
Voting Information Project Specification Documentation

Release 5.1.2

Jan 19, 2022

Contents

1	Publishers of VIP data	3
2	XML Documentation	5
2.1	XML Specification	5
3	CSV Documentation	127
3.1	CSV Specification	127
4	Best Practices	237
4.1	Data Best Practices	237

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* 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

Publishers of VIP data

The following projects publish VIP data for end user consumption.

- [The Google Civic Information API](#)

2.1 XML Specification

- *Getting Started*
- *Elements & Enumerations*
 - *Single-page Format*
 - *Elements (Separate Pages)*
 - *Enumerations (Separate Pages)*

2.1.1 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 `VipObject`. 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.

2.1.2 Elements & Enumerations

Single-page Format

XML Elements & Enumerations (Single Page)

- *Elements*
 - *BallotMeasureContest*
 - * *ContestBase*
 - *BallotMeasureSelection*
 - * *BallotSelectionBase*
 - *BallotSelectionBase*
 - *BallotStyle*
 - *Candidate*
 - *CandidateContest*
 - * *ContestBase*
 - *CandidateSelection*
 - * *BallotSelectionBase*
 - *Checksum*
 - *ContactInformation*
 - * *Name and AddressLine Usage Note*
 - *ContestBase*
 - *Department*
 - * *VoterService*
 - * *ContactInformation*
 - * *Name and AddressLine Usage Note*
 - *Election*
 - *ElectionAdministration*
 - * *Department*
 - *VoterService*
 - *ContactInformation*
 - * *Name and AddressLine Usage Note*
 - * *ElectionNotice*
 - * *VoterService*
 - *ElectionNotice*
 - *ElectoralDistrict*

- *ExternalFile*
 - * *Checksum*
- *ExternalGeospatialFeature*
 - * *FeatureIdentifier*
- *ExternalIdentifier*
- *ExternalIdentifiers*
- *FeatureIdentifier*
- *Hours*
 - * *TimeWithZone*
- *HoursOpen*
 - * *Schedule*
 - *Hours*
- *TimeWithZone*
- *HtmlColorString*
- *InternationalizedText*
- *LanguageString*
- *LatLng*
- *Locality*
- *Office*
 - * *Term*
 - * *ContactInformation*
 - * *Name and AddressLine Usage Note*
- *OrderedContest*
- *Party*
 - * *HtmlColorString*
- *PartyContest*
 - * *ContestBase*
- *PartySelection*
 - * *BallotSelectionBase*
- *Person*
 - * *ContactInformation*
 - * *Name and AddressLine Usage Note*
- *PollingLocation*
 - * *LatLng*
 - * *SimpleAddressType*

- *Precinct*
 - * *SpatialBoundary*
 - *ExternalGeospatialFeature*
 - * *FeatureIdentifier*
- *RetentionContest*
- *Schedule*
 - * *Hours*
 - *TimeWithZone*
- *SimpleAddressType*
- *Source*
- *SpatialBoundary*
 - * *ExternalGeospatialFeature*
 - *FeatureIdentifier*
- *State*
- *StreetSegment*
- *Term*
- *TimeWithZone*
- *VoterService*
- *Enumerations*
 - *BallotMeasureType*
 - *CandidatePostElectionStatus*
 - *CandidatePreElectionStatus*
 - *ChecksumAlgorithm*
 - *DistrictType*
 - *GeospatialFormat*
 - *IdentifierType*
 - *OebEnum*
 - *OfficeTermType*
 - *VoteVariation*
 - *VoterServiceType*

