	xs:anyURI	Optional	Single	Specifies the web address for information on absentee voting.	If the field is invalid or not present, then the implementation is required to ignore it.
AmIRegisteredUri	xs:anyURI	Optional	Single	Specifies the web address for information on whether an individual is registered.	If the field is invalid or not present, then the implementation is required to ignore it.
Department	*Department*	**Required**	Repeats	Describes the administrative body for a particular voter service.	There must be at least one valid *Department* in each *ElectionAdministration* element. If no valid *Department* objects are present, the implementation is required to ignore the *ElectionAdministration* object that contains it/them.
ElectionsUri	xs:anyURI	Optional	Single	Specifies web address the administration's website.	If the field is invalid or not present, then the implementation is required to ignore it.
RegistrationUri	xs:anyURI	Optional	Single	Specifies web address for information on registering to vote.	If the field is invalid or not present, then the implementation is required to ignore it.
RulesUri	xs:anyURI	Optional	Single	Specifies a URI for the election rules and laws (if any) for the jurisdiction of the administration.	If the field is invalid or not present, then the implementation is required to ignore it.
WhatIsOnMyBallotUri	xs:anyURI	Optional	Single	Specifies web address for information on what is on an individual's ballot.	If the field is invalid or not present, then the implementation is required to ignore it.
WhereDoIVoteUri	xs:anyURI	Optional	Single	The Specifies web address for information on where an individual votes based on their address.	If the field is invalid or not present, then the implementation is required to ignore it.

Department

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ContactInformation	*ContactInformation*	Optional	Single	Contact and physical address information for the election administration body (see *ContactInformation*).	If the element is invalid or not present, then the implementation is required to ignore it.
ElectionOfficialPersonId	xs:IDREF	Optional	Single	The individual to contact at the election administration office. The specified person should be the *election official*.	If the field is invalid or not present, then the implementation is required to ignore it.
VoterService	*VoterService*	Optional	Repeats	The types of services and appropriate contact individual available to voters.	If the element is invalid or not present, then the implementation is required to ignore it.

VoterService

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ContactInformation	*ContactInformation*	Optional	Single	The contact for a particular voter service.	If the element is invalid or not present, then the implementation is required to ignore it.
Description	*InternationalizedText*	Optional	Single	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
ElectionOfficialPersonId	`xs:IDREF`	Optional	Single	The *authority* for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
Type	*VoterServiceType*	Optional	Single	The type of *voter service*.	If the field is invalid or not present, then the implementation is required to ignore it.
OtherType	`xs:string`	Optional	Single	If Type is "other", OtherType allows for cataloging another type of voter service.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<ElectionAdministration id="ea40133">
   <AbsenteeUri>http://www.sbe.virginia.gov/absenteevoting.html</AbsenteeUri>
   <AmIRegisteredUri>https://www.vote.virginia.gov/</AmIRegisteredUri>
   <Department>
     <ContactInformation label="ci60000">
       <AddressLine>Washington Building First Floor</AddressLine>
       <AddressLine>1100 Bank Street</AddressLine>
       <AddressLine>Richmond, VA 23219</AddressLine>
       <Name>State Board of Elections</Name>
     </ContactInformation>
   </Department>
   <ElectionsUri>http://www.sbe.virginia.gov/</ElectionsUri>
   <RegistrationUri>https://www.vote.virginia.gov/</RegistrationUri>
   <RulesUri>http://www.sbe.virginia.gov/</RulesUri>
   <WhatIsOnMyBallotUri>https://www.vote.virginia.gov/</WhatIsOnMyBallotUri>
   <WhereDoIVoteUri>https://www.vote.virginia.gov/</WhereDoIVoteUri>
</ElectionAdministration>
```

ElectoralDistrict

The `ElectoralDistrict` object represents the geographic area in which contests are held. Examples of `ElectoralDistrict` include: "the state of Maryland", "Virginia's 5th Congressional District", or "Union School District". The geographic area that comprises a `ElectoralDistrict` is defined by which precincts link to the `ElectoralDistrict`.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifiers that link to external datasets (e.g. OCD-IDs)	If the element is invalid or not present, then the implementation is required to ignore it.
Name	`xs:string`	**Required**	Single	Specifies the electoral area's name.	If the field is invalid or not present, then the implementation is required to ignore the `ElectoralDistrict` object containing it.
Number	`xs:integer`	Optional	Single	Specifies the district number of the district (e.g. 34, in the case of the 34th State Senate District). If a number is not applicable, instead of leaving the field blank, leave this field out of the object; empty strings are not valid for xs:integer fields.	If the field is invalid or not present, then the implementation is required to ignore it.
Type	*DistrictType*	**Required**	Single	Specifies the type of electoral area.	If the field is invalid or not present, then the implementation is required to ignore the `ElectoralDistrict` object containing it.
OtherType	`xs:string`	Optional	Single	Allows for cataloging a new *DistrictType* option when `Type` is specified as "other".	If the field is invalid or not present, then the implementation is required to ignore it.

```
<ElectoralDistrict id="ed60129">
   <ExternalIdentifiers>
     <ExternalIdentifier>
       <Type>ocd-id</Type>
       <Value>ocd-division/country:us/state:va</Value>
     </ExternalIdentifier>
     <ExternalIdentifier>
       <Type>fips</Type>
       <Value>51</Value>
     </ExternalIdentifier>
   </ExternalIdentifiers>
   <Name>Commonwealth of Virginia</Name>
   <Type>state</Type>
</ElectoralDistrict>
```

ExternalIdentifiers

The `ExternalIdentifiers` element allows VIP data to connect with external datasets (e.g. candidates with campaign finance datasets, electoral geographies with OCD-IDs that allow for greater connectivity with additional datasets, etc...). Examples for `ExternalIdentifiers` can be found on the objects that support them:

- *Candidate*
- Any element that extends *ContestBase*
- *ElectoralDistrict*

- *Locality*
- *Office*
- *Party*
- *Precinct*
- *State*

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ExternalIdentifier	*ExternalIdentifier*	**Required**	Repeats	Defines the identifier and the type of identifier it is (see *ExternalIdentifier* for complete information).	At least one valid *ExternalIdentifier* must be present for `ExternalIdentifiers` to be valid. If no valid *ExternalIdentifier* is present, the implementation is required to ignore the `ExternalIdentifiers` element.

ExternalIdentifier

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Type	*IdentifierType*	**Required**	Single	Specifies the type of identifier. Must be one of the valid types as defined by *IdentifierType*.	If the field is invalid or not present, the implementation is required to ignore the `ElectionIdentifier` containing it.
OtherType	`xs:string`	Optional	Single	Allows for cataloging an `ExternalIdentifier` type that falls outside the options listed in *IdentifierType*. `Type` should be set to "other" when using this field.	If the field is invalid or not present, then the implementation is required to ignore it.
Value	`xs:string`	**Required**	Single	Specifies the identifier.	If the field is invalid or not present, the implementation is required to ignore the `ElectionIdentifier` containing it.

```
<ExternalIdentifiers>
   <ExternalIdentifier>
      <Type>ocd-id</Type>
      <Value>ocd-division/country:us/state:nc/county:durham</Value>
   </ExternalIdentifier>
   <ExternalIdentifier>
      <Type>FIPS</Type>
      <Value>37063</Value>
   </ExternalIdentifier>
   <ExternalIdentifier>
      <Type>OTHER</Type>
      <OtherType>GNIS</OtherType>
      <Value>1008550</Value>
   </ExternalIdentifier>
   <external_identifer>
```

```
      <Type>OTHER</Type>
      <OtherType>census</OtherType>
      <Value>99063</Value>
   </ExternalIdentifier>
</ExternalIdentifiers>
```

HoursOpen

A structured way of describing the days and hours that a place such as a *Office* or *PollingLocation* is open, or that an event such as an *Election* is happening. The range of days indicated by the *StartDate* and *EndDate* in each *Schedule* element should not overlap with peer *Schedule* elements. For example, it is invalid to specify a schedule from 10/01/2016 to 10/31/2016 and also specify a schedule from 10/10/2016 to 10/11/2016 within the same *HoursOpen* element.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Schedule	*Schedule*	**Required**	Repeats	Defines a block of days and hours that a place will be open.	At least one valid *Schedule* must be present for `HoursOpen` to be valid. If no valid *Schedule* is present, the implementation is required to ignore the `HoursOpen` element.

Schedule

A sub-portion of the schedule. This describes a range of days, along with one or more set of open and close times for those days, as well as the options describing whether or not appointments are necessary or possible.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Hours	*Hours*	Optional	Repeats	Blocks of hours in the date range in which the place is open.	If the element is invalid or not present, then the implementation is required to ignore it.
IsOnlyByAppointment	`xs:boolean`	Optional	Single	If true, the place is only open during the specified time window with an appointment.	If the field is invalid or not present, then the implementation is required to ignore it.
IsOrByAppointment	`xs:boolean`	Optional	Single	If true, the place is open during the hours specified time window and may also be open with an appointment.	If the field is invalid or not present, then the implementation is required to ignore it.
IsSubjectToChange	`xs:boolean`	Optional	Single	If true, the place should be open during the specified time window, but may be subject to change. People should contact prior to arrival to confirm hours are still accurate.	If the field is invalid or not present, then the implementation is required to ignore it.
StartDate	`xs:date`	Optional	Single	The date at which this collection of start and end times and options begin.	If the field is invalid or not present, then the implementation is required to ignore it.
EndDate	`xs:date`	Optional	Single	The date at which this collection of start and end times and options end.	If the field is invalid or not present, then the implementation is required to ignore it.

Hours

The open and close time for this place. All times must be fully specified, including a timezone offset from UTC.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
StartTime	*TimeWithZone*	Optional	Single	The time at which this place opens.	If the element is invalid or not present, then the implementation is required to ignore it.
EndTime	*TimeWithZone*	Optional	Single	The time at which this place closes.	If the element is invalid or not present, then the implementation is required to ignore it.

TimeWithZone

A string pattern restricting the value to a time with an included offset from UTC. The pattern is