Elements

BallotMeasureContest

The `BallotMeasureContest` provides information about a ballot measure before the voters, including summary statements on each side. Extends `single-xml-contest-base`.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Con-State-ment	<i>Internationalized-Text</i>	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	<i>Internationalized-Text</i>	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.
Full-Text	<i>Internationalized-Text</i>	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	<code>xs:anyURI</code>	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	<i>Internationalized-Text</i>	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 <i>competing initiatives</i> , information about their effect on the passage of the <i>BallotMeasureContest</i> would go here.	If the element is invalid or not present, then the implementation is required to ignore it.
ProStatement	<i>Internationalized-Text</i>	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.
Summary-Text	<i>Internationalized-Text</i>	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	<i>Ballot-Measure-Type</i>	Optional	Single	Specifies the particular type of ballot measure. Must be one of the valid <i>BallotMeasureType</i> options.	If the field is invalid or not present, then the implementation is required to ignore it.
2.1. XML Specification					9
Other-Type	<code>xs:string</code>	Optional	Single	Allows for cataloging a new <i>BallotMeasureType</i> option, when Type is specified as “other.”	If the field is invalid or not present, then the

```
1 <BallotMeasureContest id="bmc30001">
2   <BallotSelectionIds>bms30001a bms30001b</BallotSelectionIds>
3   <BallotTitle>
4     <Text language="en">State of the State</Text>
5     <Text language="es">Estado del Estado.</Text>
6   </BallotTitle>
7   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
8   <Name>Referendum on Virginia</Name>
9   <ConStatement label="bmc30001con">
10    <Text language="en">This is no good.</Text>
11    <Text language="es">Esto no es bueno.</Text>
12  </ConStatement>
13  <EffectOfAbstain label="bmc30001abs">
14    <Text language="en">Nothing will happen.</Text>
15    <Text language="es">Nada pasará.</Text>
16  </EffectOfAbstain>
17  <ProStatement label="bmc30001pro">
18    <Text language="en">Everything will be great.</Text>
19    <Text language="es">Todo va a estar bien.</Text>
20  </ProStatement>
21  <Type>referendum</Type>
22 </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
Ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Se- lec- tion- Ids	xs:IDREFS	Optional	Single	References a set of BallotSelections, which could be of any selection type that extends single-xml-ballot-selection-base.	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Sub- Ti- tle	<i>Internationalized-Text</i>	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.
Bal- lot- Ti- tle	<i>Internationalized-Text</i>	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.
Elec- toral- Dis- tric- tId	xs:IDREF	Required	Single	References an <i>ElectoralDistrict</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the ContestBase element containing it.
Elec- torate- Spec- ifi- ca- tion	<i>Internationalized-Text</i>	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.
Ex- ter- nal- den- ti- fi- ers	<i>ExternalIdentifiers</i>	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.
Has- Ro- ta- tion	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	Required	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, then the implementation is required to ignore the

2.1. XML Specification

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 *single-xml-ballot-selection-base*.

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

```

1 <BallotMeasureSelection id="bms30001a">
2   <Selection label="bms30001at">
3     <Text language="en">Yes</Text>
4     <Text language="es">Sí</Text>
5   </Selection>
6 </BallotMeasureSelection>
7 <BallotMeasureSelection id="bms30001b">
8   <Selection label="bms30001bt">
9     <Text language="en">No</Text>
10    <Text language="es">No</Text>
11  </Selection>
12 </BallotMeasureSelection>

```

BallotSelectionBase

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

Tag	Data Type	Required	Repeats?	Description	Error Handling
Sequence-Order	<i>xs:int</i>	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 <i>OrderedBallotSlectionIds</i> in <i>OrderedContest</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

BallotSelectionBase

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

Tag	Data Type	Required	Repeats?	Description	Error Handling
Sequence-Order	<i>xs:int</i>	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 <i>OrderedBallotSlectionIds</i> in <i>OrderedContest</i> .	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	Re-quired?	Re-peats?	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.
Ordered-ContestIds	xs:IDREFS	Optional	Single	Reference to a set of <i>OrderedContest</i>	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.

```

1 <BallotStyle id="bs00000">
2   <OrderedContestIds>oc20003 oc20004 oc20005 oc20025 oc20355 oc20449</
3   ↪OrderedContestIds>
4 </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
Ballot-Name	<i>InternalizedText</i>	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, then the implementation is required to ignore the Candidate element containing it.
Contact-Information	single-xml-contact-information	Optional	Single	Contact and physical address information for this Candidate and/or their campaign (see single-xml-contact-information).	If the element is invalid or not present, then the implementation is required to ignore it.
External-Identifiers	<i>ExternalIdentifiers</i>	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.
File-Date	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 <i>Party</i> 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 <i>Person</i> element with additional information about the candidate.	If the field is invalid or not present, then the implementation is required to ignore it.
Post-Election-Status	<i>CandidatePost-ElectionStatus</i>	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.
Pre-Election-Status	<i>CandidatePre-ElectionStatus</i>	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.

```

1 <Candidate id="can10961">
2   <BallotName>
3     <Text language="en">Ken T. Cuccinelli II</Text>
4   </BallotName>
5   <PartyId>par0001</PartyId>
6   <PersonId>per10961</PersonId>

```

(continues on next page)

(continued from previous page)

</Candidate>

CandidateContest

CandidateContest extends single-xml-contest-base and represents a contest among candidates.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Number-Elected	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 <i>Office</i> 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 <i>BallotSelectionIds</i> . E.g., if the various <i>BallotSelectionIds</i> reference <i>CandidateSelection</i> elements which reference the candidate for President first and Vice-President second, the <i>OfficeIds</i> 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.
Primary-PartyIds	xs:IDREFS	Optional	Single	References <i>Party</i> 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.

```

1 <CandidateContest id="cc20003">
2   <BallotSelectionIds>cs10961 cs10962 cs10963</BallotSelectionIds>
3   <BallotTitle>
4     <Text language="en">Governor of Virginia</Text>
5   </BallotTitle>
6   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
7   <Name>Governor</Name>
8   <NumberElected>1</NumberElected>
9   <OfficeId>off0000</OfficeId>

```

(continues on next page)

(continued from previous page)

```
10     <VotesAllowed>1</VotesAllowed>  
11 </CandidateContest>
```

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
Ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
Ballot- Selec- tion- Ids	xs:IDREFS	Optional	Single	References a set of <code>BallotSelections</code> , which could be of any selection type that extends <code>single-xml-ballot-selection-base</code> .	If the field is invalid or not present, then the implementation should ignore it.
Ballot- Sub- Ti- tle	<i>Internationalized-Text</i>	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.
Ballot- Ti- tle	<i>Internationalized-Text</i>	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.
Elec- toral- Dis- tric- tId	xs:IDREF	Required	Single	References an <i>ElectoralDistrict</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the <code>ContestBase</code> element containing it.
Elec- torate- Spec- ifi- ca- tion	<i>Internationalized-Text</i>	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.
Ex- ter- nal- iden- ti- fiers	<i>ExternalIdentifiers</i>	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.
Has- Ro- ta- tion	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	Required	Single	Name of the contest, not necessarily how it appears on the ballot (NB: <code>BallotTitle</code> should be used for this purpose).	If the field is invalid, then the implementation is required to ignore the

2.1. XML Specification

CandidateSelection

CandidateSelection extends single-xml-ballot-selection-base 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 <i>Candidate</i> 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 <i>Party</i> 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.

```

1 <CandidateSelection id="cs10861">
2   <CandidateIds>can10861a can10861b</CandidateIds>
3   <EndorsementPartyIds>par0001</EndorsementPartyIds>
4 </CandidateSelection>

```

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 <i>OrderedBallotSlectionIds</i> in <i>OrderedContest</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

Checksum

The *Checksum* object contains information about a cryptographic checksum, including the raw checksum value and the cryptographic hash algorithm used to compute it.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Algorithm	<i>ChecksumAlgorithm</i>	Required	Single	The cryptographic hash algorithm used to compute the checksum value.	If the field is invalid, then the implementation is required to ignore the <i>Checksum</i> element containing it.
Value	xs:string	Required	Single	The raw cryptographic checksum value encoded as a non-delimited, lowercase hexadecimal string.	If the field is invalid, then the implementation is required to ignore the <i>Checksum</i> element containing it.

```
1 <Checksum>
2   <Algorithm>sha-256</Algorithm>
3   <Value>65b634c5037f8a344616020d8060d233daa37b0f032a71d0d15ad7a5d3afa68e</Value>
4 </Checksum>
```