```
(([01][0-9]|2[0-3]):[0-5][0-9]:[0-5][0-9]|(24:00:00))(Z|[+-]((0[0-9]|1[0-3]):[0-5][0-9]|14
```

```
<HoursOpen id="hours0001">
  <Schedule>
    <Hours>
      <StartTime>06:00:00-05:00</StartTime>
      <EndTime>12:00:00-05:00</EndTime>
```

```
    </Hours>
    <Hours>
      <StartTime>13:00:00-05:00</StartTime>
      <EndTime>19:00:00-05:00</EndTime>
    </Hours>
    <StartDate>2013-11-05</StartDate>
    <EndDate>2013-11-05</EndDate>
  </Schedule>
</HoursOpen>
```

InternationalizedText

`InternationalizedText` allows for support of multiple languages for a string. `InternationalizedText` has an optional attribute `label`, which allows the feed to refer back to the original label for the information (e.g. if the contact information came from a CSV, `label` may refer to a row ID). Examples of `InternationalizedText` can be seen in:

- Any element that extends *ContestBase*
- Any element that extends *BallotSelectionBase*
- *Candidate*
- *ContactInformation*
- *Election*
- *ElectionAdministration*
- *Office*
- *Party*
- *Person*
- *PollingLocation*
- *Source*

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Text	*LanguageString*	**Required**	Repeats	Contains the translations of a particular string of text.	At least one valid `Text` must be present for `InternationalizedText` to be valid. If no valid `Text` is present, the implementation is required to ignore the `InternationalizedText` element.

LanguageString

`LanguageString` extends xs:string and can contain text from any language. `LanguageString` has one required attribute, `language`, that must contain the 2-character language code for the type of language `LanguageString` contains.

```
<BallotTitle>
   <Text language="en">Retention of Supreme Court Justice</Text>
   <Text language="es">La retención de juez de la Corte Suprema</Text>
</BallotTitle>
```

LatLng

The latitude and longitude of a polling location in WGS 84 format. Both latitude and longitude values are measured in decimal degrees.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Latitude	`xs:float`	**Required**	Single	The latitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
Longitude	`xs:float`	**Required**	Single	The longitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
Source	`xs:string`	Optional	Single	The system used to perform the lookup from location name to lat/lng. For example, this could be the name of a geocoding service.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<PollingLocation id="pl81274">
   <AddressLine>ALBEMARLE HIGH SCHOOL</AddressLine>
   <AddressLine>2775 Hydraulic Rd</AddressLine>
   <AddressLine>Charlottesville, VA 229018917</AddressLine>
   <HoursOpenId>hours0001</HoursOpenId>
   <LatLng>
     <Latitude>38.0754627</Latitude>
     <Longitude>-78.5014875</Longitude>
     <Source>Google Maps</Source>
   </LatLng>
</PollingLocation>
```

Locality

The Locality object represents the jurisdiction below the *State* (e.g. county).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ElectionAdministrationId	xs:IDREF	Optional	Single	Links to the locality's *ElectionAdministration* object.	If the field is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Another identifier for a locality that links to another dataset (e.g. OCD-ID)	If the element is invalid or not present, then the implementation is required to ignore it.
Name	xs:string	**Required**	Single	Specifies the name of a locality.	If the field is not present or invalid, the implementation is required to ignore the Locality element containing it.
PollingLocationIds	xs:IDREFS	Optional	Single	Specifies a link to a set of the locality's :ref:'polling locations <multi-xml-polling-location>'s. If early vote centers or ballot drop locations are locality-wide, they should be specified here.	If the field is invalid or not present, the implementation is required to ignore it. However, the implementation should still check to see if there are any polling locations associated with this locality's state.
StateId	xs:IDREF	**Required**	Single	References the locality's *State*.	If the field is invalid or not present, the implementation is required to ignore the Locality element containing.
Type	*DistrictType*	Optional	Single	Defines the kind of locality (e.g. county, town, et al.), which is one of the various *DistrictType enumerations*.	If the field is invalid or not present, then the implementation is required to ignore it.
OtherType	xs:string	Optional	Single	Allows for defining a type of locality that falls outside the options listed in *DistrictType*.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<Locality id="loc70001">
  <ElectionAdministrationId>ea40001</ElectionAdministrationId>
  <ExternalIdentifiers>
    <ExternalIdentifier>
      <Type>ocd-id</Type>
      <Value>ocd-division/country:us/state:va/county:albemarle</Value>
    </ExternalIdentifier>
  </ExternalIdentifiers>
  <Name>ALBEMARLE COUNTY</Name>
  <StateId>st51</StateId>
  <Type>county</Type>
</Locality>
```

Office

`Office` represents the office associated with a contest or district (e.g. Alderman, Mayor, School Board, et al).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ContactInformation	*ContactInformation*	Optional	Repeats	Specifies the contact information for the office and/or individual holding the office.	If the element is invalid or not present, then the implementation is required to ignore it.
Description	*InternationalizedText*	Optional	Single	A brief description of the office and its purpose.	If the element is invalid or not present, then the implementation is required to ignore it.
ElectoralDistrictId	`xs:IDREF`	**Required**	Single	Links to the *ElectoralDistrict* element associated with the office.	If the field is invalid or not present, the implementation is required to ignore the `Office` element containing it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifiers that link this office to other related datasets (e.g. campaign finance systems, OCD IDs, et al.).	If the element is invalid or not present, then the implementation is required to ignore it.
FilingDeadline	`xs:date`	Optional	Single	Specifies the date and time when a candidate must have filed for the contest for the office.	If the field is invalid or not present, then the implementation is required to ignore it.
IsPartisan	`xs:boolean`	Optional	Single	Indicates whether the office is partisan.	If the field is invalid or not present, then the implementation is required to ignore it.
Name	*InternationalizedText*	**Required**	Single	The name of the office.	If the field is invalid or not present, the implementation is required to ignore the `Office` element containing it.
OfficeHolderPersonIds	`xs:IDREFS`	Optional	Single	Links to the *Person* element(s) that hold additional information about the current office holder(s).	If the field is invalid or not present, then the implementation is required to ignore it.
Term	*Term*	Optional	Single	Defines the term the office can be held.	If the element is invalid or not present, then the implementation is required to ignore it.

Term

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Type	*OfficeTermType*	Optional	Single	Specifies the type of office term (see *OfficeTermType* for valid values).	If the field is invalid or not present, the implementation is required to ignore the `Office` element containing it.
StartDate	`xs:date`	Optional	Single	Specifies the start date for the current term of the office.	If the field is invalid or not present, then the implementation is required to ignore it.
EndDate	`xs:date`	Optional	Single	Specifies the end date for the current term of the office.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<Office id="off0000">
  <ElectoralDistrictId>ed60129</ElectoralDistrictId>
  <FilingDeadline>2013-01-01</FilingDeadline>
```

```
  <IsPartisan>false</IsPartisan>
  <Name>
    <Text language="en">Governor</Text>
  </Name>
  <Term>
    <Type>full-term</Type>
  </Term>
</Office>
```

OrderedContest

`OrderedContest` encapsulates links to the information that comprises a contest and potential ballot selections. `OrderedContest` elements can be collected within a *BallotStyle* to accurate depict exactly what will show up on a particular ballot in the proper order.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ContestId	`xs:IDREF`	**Required**	Single	Links to elements that extend *ContestBase*.	If the field is invalid or not present, the implementation is required to ignore the `OrderedContest` element containing it.
OrderedBallotSelectionIds	`xs:IDREFS`	Optional	Single	Links to elements that extend *BallotSelectionBase*.	If the field is invalid or not present, the implementation is required to ignore it. If an `OrderedBallotSelectionIds` element is not present, the presumed order of the selection will be the order of *BallotSelectionBase*-extended elements referenced by the underlying *ContestBase*-extended elements.

```
<OrderedContest id="oc20003abc">
   <ContestId>cc20003</ContestId>
   <OrderedBallotSelectionIds>cs10961 cs10962 cs10963</OrderedBallotSelectionIds>
</OrderedContest>
```

Party

This element describes a political party and the metadata associated with them. These can also include "dummy" parties to indicate a type of contest (e.g., a Voter Nominated *CandidateContest* can use the **PrimaryPartyIds** field and a dummy Party object to indicate that the contest is a "Top-Two" primary).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Abbreviation	`xs:string`	Optional	Single	An abbreviation for the party name.	If the field is invalid or not present, then the implementation is required to ignore it.
Color	*HtmlColorString*	Optional	Single	The preferred display color for the party, for use in maps and other displays.	If the element is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifiers that link this party to other related data sets (e.g. a campaign finance system, etc).	If the element is invalid or not present, then the implementation is required to ignore it.
IsWriteIn	`xs:boolean`	Optional	Single	Signals if this political party is one that is officially recognized by a local, state, or federal organization, or is a "write-in" in jurisdictions which allow candidates to free-form enter their political affiliation. If this field is not present then it is assumed to be false.	If the field is invalid or not present, then the implementation is required to ignore it.
LogoUri	`xs:anyURI`	Optional	Single	Web address of a logo to use in displays.	If the field is invalid or not present, then the implementation is required to ignore it.
Name	*InternationalizedText*	Optional	Single	The name of the party.	If the element is invalid or not present, then the implementation is required to ignore it.