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	<i>Internationalized-Text</i>	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]	<i>Internationalized-Text</i>	Optional	Single	Contains the hours (in local time) that the location is open (NB: <i>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</i>).	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	xs:ID	Optional	Single	References an <i>HoursOpen</i> 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	single-xml-lat-lng	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
```

```
1 <ContactInformation label="ci10861a">
2   <AddressLine>1600 Pennsylvania Ave</AddressLine>
3   <AddressLine>Washington, DC 20006</AddressLine>
4   <Email>president@whitehouse.gov</Email>
5   <Phone>202-456-1111</Phone>
6   <Phone annotation="TDD">202-456-6213</Phone>
7   <Uri>http://www.whitehouse.gov</Uri>
8 </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
Ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Se- lec- tion- Ids	xs:IDREFS	Optional	Single	References a set of <code>BallotSelections</code> , which could be of any selection type that extends <code>single-xml-ballot-selection-base</code> .	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Sub- Ti- tle	<i>Inter-national-ized-Text</i>	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.
Bal- lot- Ti- tle	<i>Inter-national-ized-Text</i>	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.
Elec- toral- Dis- tric- tId	xs:IDREF	Required	Single	References an <i>ElectoralDistrict</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the <code>ContestBase</code> element containing it.
Elec- torate- Spec- ifi- ca- tion	<i>Inter-national-ized-Text</i>	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.
Ex- ter- nal- iden- ti- fiers	<i>External-identifiers</i>	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.
Has- Ro- ta- tion	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	Required	Single	Name of the contest, not necessarily how it appears on the ballot (NB: <code>BallotTitle</code> should be used for this purpose).	If the field is invalid, then the implementation is required to ignore the <code>ContestBase</code>

Department

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Contact-Information	single-xml-contact-information	Optional	Single	Contact and physical address information for the election administration body (see single-xml-contact-information).	If the element is invalid or not present, then the implementation is required to ignore it.
ElectionOfficialPersonId	<code>xs:IDREF</code>	Optional	Single	The individual to contact at the election administration office. The specified person should be the <i>election official</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
VoterService	single-xml-voter-service	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
Contact-Information	single-xml-contact-information	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	<i>InternationalizedText</i>	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	<code>xs:IDREF</code>	Optional	Single	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
Type	<i>VoterServiceType</i>	Optional	Single	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Other-Type	<code>xs:string</code>	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.

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	<i>Internationalized-Text</i>	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]	<i>Internationalized-Text</i>	Optional	Single	Contains the hours (in local time) that the location is open (<i>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).</i>	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	xs:ID	Optional	Single	References an <i>HoursOpen</i> 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	single-xml-lat-lng	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.
24				Chapter 2. XML Documentation	

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
```

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

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	<code>xs:date</code>	Required	Single	Specifies when the election is being held. The <i>Date</i> 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.
Election-Type	<i>Internationalized-Text</i>	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	<code>xs:IDREF</code>	Required	Single	Specifies a link to the <i>State</i> element where the election is being held.	If the field is invalid, then the implementation is required to ignore the Election element containing it.
Is-Statewide	<code>xs:boolean</code>	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	<i>Internationalized-Text</i>	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	<i>Internationalized-Text</i>	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.
Absentee-Ballot-Info	<i>Internationalized-Text</i>	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	<code>xs:anyURI</code>	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	Hours	Optional	Single	Contains the hours (in local time) that Election Day polling loca-	If the element

```
1 <Election id="ele30000">
2   <AbsenteeRequestDeadline>2013-10-30</AbsenteeRequestDeadline>
3   <Date>2013-11-05</Date>
4   <ElectionType>
5     <Text language="en">General</Text>
6     <Text language="es">Generales</Text>
7   </ElectionType>
8   <HasElectionDayRegistration>>false</HasElectionDayRegistration>
9   <HoursOpenId>hours0001</HoursOpenId>
10  <IsStatewide>true</IsStatewide>
11  <Name>
12    <Text language="en">2013 Statewide General</Text>
13  </Name>
14  <RegistrationDeadline>2013-10-15</RegistrationDeadline>
15  <ResultsUri>http://www.sbe.virginia.gov/ElectionResults.html</ResultsUri>
16  <StateId>st51</StateId>
17 </Election>
```

ElectionAdministration

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

Tag	Data Type	Re-quired?	Re-peats?	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.
BallotTrackingUri	xs:anyURI	Optional	Single	Specifies the web address for tracking information for a ballot cast by mail	If the field is invalid or not present, then the implementation is required to ignore it.
BallotProvisionalTrackingUri	xs:anyURI	Optional	Single	Specifies the web address for tracking information for a provisional ballot. To support EAC guidelines for “Processing Provisional Ballots” (https://www.eac.gov/research-and-data/provisional-voting/)	If the field is invalid or not present, then the implementation is required to ignore it.
Department	single-xml-department	Required	Re-peats	Describes the administrative body for a particular voter service.	There must be at least one valid <i>Department</i> in each <i>ElectionAdministration</i> element. If no valid <i>Department</i> objects are present, the implementation is required to ignore the <i>ElectionAdministration</i> object that contains it/them.
ElectionNotice	single-xml-election-notice	Optional	Single	A place for election administrators to post last minute and emergency notifications pertaining to the election.	If the element is invalid or not present, then the implementation is required to ignore it.
ElectionUri	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.

```

1 <ElectionAdministration id="ea40133">
2   <AbsenteeUri>http://www.sbe.virginia.gov/absenteevoting.html</AbsenteeUri>
3   <AmIRegisteredUri>https://www.vote.virginia.gov/</AmIRegisteredUri>

```

(continues on next page)

(continued from previous page)

```

4    <BallotTrackingUri>https://www.vote.virginia.gov/</BallotTrackingUri>
5    <BallotProvisionalTrackingUri>https://www.vote.virginia.gov/</
6    <BallotProvisionalTrackingUri>
7    <Department>
8      <ContactInformation label="ci60000">
9        <AddressLine>Washington Building First Floor</AddressLine>
10       <AddressLine>1100 Bank Street</AddressLine>
11       <AddressLine>Richmond, VA 23219</AddressLine>
12       <Name>State Board of Elections</Name>
13     </ContactInformation>
14   </Department>
15   <ElectionNotice>
16     <NoticeText>
17       <Text language="en">This is an emergency notification for this election.</Text>
18     </NoticeText>
19     <NoticeURI>https://www.yadayada.gov</NoticeURI>
20   </ElectionNotice>
21   <ElectionsUri>http://www.sbe.virginia.gov/</ElectionsUri>
22   <RegistrationUri>https://www.vote.virginia.gov/</RegistrationUri>
23   <RulesUri>http://www.sbe.virginia.gov/</RulesUri>
24   <WhatIsOnMyBallotUri>https://www.vote.virginia.gov/</WhatIsOnMyBallotUri>
25   <WhereDoIVoteUri>https://www.vote.virginia.gov/</WhereDoIVoteUri>
</ElectionAdministration>

```

Department

Tag	Data Type	Re- quired	Re- peats?	Description	Error Handling
Contact- Infor- mation	single- xml- contact- information	Op- tional	Sin- gle	Contact and physical address information for the election administration body (see single-xml-contact-information).	If the element is invalid or not present, then the implementation is required to ignore it.
Elec- tionOffi- cialPer- sonId	xs:IDREF	Op- tional	Sin- gle	The individual to contact at the election administration office. The specified person should be the <i>election official</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Vot- erSer- vice	single- xml-voter- service	Op- tional	Re- peats	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	Re- quired?	Re- peats?	Description	Error Handling
Contact- Informa- tion	single-xml- contact- information	Op- tional	Sin- gle	The contact for a particular voter service.	If the element is invalid or not present, then the implementation is required to ignore it.
Descrip- tion	<i>InternationalizedText</i>	Op- tional	Sin- gle	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
Elec- tionOffi- cialPer- sonId	xs:IDREF	Op- tional	Sin- gle	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
Type	<i>VoterServiceType</i>	Op- tional	Sin- gle	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Other- Type	xs:string	Op- tional	Sin- gle	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.

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	<i>Internationalized-Text</i>	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]	<i>Internationalized-Text</i>	Optional	Single	Contains the hours (in local time) that the location is open (NB: <i>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</i>).	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	xs:ID	Optional	Single	References an <i>HoursOpen</i> 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	single-xml-lat-lng	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.
2.1. XML Specification					31

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
```