HtmlColorString

A restricted string pattern for a six-character hex code representing an HTML color string. The pattern is:

`[0-9a-f]{6}`

```
<Party id="par0001">
  <Abbreviation>REP</Abbreviation>
  <Color>e91d0e</Color>
  <Name>
    <Text language="en">Republican</Text>
  </Name>
</Party>
```

PartyContest

An extension of *ContestBase* which describes a contest in which the possible ballot selections are of type *PartySelection*. These could include contests in which straight-party selections are allowed, or party-list contests (although these are more common outside of the United States).

PartySelection

This element extends *BallotSelectionBase* to support contests in which the selections can be groups of one or more parties.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
PartyIds	`xs:IDREFS`	**Required**	Single	One or more *Party* IDs which collectively represent a ballot selection.	If one or more parties referenced are invalid or not present, the implementation is required to ignore the PartySelection containing it.

Person

`Person` defines information about a person. The person may be a candidate, election administrator, or elected official. These elements reference `Person`:

- *Candidate*
- *ElectionAdministration*
- *Office*

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ContactInformation	*ContactInformation*	Optional	Repeats	Specifies contact information for the person.	If the element is invalid or not present, then the implementation is required to ignore it.
DateOfBirth	`xs:date`	Optional	Single	Represents the individual's date of birth.	If the field is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Identifiers for this person.	If the element is invalid or not present, then the implementation is required to ignore it.
FirstName	`xs:string`	Optional	Single	Represents an individual's first name.	If the field is invalid or not present, then the implementation is required to ignore it.
FullName	*InternationalizedText*	Optional	Single	Specifies a person's full name (**NB:** this information is *InternationalizedText* because it sometimes appears on ballots in multiple languages).	If the element is invalid or not present, then the implementation is required to ignore it.
Gender	`xs:string`	Optional	Single	Specifies a person's gender.	If the field is invalid or not present, then the implementation is required to ignore it.
LastName	`xs:string`	Optional	Single	Represents an individual's last name.	If the field is invalid or not present, then the implementation is required to ignore it.
MiddleName	`xs:string`	Optional	Repeats	Represents any number of names between an individual's first and last names (e.g. John **Ronald Reuel** Tolkien).	If the field is invalid or not present, then the implementation is required to ignore it.
Nickname	`xs:string`	Optional	Single	Represents an individual's nickname.	If the field is invalid or not present, then the implementation is required to ignore it.
PartyId	`xs:IDREF`	Optional	Single	Refers to the associated *Party*. This information is intended to be used by feed consumers to help them disambiguate the person's identity, but not to be presented as part of any ballot information. For that see *Candidate* **PartyId**.	If the field is invalid or not present, then the implementation is required to ignore it.
Prefix	`xs:string`	Optional	Single	Specifies a prefix associated with a person (e.g. Dr.).	If the field is invalid or not present, then the implementation is required to ignore it.
Profession	*InternationalizedText*	Optional	Single	Specifies a person's profession (**NB:** this information is *InternationalizedText* because it sometimes appears on ballots in multiple languages).	If the element is invalid or not present, then the implementation is required to ignore it.

```
<Person id="per50001">
   <ContactInformation label="ci60002">
     <Email>rwashburne@albemarle.org</Email>
     <Phone>4349724173</Phone>
   </ContactInformation>
   <FirstName>RICHARD</FirstName>
   <LastName>WASHBURNE</LastName>
   <MiddleName>J.</MiddleName>
   <Nickname>JAKE</Nickname>
   <Title>
     <Text language="en">General Registrar Physical</Text>
   </Title>
</Person>
```

PollingLocation

The PollingLocation object represents a site where voters cast or drop off ballots.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
AddressLine	`xs:string`	**Required**	Repeats	Represents the various parts of an address to a polling location.	At least one valid `AddressLine` must be present for `PollingLocation` to be valid. If no valid `AddressLine` is present, the implementation is required to ignore the `PollingLocation` element containing it.
Directions	*InternationalizedText*	Optional	Single	Specifies further instructions for locating the polling location.	If the element is invalid or not present, then the implementation is required to ignore it.
Hours **[deprecated]**	*InternationalizedText*	Optional	Single	Contains the hours (in local time) that the polling location is open (**NB:** this element is deprecated in favor of the more structured *HoursOpen* element. It is strongly encouraged that data providers move toward contributing hours in this format).	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	`xs:IDREF`	Optional	Single	Links to an *HoursOpen* element, which is a schedule of dates and hours during which the polling location is available.	If the field is invalid or not present, then the implementation is required to ignore it.
IsDropBox	`xs:boolean`	Optional	Single	Indicates if this polling location is a drop box.	If the field is invalid or not present, then the implementation is required to ignore it.
IsEarlyVoting	`xs:boolean`	Optional	Single	Indicates if this polling location is an early vote site.	If the field is invalid or not present, then the implementation is required to ignore it.
LatLng	*LatLng*	Optional	Single	Specifies the latitude and longitude of this polling location.	If the element is invalid or not present, then the implementation is required to ignore it.
Name	`xs:string`	Optional	Single	Name of the polling location.	If the field is invalid or not present, then the implementation is required to ignore it.
PhotoUri	`xs:anyURI`	Optional	Single	Contains a link to an image of the polling location.	If the field is invalid or not present, then the implementation is required to ignore it.

Precinct

The Precinct object represents a precinct, which is contained within a Locality. While the id attribute does not have to be static across feeds for one election, the combination of *Source.VipId*, *Locality.Name*, *Precinct.Ward*, *Precinct.Name*, and *Precinct.Number* should remain constant across feeds for one election (NB: not all of the fields just mentioned are required – omitting those non-required fields is fine).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
BallotStyleId	xs:IDREF	Optional	Single	Links to the *BallotStyle*, which a person who lives in this precinct will vote.	If the field is invalid or not present, then the implementation is required to ignore it.
ElectoralDistrictIds	xs:IDREFS	Optional	Single	Links to the :ref:'multi-xml-electoral-district's (e.g., congressional district, state house district, school board district) to which the entire precinct/precinct split belongs. **Highly Recommended** if candidate information is to be provided.	If the field is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifier for the precinct that relates to another dataset (e.g. OCD-ID).	If the element is invalid or not present, then the implementation is required to ignore it.
IsMailOnly	xs:boolean	Optional	Single	Determines if the precinct runs mail-only elections.	If the field is missing or invalid, the implementation is required to assume *IsMailOnly* is false.
LocalityId	xs:IDREF	**Required**	Single	Links to the *Locality* that comprises the precinct.	If the field is invalid or not present, the implementation is required to ignore the precinct element containing it.
Name	xs:string	**Required**	Single	Specifies the precinct's name (or number if no name exists).	If the field is invalid or not present, the implementation is required to ignore the precinct element containing it.
Number	xs:string	Optional	Single	Specifies the precinct's number (e.g., 32 or 32A – alpha characters are legal). Should be used if the *Name* field is populated by a name and not a number.	If the field is invalid or not present, then the implementation is required to ignore it.
PollingLocationIds	xs:IDREFS	Optional	Single	Specifies a link to the precinct's *PollingLocation* object(s).	If the field is invalid or not present, then the implementation is required to ignore it.
PrecinctSplitName	xs:string	Optional	Single	If this field is empty, then this *Precinct* object represents a full precinct. If this field is present, then this *Precinct* object represents one portion of a split precinct. Each *Precinct* object that represents one portion of a split precinct **must** have the same *Name* value, but different *PrecinctSplitName* values. See the *sample_feed.xml* file for examples.	If the field is invalid or not present, then the implementation is required to ignore it.
Ward	xs:string	Optional	Single	Specifies the ward the precinct is contained within.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<Precinct id="pre90111">
   <BallotStyleId>bs00010</BallotStyleId>
   <ElectoralDistrictIds>ed60129 ed60311 ed60054</ElectoralDistrictIds>
   <IsMailOnly>false</IsMailOnly>
   <LocalityId>loc70001</LocalityId>
   <Name>203 - GEORGETOWN</Name>
   <Number>0203</Number>
   <PollingLocationIds>pl81274</PollingLocationIds>
</Precinct>
<!--
  Precinct split. Name and PollingLocationIds are the same but
  PrecinctSplitName is present, the ElectoralDistrictIds are different,
  and the BallotStyleId is different.
-->
<Precinct id="pre90348sp0000">
  <BallotStyleId>bs00002</BallotStyleId>
  <ElectoralDistrictIds>ed60129 ed60054 ed60150</ElectoralDistrictIds>
  <IsMailOnly>false</IsMailOnly>
  <LocalityId>loc70001</LocalityId>
  <Name>201 - JACK JOUETT</Name>
  <Number>0201</Number>
  <PollingLocationIds>pl00000 pl81273 pl81662</PollingLocationIds>
  <PrecinctSplitName>0000</PrecinctSplitName>
</Precinct>
<Precinct id="pre90348sp0001">
  <BallotStyleId>bs00015</BallotStyleId>
  <ElectoralDistrictIds>ed60129 ed60054 ed60267</ElectoralDistrictIds>
  <IsMailOnly>false</IsMailOnly>
  <LocalityId>loc70001</LocalityId>
  <Name>201 - JACK JOUETT</Name>
  <Number>0201</Number>
  <PollingLocationIds>pl00000 pl81273 pl81662</PollingLocationIds>
  <PrecinctSplitName>0001</PrecinctSplitName>
</Precinct>
```

RetentionContest

`RetentionContest` extends *BallotMeasureContest* and represents a contest where a candidate is retained in a position (e.g. a judge).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
CandidateId	`xs:IDREF`	**Required**	Single	Links to the *Candidate* being retained.	If the field is invalid or not present, the implementation is required to ignore the `RetentionContest` element containing it.
OfficeId	`xs:IDREF`	Optional	Single	Links to the information about the office.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<RetentionContest id="rc40001">
   <BallotSelectionIds>rc40001a rc40001b</BallotSelectionIds>
   <BallotTitle>
      <Text language="en">Retention of Supreme Court Justice</Text>
      <Text language="es">La retención de juez de la Corte Suprema</Text>
```

```
   </BallotTitle>
   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
   <Name>Judicial Retention, Supreme Court</Name>
   <CandidateId>can14444</CandidateId>
   <OfficeId>off20006</OfficeId>
</RetentionContest>
```

Source

The Source object represents the organization that is publishing the information. This object is the only required object in the feed file, and only one source object is allowed to be present.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Name	xs:string	**Required**	Single	Specifies the name of the organization that is providing the information.	If the field is invalid, then the implementation is required to ignore the Source element containing it.
VipId	xs:string	**Required**	Single	Specifies the ID of the organization. VIP uses FIPS codes for this ID.	If the field is invalid, then the implementation is required to ignore the Source element containing it.
DateTime	xs:dateTime	**Required**	Single	Specifies the date and time of the feed production. The date/time is considered to be in the timezone local to the organization.	If the field is invalid, then the implementation is required to ignore it.
Description	*InternationalizedText*	Optional	Single	Specifies both the nature of the organization providing the data and what data is in the feed.	If the element is invalid or not present, then the implementation is required to ignore it.
OrganizationUri	xs:string	Optional	Single	Specifies a URI to the home page of the organization publishing the data.	If the field is invalid or not present, then the implementation is required to ignore it.
FeedContactId	xs:IDREF	Optional	Single	Reference to the *Person* who will respond to inquiries about the information contained within the file.	If the field is invalid or not present, then the implementation is required to ignore it.
TouUri	xs:anyURI	Optional	Single	Specifies the website where the Terms of Use for the information in this file can be found.	If the field is invalid or not present, then the implementation is required to ignore it.
Version	xs:string	**Required**	Single	Specifies the version of the data	If the field is invalid, then the implementation is required to ignore it.

```
<Source id="src1">
   <DateTime>2013-10-24T14:25:28</DateTime>
   <Description>
      <Text language="en">SBE is the official source for Virginia data</Text>
   </Description>
   <Name>State Board of Elections, Commonwealth of Virginia</Name>
   <OrganizationUri>http://www.sbe.virginia.gov/</OrganizationUri>
   <VipId>51</VipId>
   <Version>5.0</Version>
</Source>
```

State

The State object includes state-wide election information. The ID attribute is recommended to be the state's FIPS code, along with the prefix "st".

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ElectionAdministrationId	xs:IDREF	Optional	Single	Links to the state's election administration object.	If the field is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	*ExternalIdentifiers*	Optional	Single	Other identifier for the state that relates to another dataset (e.g. OCD-ID).	If the element is invalid or not present, then the implementation is required to ignore it.
Name	xs:string	**Required**	Single	Specifiers the name of a state, such as Alabama.	If the field is invalid, then the implementation is required to ignore it.
PollingLocationIds	xs:IDREFS	Optional	Single	Specifies a link to the state's *polling locations*. If early vote centers or ballot drop locations are state-wide (e.g., anyone in the state can use them), they can be specified here, but you are encouraged to only use the *Precinct* element.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<State id="st51">
   <ElectionAdministrationId>ea40133</ElectionAdministrationId>
   <ExternalIdentifiers>
     <ExternalIdentifier>
       <Type>ocd-id</Type>
       <Value>ocd-division/country:us/state:va</Value>
     </ExternalIdentifier>
   </ExternalIdentifiers>
   <Name>Virginia</Name>
</State>
```

StreetSegment

A Street Segment objection represents a portion of a street and the links to the precinct that this geography (i.e., segment) is contained within. The start address house number must be less than the end address house number unless the segment consists of only one address, in which case these values are equal.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
AddressDirection	xs:string	Optional	Single	Specifies the (inter-)cardinal direction of the entire address. An example is "NE" for the address "100 E Capitol St NE."	If the field is invalid or not present, then the implementation is required to ignore it.
City	xs:string	**Required**	Single	The city specifies the city or town of the address.	If the field is invalid, then the implementation is required to ignore it.
IncludesAllAddresses	xs:boolean	Optional	Single	Specifies if the segment covers every address on this street. If this is *true*, then the values of **StartHouseNumber** and **EndHouseNumber** should be ignored. The value of **OddEvenBoth** must be *both*.	If the field is invalid or not present, then the implementation is required to ignore it.
IncludesAllStreets	xs:boolean	Optional	Single	Specifies if the segment covers every street in this city. If this is *true*, then the values of **OddEvenBoth**, **StartHouseNumber**, **EndHouseNumber**, **StreetName**, and **Zip** should be ignored.	If the field is invalid or not present, then the implementation is required to ignore it.
OddEvenBoth	*OebEnum*	Optional	Single	Specifies whether the odd side of the street (in terms of house numbers), the even side, or both are in included in the street segment.	If the field is not present or invalid, the implementation is required to ignore the StreetSegment containing it.
PrecinctId	xs:IDREF	Optional	Single	References the *Precinct* that contains the entire street segment.	If the field is not present or invalid, the implementation is required to ignore the StreetSegment element containing it.
StartHouseNumber	xs:integer	Optional	Single	The house number at which the street segment starts. This value is necessary for the street segment to make any sense. Unless **IncludesAllAddresses** or **IncludesAllStreets** are true, this value must be less than or equal to **EndHouseNumber**. If **IncludesAllAddresses** or **IncludesAllStreets** are true, this value is ignored.	Unless **IncludesAllAddresses** or **IncludesAllStreets** are true, if the field is not present or invalid, the implementation is required to ignore the StreetSegment element containing it. If the **StartHouseNumber** is greater than the **EndHouseNumber**, the implementation should ignore the element containing them.
EndHouseNumber	xs:integer	Optional	Single	The house number at which the street segment ends. This value is necessary for the street segment to make any sense. Unless **IncludesAllAddresses** or **IncludesAllStreets** are true, it must be greater than or equal to **StartHouseNumber**. If **IncludesAllAddresses** or **IncludesAllStreets** are true, this value is ignored.	Unless **IncludesAllAddresses** or **IncludesAllStreets** are true, if the field is not present or invalid, the implementation is required to ignore the StreetSegment element containing it. If the **EndHouseNumber** is less than the **StartHouseNumber**, the implementation should ignore the element containing it.
State	xs:string	**Required**	Single	Specifies the two-letter state abbreviation of the address.	If the field is invalid, then the implementation is required to ignore it.
StreetDirection	xs:string	Optional	Single	Specifies the (inter-)cardinal direction of the street address (e.g., the "E" in "100 E Capitol St NE").	If the field is invalid or not present, then the implementation is required to ignore it.
StreetName	xs:string	Optional	Single	Represents the name of the street for the address. A special wildcard, "*", denotes every street in the given city/town. It optionally may contain street direction, street suffix or address direction (e.g., both "Capitol" and "E Capitol St NE" are acceptable for the address "100 E Capitol St NE"), however this is not preferred. Preferred is street name alone (e.g.	If the field is invalid or not present, then the implementation is required to ignore it.