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

ElectionNotice

Tag	Data Type	Required?	Repeats?	Description	Error Handling
No- tice- Text	<i>Inter- nation- alized- Text</i>	Re- quired	Single	The last minute or emergency notification text should be placed here.	If the element is invalid, then the implementation is required to ignore the ElectionNotice element containing it.
No- ticeUri	xs:anyURI	Optional	Single	Optional URL for additional information related to the last minute or emergency notification.	If the field is invalid or not present, then the implementation is required to ignore it.

VoterService

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
Contact- Informa- tion	single-xml- contact- information	Op- tional	Sin- gle	The contact for a particular voter service.	If the element is invalid or not present, then the implementation is required to ignore it.
Descrip- tion	<i>InternationalizedText</i>	Op- tional	Sin- gle	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
Elec- tionOffi- cialPer- sonId	xs:IDREF	Op- tional	Sin- gle	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
Type	<i>VoterServiceType</i>	Op- tional	Sin- gle	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Other- Type	xs:string	Op- tional	Sin- gle	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.

ElectionNotice

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
No- tice- Text	<i>InternationalizedText</i>	Re- quired	Sin- gle	The last minute or emergency notification text should be placed here.	If the element is invalid, then the implementation is required to ignore the ElectionNotice element containing it.
No- ticeUri	xs:anyURI	Op- tional	Sin- gle	Optional URL for additional information related to the last minute or emergency notification.	If the field is invalid or not present, then the implementation is required to ignore it.

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
External-identifiers	<i>External-identifiers</i>	Optional	Single	Other identifiers that link to external datasets (e.g. <i>OCD-IDs</i>)	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 <code>ElectoralDistrict</code> 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	<i>District-Type</i>	Required	Single	Specifies the type of electoral area.	If the field is invalid or not present, then the implementation is required to ignore the <code>ElectoralDistrict</code> object containing it.
Other-Type	xs:string	Optional	Single	Allows for cataloging a new <i>DistrictType</i> option when Type is specified as "other".	If the field is invalid or not present, then the implementation is required to ignore it.

```

1 <ElectoralDistrict id="ed60129">
2   <ExternalIdentifiers>
3     <ExternalIdentifier>
4       <Type>ocd-id</Type>
5       <Value>ocd-division/country:us/state:va</Value>
6     </ExternalIdentifier>
7     <ExternalIdentifier>
8       <Type>fips</Type>
9       <Value>51</Value>
10    </ExternalIdentifier>
11  </ExternalIdentifiers>
12  <Name>Commonwealth of Virginia</Name>
13  <Type>state</Type>
14 </ElectoralDistrict>

```

ExternalFile

The `ExternalFile` object holds a reference to a file external to the feed itself. External files are packaged along with the VIP feed into a single, archived file.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
FileUri	xs:anyURI	Required	Single	The URI of the external file.	If the field is invalid, then the implementation is required to ignore the ExternalFile element containing it.
Checksum	single-xml-checksum	Required	Single	The cryptographic checksum of the referenced external file.	If the element is invalid, then the implementation is required to ignore the ExternalFile element containing it.

```

1 <ExternalFile id="ef1">
2   <FileUri>precinct_shapes.zip</FileUri>
3   <Checksum>
4     <Algorithm>sha-256</Algorithm>
5     <Value>65b634c5037f8a344616020d8060d233daa37b0f032a71d0d15ad7a5d3afa68e</Value>
6   </Checksum>
7 </State>