```
<StreetSegment id="ss999999">
   <City>Charlottesville</City>
   <IncludesAllAddresses>true</IncludesAllAddresses>
   <OddEvenBoth>both</OddEvenBoth>
   <PrecinctId>pre99999</PrecinctId>
   <State>VA</State>
   <StreetName>CHAPEL HILL</StreetName>
   <StreetSuffix>RD</StreetSuffix>
   <Zip>22901</Zip>
</StreetSegment>
<StreetSegment id="ss309904">
   <City>GREENWOOD</City>
   <OddEvenBoth>both</OddEvenBoth>
   <PrecinctId>pre92145</PrecinctId>
   <StartHouseNumber>1</StartHouseNumber>
   <EndHouseNumber>201</EndHouseNumber>
   <State>VA</State>
   <StreetName>MISTY MOUNTAIN</StreetName>
   <StreetSuffix>RD</StreetSuffix>
   <Zip>22943</Zip>
</StreetSegment>
```

Enumerations (Separate Pages)

BallotMeasureType

A list of the various types of ballot measures. States may have different legal definitions of each type; Wikipedia has more details about each type. These values are to help states with multiple types of non-candidate-based contests distinguish between each type; as such, the definitions in this table are simple guidelines. Ultimately it is up to the state or local election official to choose the value which best describes the ballot measure(s) in their jurisdiction.

Tag	Description
ballot-measure	A catch-all for generic types of non-candidate-based contests.
initiative	These are usually citizen-driven measures to be placed on the ballot. These could include both statutory changes and constitutional amendments.
referendum	These could include measures to repeal existing acts of legislation, legislative referrals, and legislatively-referred state constitutional amendments.
other	Anything that does not fall into the above categories.

CandidatePostElectionStatus

Tag	Description
advanced-to-runoff	For contests in which the top *N* candidates advance to the next round.
projected-winner	A candidate is expected to win, but official results are not yet complete.
winner	The candidate has officially won.
withdrawn	The candidate has withdrawn from the contest.

CandidatePreElectionStatus

Tag	Description
filed	The candidate has filed for office but not yet been qualified.
qualified	The candidate has qualified for the contest.
withdrawn	The candidate has withdrawn from the contest (but may still be on the ballot).
write-in	

DistrictType

Enumeration describing the set of possible jurisdiction and district types. Please use the enumeration value which most accurately reflects the type of district or jurisdiction in your state or county. For example, "town" and "township" may mean different things – or not be defined at all – in your state, so please use the definition which best matches your local meaning.

Tag	Description
borough	A borough
city	A city.
city-council	A specific seat/jurisdiction for a city, town, or village council.
congressional	A United States congressional district.
county	A county.
county-council	A county council district, either in its entirety or for a specific seat.
judicial	A judicial district.
municipality	A civil division which is not a town, city, village, or county.
national	The United States.
school	A school district.
special	A special-purpose district that exist separate from general-purpose districts.
state	A state, district, commonwealth, or U.S. territory.
state-house	The lower house of a state legislature.
state-senate	The upper house of a state legislature.
town	A town.
township	A township, which may be different than a town. See the Wikipedia article.
utility	A non-water public or municipal utility district.
village	A village district.
ward	A ward.
water	A water district.
other	Any district not described above. Use the *OtherType* field to describe it.

IdentifierType

Tag	Description
fips	Federal Information Processing Standards codes for states, counties, and cities.
local-level	An identifier generated or used by local governments or organizations.
national-level	An identifier generated or used by national organizations.
ocd-id	An Open Civic Data Division Identifier.
state-level	An identifier generated or used by state governments or organizations.
other	Any identifier which doesn't fall into any of the above categories.

OebEnum

Tag	Description
both	Both even and odd addresses within the range.
even	Only even-numbered addresses within the range.
odd	Only odd-numbered addresses within the range.

OfficeTermType

Tag	Description
full-term	This election is for an office for which the existing term has been completed.
unexpired-term	This election is for an office for which the original term is not yet complete.

VoteVariation

Note that the descriptions below describe what the enumeration names stand for in the context of the VIP spec, rather than provide general definitions of the election terms that the names correspond to. For example, even though there are majority voting methods that are not "1-of-m" (e.g. ranked choice voting), we constrain "majority" to 1-of-m. We do this to eliminate any source of ambiguity when a single enumeration value needs to be assigned to a contest.

Tag	Description
1-of-m	A method where each voter can select up to one option.
approval	Approval voting, where each voter can select as many options as desired.
borda	Borda count, where each voter can rank the options, and the rankings are assigned point values.
cumulative	Cumulative voting, where each voter can distribute their vote to up to *N* options.
majority	A 1-of-m method where the winner needs more than 50% of the vote to be elected.
n-of-m	A method where each voter can select up to *N* options.
plurality	A 1-of-m method where the option with the most votes is elected, regardless of whether the option has more than 50% of the vote.
proportional	A proportional representation method (other than STV), which is any system that elects winners in proportion to the total vote.
range	Range voting, where each voter can select a score for each option.
rcv	Ranked choice voting (RCV), where each voter can rank the options, and the ballots are counted in rounds. Also known as instant-runoff voting (IRV) and the single transferable vote (STV).
supermajority	A 1-of-m method where the winner needs more than some predetermined fraction of the vote to be elected, where the fraction is more than 50% (e.g. three-fifths or two-thirds).
other	Used when the vote variation type is not included in this enumeration.

VoterServiceType

Tag	Description
absentee-ballots	This department handles the dispatch, tracking, and return of absentee ballots.
overseas-voting	The department for overseas, military, and other outside-the-U.S. voters.
polling-places	This deparment handles the selection and management of polling places.
voter-registration	The deparment that manages voter registration.
other	Any other service not covered by the above descriptions.

CHAPTER 2

CSV Documentation

2.1 CSV Specification

2.1.1 Getting Started

The CSV files contain election information, with files containing links between each other, that is compiled into an XML feed that represents the data according to the XML specification. See the sample xml file and xsd file for more details.

Certain files are required to serve different types of information. Below is a listing of which files are required for different VIP data sets.

Required files:

- election.txt
- source.txt
- state.txt

Files to serve polling locations:

- election_administration.txt
- department.txt
- locality.txt
- polling_location.txt
- precinct.txt

- street_segment.txt

Files to serve candidate contests:

- candidate.txt
- candidate_contest.txt
- candidate_selection.txt
- office.txt

Files to serve referenda and ballot measures:

- ballot_measure_contest.txt
- ballot_measure_selection.txt

Files to serve retention contests:

- retention_contest.txt

CSV files must be comma-delimited, UTF-8 .txt files, named according to the specification. The id attribute for the state object should be the state's FIPS number. The id attributes are not required to remain constant for the same piece of semantic data across multiple productions of the feed (e.g. candidate Michael Smith, running for dogcatcher in Iowa, is not required to have the same candidate id attribute each time the state of Iowa publishes data). Please see our XML *Best Practices* document when creating IDs and labels for your elements.

For the data itself, the special characters &, <, and > need to be encoded as &, <, and >, respectively.

2.1.2 Element Files

ballot_measure_contest.txt

The ballot_measure_contest.txt file provides information about a ballot measure before the voters, including summary statements on each side.

Table 2.1: ballot_measure_contest

CSV Element	Description
id	
abbreviation	An abbreviation for the Ballot Contest
ballot_selection_ids	References the relevant selection object
ballot_sub_title	Subtitle of the Ballot item
ballot_title	Title of the Ballot item
electoral_district_id	References the relevant electoral district
electorate_specification	Specifies any changes to the eligible electorate for this contest past the usual "all registered voters" electorate.
external_identifier_type	Specifies the type of identifier. external_identifier_othertype. Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
has_rotation	Indicates whether the selections in the contest are rotated.
name	Name of the contest not necessarily how it appears on the ballot
sequence_order	Order in which the contests are listed on the ballot.
vote_variation	Vote variation associated with the contest
other_vote_variation	
con_statement	Specifies a statement in opposition to the referendum.
effect_of_abstain	Specifies what effect abstaining will have
full_text	Specifies the full text of the referendum as it appears on the ballot.
info_uri	Specifies a URI that links to additional information about the referendum.
passage_threshold	Specifies the threshold of votes that the referendum needs in order to pass.
pro_statement	Specifies a statement in favor of the referendum.
summary_text	Specifies a short summary of the referendum that is on the ballot below the title but above the text.
type	Specifies the particular type of ballot measure.
other_type	Allows for cataloging a new

```
id,abbreviation,Contest,ballot_selection_ids,ballot_sub_title,ballot_title,elecoral_
↪district_id,electorate_specification,external_identifier_type,external_identifier_
↪othertype,external_identifier_value,has_rotation,name,sequence_order,vote_variation,
↪other_vote_variation,con_statement,effect_of_abstain,full_text,info_uri,passage_
↪threshold,pro_statement,summary_text,type,other_type
bmc0001,HB2,bs001 bs002 bs003,Raising levy for School Bond,School Bond Issue,ed001,
↪all registered voters,,54,false,School Bond,42,majority,,This is no good.,No effect,
↪A measure to do raise funds for etc etc,www.ballotmeasure.com,two-thirds,Everything␣
↪will be great.,It's a referendum about school funding,referendum,
```

ballot_measure_selection.txt

Represents the possible selection (e.g. yes/no, recall/do not recall, et al) for a ballot_measure_contest that would appear on the ballot.

Table 2.2: ballot_measure_selection

CSV Element	Description
id	
sequence_order	Order the selection should be placed
selection	Text of selection object

```
id,sequence_order,selection
bms001,1,Proposition A
bms002,2,Proposition B
```

ballot_style.txt

A container for the contests/measures on the ballot.

Table 2.3: ballot_style

CSV Element	Description
id	
image_uri	Specifies a URI that returns an image of the sample ballot.
ordered_contest_ids	Reference to an ordered contest object
party_ids	Reference to a party object.

```
id,image_uri,ordered_contest_ids,party_ids
bs00010,http://i.giphy.com/26BoCh3PgT8ai45ji.gif,oc2025,par02
bs00011,http://i.giphy.com/3oEjHYDWEICgEpAOjK.gif,oc3000,par01
```

candidate.txt

The candidate object represents a candidate in a contest. If a candidate is running in multiple contests, the same candidate object may be used.

Table 2.4: candidate

CSV Element	Description
id	
ballot_name	The candidate's name as it will be displayed on the official ballot.
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by <link to IdentifierType>
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
file_date	Date when the candidate filed for the contest.
is_incumbent	Indicates whether the candidate is the incumbent for the office associated with the contest.
is_top_ticket	Indicates whether the candidate is the top of a ticket that includes multiple candidates.
party_id	Reference to a party element with additional information about the candidate's affiliated party.
person_id	Reference to a person element with additional information about the candidate.
post_election_status	Final status of the candidate
pre_election_status	Registration status of the candidate

```
id,ballot_name,external_identifier_type,external_identifier_othertype,external_
↪identifier_value,file_date,is_incumbent,is_top_ticket,party_id,person_id,post_
↪election_status,pre_election_status
can001,Jude Fawley,,,,2016-12-01,true,false,par01,per50001,,filed
can002,Arabella Donn,,,,2016-12-01,false,false,par02,per50002,,qualified
can003,John Coltrane,,,,2016-09-23,false,false,par02,per50003,,qualified
can004,Miles Davis,,,,2016-05-26,false,false,par01,per50004,,qualified
```

candidate_contest.txt

Represents a contest among candidates.

Table 2.5: candidate_contest

CSV Element	Description
id	
abbreviation	An abbreviation for the contest.
ballot_selection_ids	References a particular candidate_selection which could be of any selection type that extends
ballot_sub_title	Subtitle of the contest as it appears on the ballot.
ballot_title	Title of the contest as it appears on the ballot.
electoral_district_id	References an electoral_district element that represents the geographical scope of the contest.
electorate_specification	Specifies any changes to the eligible electorate for this contest past the usual "all registered voters" electorate.
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by IdentifierType.
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
has_rotation	Indicates whether the selections in the contest are rotated.
name	Name of the contest (not necessarily how it appears on the ballot)
sequence_order	Order in which the candidates are listed on the ballot.
vote_variation	Vote variation associated with the contest.
other_vote_variation	
number_elected	Number of candidates that are elected in the contest (i.e. "N" of N-of-M).
office_ids	References an Office element which gives additional information about the office.
primary_party_ids	References a party element if the contest is related to a particular party.
votes_allowed	Maximum number of votes/write-ins per voter in this contest.

```
1 id,abbreviation,ballot_selection_ids,ballot_sub_title,ballot_title,electoral_district_
  ↪id,electorate_specification,external_identifier_type,external_identifier_othertype,
  ↪external_identifier_value,has_rotation,name,sequence_order,vote_variation,other_
  ↪vote_variation,number_elected,office_ids,primary_party_ids,votes_allowed
2 cancon001,SE-1,"bs001 bs002",,Governor of Virginia,ed001,all registered voters,fips,,
  ↪49,true,Governor,1,,,1,"off001","par01",1
3 cancon002,SE-2,"bs003 bs004",,Lieutenant Governor of Virginia,ed001,all registered␣
  ↪voters,fips,,49,true,Lt Governor,2,,,1,"off002","par01",1
```

candidate_selection.txt

Represents a ballot selection for a candidate contest.

Table 2.6: candidate_selection

CSV Element	Description
id	
abbreviation	An abbreviation for the contest.
ballot_selection_ids	References a particular candidate_selection which could be of any selection type that extends
ballot_sub_title	Subtitle of the contest as it appears on the ballot.
ballot_title	Title of the contest as it appears on the ballot.
electoral_district_id	References an electoral_district element that represents the geographical scope of the contest.
electorate_specification	Specifies any changes to the eligible electorate for this contest past the usual "all registered voters" electorate.
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by IdentifierType.
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
has_rotation	Indicates whether the selections in the contest are rotated.
name	Name of the contest (not necessarily how it appears on the ballot)
sequence_order	Order in which the candidates are listed on the ballot.
vote_variation	Vote variation associated with the contest.
other_vote_variation	
number_elected	Number of candidates that are elected in the contest (i.e. "N" of N-of-M).
office_ids	References an Office element which gives additional information about the office.
primary_party_ids	References a party element if the contest is related to a particular party.
votes_allowed	Maximum number of votes/write-ins per voter in this contest.

```
id,abbreviation,ballot_selection_ids,ballot_sub_title,ballot_title,electoral_district_
↪id,electorate_specification,external_identifier_type,external_identifier_othertype,
↪external_identifier_value,has_rotation,name,sequence_order,vote_variation,other_
↪vote_variation,number_elected,office_ids,primary_party_ids,votes_allowed
cancon001,SE-1,"bs001 bs002",,Governor of Virginia,ed001,all registered voters,fips,,
↪49,true,Governor,1,,,1,"off001","par01",1
cancon002,SE-2,"bs003 bs004",,Lieutenant Governor of Virginia,ed001,all registered␣
↪voters,fips,,49,true,Lt Governor,2,,,1,"off002","par01",1
```

contact_information.txt

For defining contact information about objects such as persons, boards of authorities, organizations, etc. ContactInformation is always a sub-element of another object (e.g. election_administration, office, person, source).

Table 2.7: contact_information

CSV Element	Description
id	
address_line_1	Street name and number
address_line_2	Additional address identifier
address_line_3	City and State Zip
email	An email address for the contact.
fax	A fax line for the contact.
hours	Contains the hours the location is open. This is a free form text field.
hours_open_id	References a schedule record.
latitude	Latitude coordinate
longitude	Longitude coordinate
latlng_source	Geocoding source
name	The name of the location or contact.
phone	A phone number for the contact.
uri	An informational URI for the contact or location.
parent_id	References election_administration record.

```
id,address_line_1,address_line_2,address_line_3,directions,email,fax,hours,hours_open_
↪id,latitude,longitude,latlng_source,name,phone,uri,parent_id
ci0827,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,,
↪,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,off001
ci0828,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,,
↪,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,vs01
```

department.txt

Describes the administrative body for a particular voter service.

Table 2.8: department

CSV Element	Description
id	
election_official_person_id	The individual to contact at the election administration office. The specified person should be the election official.
election_administration_id	Should reference an election_administration record

```
id,election_official_person_id,election_administration_id
dep01,per50002,ea123
dep02,per50002,ea345
dep03,per50002,ea625
dep04,per50002,ea625
```

election.txt

The Election object represents an Election Day, which usually consists of many individual contests and/or referenda. A feed must contain exactly one Election object.

Table 2.9: election

CSV Element	Description
id	
date	Specifies when the election is being held. The Date is considered to be in the timezone local to the state holding the election.
name	The name for the election.
election_type	Specifies the highest controlling authority for election.
state_id	Specifies a link to the State element where the election is being held.
is_statewide	Indicates whether the election is statewide.
registration_info	Specifies information about registration for this election either as text or a URI.
absentee_ballot_info	Specifies information about requesting absentee ballots either as text or a URI
results_uri	Contains a URI where results for the election may be found
polling_hours	Contains the hours (in local time) that Election Day polling locations are open.
has_election_day_registration	Specifies if a voter can register on the same day of the election (i.e. the last day of the election).
registration_deadline	Specifies the last day to register for the election with the possible exception of Election Day registration.
absentee_request_deadline	Specifies the last day to request an absentee ballot.
hours_open_id	References a schedule record.

```
id,date,name,election_type,state_id,is_statewide,registration_info,absentee_ballot_info,results_uri,polling_hours,has_election_day_registration,registration_deadline,absentee_request_deadline,hours_open_id
e001,10/08/2016,Best Hot Dog,State,st51,true,www.registrationinfo.com,You can vote absentee,http://hotdogcontest.gov/results,Noon to 3p.m.,true,10/08/2016,,ho002
```

election_administration.txt

The Election Administration represents an institution for serving a locality's (or state's) election functions.

Table 2.10: election_administration

CSV Element	Description
id	
absentee_uri	Specifies the web address for information on absentee voting.
am_i_registered_uri	Specifies the web address for information on whether an individual is registered.
elections_uri	Specifies web address the administration's website.
registration_uri	Specifies web address for information on registering to vote.
rules_uri	Specifies a URI for the election rules and laws (if any) for the jurisdiction of the administration.
what_is_on_my_ballot_uri	Specifies web address for information on what is on an individual's ballot.
where_do_i_vote_uri	The Specifies web address for information on where an individual votes based on their address.

```
id,absentee_uri,am_i_registered_uri,elections_uri,registration_uri,rules_uri,what_is_on_my_ballot_uri,where_do_i_vote_uri
ea123,https://example.com/absentee,https://example.com/am-i-registered,https://example.com/elections,https://example.com/registration,https://example.com/rules,https://example.com/what-is-on-my-ballot,https://example.com/where-do-i-vote
ea345,https://example.com/absentee2,https://example.com/am-i-registered2,https://example.com/elections2,https://example.com/registration2,https://example.com/rules2,https://example.com/what-is-on-my-ballot2,https://example.com/where-do-i-vote2
```

```
ea625,https://example.com/absentee3,https://example.com/am-i-registered3,https://
↪example.com/elections3,https://example.com/registration3,https://example.com/rules3,
↪https://example.com/what-is-on-my-ballot3,https://example.com/where-do-i-vote3
```

electoral_district.txt

The ElectoralDistrict object represents the geographic area in which contests are held. Examples of ElectoralDistrict include: "the state of Maryland", "Virginia's 5th Congressional District", or "Union School District". The geographic area that comprises a ElectoralDistrict is defined by which precincts link to the ElectoralDistrict.

Table 2.11: electoral_district

CSV Element	Description
id	
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by <link to IdentifierType>
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
name	Specifies the electoral area's name.
number	Specifies the district number of the district.
type	Specifies the type of electoral area.
other_type	

```
id,external_identifier_type,external_identifier_othertype,external_identifier_value,
↪name,number,type,other_type
ed001,ocd-id,,ocd-division/country:us/state:ny/borough:brooklyn,Brooklyn,1,borough,
ed002,other,community-board,4,CB 4,2,other,community-board
ed003,county,,4,County Office 4,2,other,county
```

locality.txt

The Locality object represents the jurisdiction below the State (e.g. county).

Table 2.12: locality

CSV Element	Description
id	
election_administration_id	Links to the locality's election administration object.
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by <link to IdentifierType>
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
name	Specifies the name of a locality.
polling_location_ids	Specifies a link to the locality's polling locations.
state_id	References the locality's state.
type	Defines the kind of locality.
other_type	Allows for defining a type of locality that falls outside the options listed in DistrictType.

```
id,election_administration_id,external_identifier_type,external_identifier_othertype,
↪external_identifier_value,name,polling_location_ids,state_id,type,other_type
loc001,ea123,"ocd-id",,ocd-division/country:us/state:co/county:denver,Locality #1,
↪"poll001 poll002",st51,city,
loc002,ea345,,,,Locality #2,,st51,other,unique type
```

office.txt

Office represents the office associated with a contest or district (e.g. Alderman, Mayor, School Board, et al).

Table 2.13: office

CSV Element	Description
id	
electoral_district_id	Links to the electoral_district element associated with the office.
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by <link to IdentifierType>
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
filing_deadline	Specifies the date and time when a candidate must have filed for the contest for the office.
is_partisan	Indicates whether the office is partisan.
name	The name of the office.
office_holder_person_ids	Links to the Person element(s) that hold additional information about the current office holder(s).
term_type	Specifies the type of office term
term_start_date	Specifies the start date for the current term of the office.
term_end_date	Specifies the end date for the current term of the office.

```
id,electoral_district_id,external_identifier_type,external_identifier_othertype,
↪external_identifier_value,filing_deadline,is_partisan,name,office_holder_person_ids,
↪term_type,term_start_date,term_end_date
off001,ed001,,,,,true,Deputy Chief of Staff,per50003,full-term,2002-01-21,
off002,ed001,,,,,true,Deputy Deputy Chief of Staff,per50001,unexpired-term,2002-01-21,
off003,ed001,,,,,false,General Secretary of Secretaries,per50004,full-term,1984-0101,
```

ordered_contest.txt

OrderedContest encapsulates links to the information that comprises a contest and potential ballot selections. OrderedContest elements can be collected within a BallotStyle to accurate depict exactly what will show up on a particular ballot in the proper order.

Table 2.14: ordered_contest

CSV Element	Description
id	
contest_id	Links to elements that extend contest_base.
ordered_ballot_selection_ids	Links to elements that extend ballot selection.

```
id,contest_id,ordered_ballot_selection_ids
oc2025,con001,"bs001 bs002 bs003"
oc3000,con002,"bs001"
```

party.txt

This element describes a political party and the metadata associated with them. These can also include "dummy" parties to indicate a type of contest (e.g., a Voter Nominated CandidateContest can use the PrimaryPartyIds field and a dummy Party object to indicate that the contest is a "Top-Two" primary).

Table 2.15: party

CSV Element	Description
id	
abbreviation	An abbreviation for the party name.
color	The preferred display color for the party for use in maps and other displays.
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by IdentifierType.
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
logo_uri	Web address of a logo to use in displays.
name	The name of the party.

```
id,abbreviation,color,external_identifier_type,external_identifier_othertype,external_
↪identifier_value,logo_uri,name
par01,REP,ff0000,,,,"http://example.com/elephant.png",Republican
par02,DEM,0000ff,,,,"http://example.com/donkey.png",Democrat
par03,GRN,efefef,,,,"http://example.com/tree.png",Green
par04,WFP,ee99aa,,,,"http://example.com/worker.png",Working Families Party
```

party_contest.txt

An extension of ContestBase which describes a contest in which the possible ballot selections are of type PartySelection. These could include contests in which straight-party selections are allowed, or party-list contests (although these are more common outside of the United States).

Table 2.16: party_contest

CSV Element	Description
id	
abbreviation	An abbreviation for the contest.
ballot_selection_ids	References a particular BallotSelection.
ballot_sub_title	Subtitle of the contest as it appears on the ballot.
ballot_title	Title of the contest as it appears on the ballot.
electoral_district_id	References an ElectoralDistrict element that represents the geographical scope of the contest.
electorate_specification	Specifies any changes to the eligible electorate for this contest past the usual "all registered voters" electorate.
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by IdentifierType.
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
has_rotation	Indicates whether the selections in the contest are rotated.
name	Name of the contest not necessarily how it appears on the ballot.
sequence_order	Order in which the candidates are listed on the ballot.
vote_variation	Vote variation associated with the contest.
other_vote_variation	The name of the variation can be specified here.

```
id,abbreviation,ballot_selection_ids,ballot_sub_title,ballot_title,electoral_district_
↪id,electorate_specification,external_identifier_type,external_identifier_othertype,
↪external_identifier_value,has_rotation,name,sequence_order,vote_variation,other_
↪vote_variation
pcon001,PC1071,"bs001 bs002",,"Party Election",ed001,all registered voters,,,,false,
↪Straight Party Vote,3,,
```

party_selection.txt

This element extends BallotSelectionBase to support contests in which the selections can be groups of one or more parties.

Table 2.17: party_selection

CSV Element	Description
id	
sequence_order	Sequence of party selection
party_ids	One or more Party IDs which collectively represent a ballot selection.

```
id,sequence_order,party_ids
ps001,1,"par01 par04"
ps002,2,"par02"
ps003,3,"par03"
```

person.txt

Person defines information about a person. The person may be a candidate, election administrator, or elected official. These elements reference Person: candidate, election_administration, office.

Table 2.18: person

CSV Element	Description
id	
date_of_birth	Represents the individual's date of birth.
first_name	Represents an individual's first name.
gender	Specifies a person's gender
last_name	Represents an individual's last name.
middle_name	Represents any number of names between an individual's first and last names (e.g. John Ronald Reuel Tolkien).
nickname	Represents an individual's nickname.
party_id	Refers to the associated Party.
prefix	Specifies a prefix associated with a person (e.g. Dr.).
profession	Specifies a person's profession.
suffix	Specifies a suffix associated with a person (e.g. Jr.).
title	A title associated with a person.

```
id,date_of_birth,first_name,gender,last_name,middle_name,nickname,party_id,prefix,
↪profession,suffix,title
per50001,1961-08-04,Barack,male,Obama,Hussein,,par02,,President,II,Mr. President
per50002,1985-11-21,Carly,female,Jepsen,Rae,,par01,,Recording Artist,,
per50003,1926-09-23,John,male,Coltrane,William,Trane,par02,,Recording Artist,Saint,
per50004,1926-05-26,Miles,male,Davis,Dewey,,par01,,Recording Artist,III,
```

polling_location.txt

The PollingLocation object represents a site where voters cast or drop off ballots.

Table 2.19: polling_location

CSV Element	Description
id	
name	Name of the polling place.
address_line	Represents the various parts of an address to a polling location.
directions	Specifies further instructions for locating the polling location.
hours	Contains the hours.
photo_uri	Contains a link to an image of the polling location.
hours_open_id	Links to an schedule record.
is_drop_box	Indicates if this polling location is a drop box.
is_early_voting	Indicates if this polling location is an early vote site.
latitude	The latitude of the polling location.
longitude	The longitude of the polling location.
latlng_source	The system used to perform the lookup from location name to lat/lng.

```
id,name,address_line,directions,hours,photo_uri,hours_open_id,is_drop_box,is_early_
↪voting,latitude,longitude,latlng_source
poll001,"ALBERMARLE HIGH SCHOOL","2775 Hydraulic Rd Charlottesville, VA 22901","Use␣
↪back door",7am-8pm,www.picture.com,ho001,false,true,38.0754627,78.5014875,Google␣
↪Maps
```

```
poll002,Public Library, Main St Denver,next to the checkout counter,7am-8pm,www.
↪picture.com,,false,true,38.0754627,78.5014875,Google Maps
poll003,Historic Society,,wheelchair entrance,7am-8pm,www.picture.com,,false,true,,,
poll004,Community Center,,behind the big oak tree,7am-8pm,www.picture.com,,false,true,
↪,,
```

precinct.txt

The Precinct object represents a precinct, which is contained within a Locality. While the id attribute does not have to be static across feeds for one election, the combination of Source.VipId, Locality.Name, Precinct.Ward, Precinct.Name, and Precinct.Number should remain constant across feeds for one election (NB: not all of the fields just mentioned are required – omitting those non-required fields is fine).

Table 2.20: precinct

CSV Element	Description
id	
ballot_style_id	Links to the ballot style.
electoral_district_ids	Links to an electoral district.
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by <link to IdentifierType>
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
is_mail_only	Determines if the precinct runs mail-only elections.
locality_id	Links to the locality that comprises the precinct.
name	Specifies the precinct's name (or number if no name exists).
number	Specifies the precinct's number.
polling_location_ids	Specifies a link to the precinct's polling location object.
precinct_split_name	Refers to name of the associated precinct split.
ward	Specifies the ward the precinct is contained within.

```
id,ballot_style_id,electoral_district_ids,external_identifier_type,external_
↪identifier_othertype,external_identifier_value,is_mail_only,locality_id,name,number,
↪polling_location_ids,precinct_split_name,ward
pre90111,bs00010,"ed001",ocd-id,,"ocd-division/country:us",false,loc001,203 -␣
↪GEORGETOWN,0203,"poll001 poll002",split13,5
pre90112,bs00011,"ed002",fips,,42,false,loc001,203 - GEORGETOWN,0203,"poll003",
↪split26,6
pre90113,bs00010,"ed003",,,,false,loc002,203 - GEORGETOWN,0203,"poll004",split54,7
```

retention_contest.txt

RetentionContest extends BallotMeasureContest and represents a contest where a candidate is retained in a position (e.g. a judge).

Table 2.21: retention_contest

CSV Element	Description
id	
abbreviation	An abbreviation for a contest.
ballot_selection_ids	References a set of ballot selections.
ballot_sub_title	Subtitle of the contest as it appears on the ballot.
ballot_title	Title of the contest.
electoral_district_id	References an electoral_district element.
electorate_specification	Specifies any changes to the eligible electorate for this contest past the usual "all registered voters" electorate.
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by IdentifierType.
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
has_rotation	Indicates whether the selections in the contest are rotated.
name	Name of the contest.
sequence_order	Order in which the contests on listed on the ballot.
vote_variation	Vote variation associated with the contest.
other_vote_variation	The name of the variation.
con_statement	Specifies a statement in opposition to the referendum. It does not necessarily appear on the ballot.
effect_of_abstain	Specifies what effect abstaining (i.e. not voting) on this proposition will have.
full_text	Specifies the full text of the referendum as it appears on the ballot.
info_uri	Specifies a URI that links to additional information about the referendum.
passage_threshold	Specifies the threshold of votes that the referendum needs in order to pass. The default is a simple majority.
pro_statement	Specifies a statement in favor of the referendum. It does not necessarily appear on the ballot.
summary_text	Specifies a short summary of the referendum that is on the ballot.
type	Specifies the particular type of ballot measure.
other_type	Allows for cataloging a new BallotMeasureType option.
candidate_id	Id of the candidate up for retention.
office_id	Office of the retention.

```
1 id,abbreviation,ballot_selection_ids,ballot_sub_title,ballot_title,electoral_district_
  ↪id,electorate_specification,external_identifier_type,external_identifier_othertype,
  ↪external_identifier_value,has_rotation,name,sequence_order,vote_variation,other_
  ↪vote_variation,con_statement,effect_of_abstain,full_text,info_uri,passage_threshold,
  ↪pro_statement,summary_text,type,other_type,candidate_id,office_id
2 rc001,Judge Fred,bs001,,Let Judge Fred keep his job?,ed001,,,,,fred,1,,,,,,,,,,other,
  ↪retention,can001,off001
```

schedule.txt

A sub-portion of the schedule. This describes a range of days, along with one or more set of open and close times for those days, as well as the options describing whether or not appointments are necessary or possible.

Table 2.22: schedule

CSV Element	Description
id	
start_time	The date at which this collection of start and end times and options begin.
end_time	The date at which this collection of start and end times and options end.
is_only_by_appointment	If true the place is only open during the specified time window with an appointment.
is_or_by_appointment	If true the place is open during the hours specified time window and may also be open with an appointment.
is_subject_to_change	If true the place should be open during the specified time window but may be subject to change. People should contact prior to arrival to confirm hours are still accurate.
start_date	The date at which this collection of start and end times and options begin.
end_date	The date at which this collection of start and end times and options end.
hours_open_id	References a block of hours.

```
id,start_time,end_time,is_only_by_appointment,is_or_by_appointment,is_subject_to_
↪change,start_date,end_date,hours_open_id
sch001,07:00:00-06:00,22:00:00-06:00,,true,,2016-10-10,2016-10-12,ho001
sch002,09:00:00-06:00,20:00:00-06:00,true,,,2016-10-13,2016-10-15,ho001
sch003,08:00:00-06:00,14:00:00-06:00,,,true,2016-10-10,2016-10-15,ho002
```

source.txt

The Source object represents the organization that is publishing the information. This object is the only required object in the feed file, and only one source object is allowed to be present.

Table 2.23: source

CSV Element	Description
id	
date_time	Specifies the date and time of the feed production.
description	Specifies both the nature of the organization providing the data and what data is in the feed.
name	Specifies the name of the organization that is providing the information.
organization_uri	Specifies a URI to the home page of the organization publishing the data.
terms_of_use_uri	Specifies the website where the Terms of Use for the information in this file can be found.
vip_id	Specifies the ID of the organization. VIP uses FIPS codes for this ID.
version	Specifies the version of the data

```
id,date_time,description,name,organization_uri,terms_of_use_uri,vip_id,version
source01,2016-06-02T10:24:08,SBE is the official source for Virginia data,"State␣
↪Board of Elections, Commonwealth of Virginia",http://www.sbe.virginia.gov/,http://
↪example.com/terms,51,5.1
```

state.txt

The State object includes state-wide election information. The ID attribute is recommended to be the state's FIPS code, along with the prefix "st".

Table 2.24: state

CSV Element	Description
id	
election_administration_id	Links to the state's election administration object.
external_identifier_type	Specifies the type of identifier. Must be one of the valid types as defined by IdentifierType.
external_identifier_othertype	Allows for cataloging an ExternalIdentifier type that falls outside the options listed in IdentifierType. Type should be set to "other" when using this field.
external_identifier_value	Specifies the identifier.
name	Specifiers the name of a state.
polling_location_ids	Specifies a link to the state's polling locations.

```
id,election_administration_id,external_identifier_type,external_identifier_othertype,
↪external_identifier_value,name,polling_location_ids
st51,ea123,ocd-id,,"ocd-division/country:us/state:va",Virginia,
```

street_segment.txt

A Street Segment objection represents a portion of a street and the links to the precinct that this geography (i.e., segment) is contained within. The start address house number must be less than the end address house number unless the segment consists of only one address, in which case these values are equal.

Table 2.25: street_segment

CSV Element	Description
id	
address_direction	Specifies the (inter-)cardinal direction of the entire address. An example is "NE" for "100 E Capitol ST NE"
city	The city specifies the city or town of the address.
includes_all_addresses	Specifies if the segment covers every address on this street.
includes_all_streets	
odd_even_both	Specifies which house number types are included within the street segment.
precinct_id	References the precinct that contains the entire street segment.
start_house_number	The house number at which the street segment starts.
end_house_number	The house number at which the street segment ends.
state	Specifies the two-letter state abbreviation of the address.
street_direction	Specifies the (inter-)cardinal direction of the street address. An example is "E" in "100 E Capitol ST NE"
street_name	Represents the name of the street for the address.
street_suffix	Abbreviated non-directional suffix to the street name. An example is "St" for the address "100 E Capitol St NE".
unit_number	The apartment/unit number for a street segment.
zip	Specifies the zip code of the address. It may be 5 or 9 digits and it may include a hyphen ('-').

```
id,address_direction,city,includes_all_addresses,includes_all_streets,odd_even_both,
↪precinct_id,start_house_number,end_house_number,state,street_direction,street_name,
↪street_suffix,unit_number,zip
ss000001,N,Washington,false,false,odd,pre90113,101,199,DC,NW,Delaware,St,,20001
ss000002,S,Washington,true,false,both,pre90112,,,DC,SE,Wisconsin,Ave,,20002
```

voter_service.txt

Table 2.26: voter_service

CSV Element	Description
id	
description	Long description of the services available.
election_official_person_id	The authority for a particular voter service. References a person element.
type	The type of voter service.
other_type	OtherType allows for cataloging another type of voter service.
department_id	References an election_administration element.

```
id,description,election_official_person_id,type,other_type,department_id
vs01,A service we provide,per50002,overseas-voting,,dep01
vs00,Elections notifications,per50002,voter-registration,,dep02
vs02,Pencil sharpening,per50002,other,office-help,dep03
vs03,Guided hike to polling place,per50002,polling-places,,dep03
vs04,Bike messenger ballot delivery,per50002,absentee-ballots,,dep03
```