```

Checksum

The Checksum object contains information about a cryptographic checksum, including the raw checksum value and the cryptographic hash algorithm used to compute it.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Algorithm	<i>ChecksumAlgorithm</i>	Required	Single	The cryptographic hash algorithm used to compute the checksum value.	If the field is invalid, then the implementation is required to ignore the Checksum element containing it.
Value	xs:string	Required	Single	The raw cryptographic checksum value encoded as a non-delimited, lowercase hexadecimal string.	If the field is invalid, then the implementation is required to ignore the Checksum element containing it.

```

1 <Checksum>
2   <Algorithm>sha-256</Algorithm>
3   <Value>65b634c5037f8a344616020d8060d233daa37b0f032a71d0d15ad7a5d3afa68e</Value>
4 </Checksum>

```

ExternalGeospatialFeature

The ExternalGeospatialFeature object contains a reference to a geospatial feature (one or more shapes) contained in a separate file external to the VIP feed.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
External-FileId	<code>xs:IDREF</code>	Required	Single	Links to the ExternalFile containing the geospatial shape(s) that define the feature's boundary.	If the field is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.
File-Format	Geospatial-Format	Required	Single	The format of the geospatial file.	If the field is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.
Feature-identifier	single-xml-feature-identifier	Required	Repeats	Identifiers indicating which specific shape(s) to use from the geospatial file. These refer to identifiers within the referenced external file. This is a repeated field in the XML specification, but a scalar field in the CSV specification. If more than one identifier is required with the CSV specification, multiple values can be provided by delimited by space.	If the element is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.

FeatureIdentifier

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Index	<code>xs:int</code>	Optional	Single	The index value for the shapefile feature.	If the field is invalid or not present, then the implementation is required to ignore it.

ExternalIdentifier

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Type	Identifier-Type	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 <code>ElectionIdentifier</code> containing it.
Other-Type	<code>xs:string</code>	Optional	Single	Allows for cataloging an <code>ExternalIdentifier</code> 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	<code>xs:string</code>	Required	Single	Specifies the identifier.	If the field is invalid or not present, the implementation is required to ignore the <code>ElectionIdentifier</code> containing it.


```

1 <ExternalIdentifiers>
2   <ExternalIdentifier>
3     <Type>ocd-id</Type>
4     <Value>ocd-division/country:us/state:nc/county:durham</Value>
5   </ExternalIdentifier>
6   <ExternalIdentifier>
7     <Type>FIPS</Type>
8     <Value>37063</Value>
9   </ExternalIdentifier>
10  <ExternalIdentifier>
11    <Type>OTHER</Type>
12    <OtherType>GNIS</OtherType>
13    <Value>1008550</Value>
14  </ExternalIdentifier>
15  <external_identifier>
16    <Type>OTHER</Type>
17    <OtherType>census</OtherType>
18    <Value>99063</Value>
19  </external_identifier>
20 </ExternalIdentifiers>

```

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 `single-xml-contest-base`
- *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 <i>ExternalIdentifier</i> for complete information).	At least one valid <i>ExternalIdentifier</i> must be present for <code>ExternalIdentifiers</code> to be valid. If no valid <i>ExternalIdentifier</i> is present, the implementation is required to ignore the <code>ExternalIdentifiers</code> element.

FeatureIdentifier

Tag	Data Type	Required?	Repeats?	Description	Error Handling
In-index	xs:int	Optional	Single	The index value for the shapefile feature.	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
Start-Time	single-xml-time-with-zone	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.
End-Time	single-xml-time-with-zone	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

```

1 <HoursOpen id="hours0001">
2   <Schedule>
3     <Hours>
4       <StartTime>06:00:00-05:00</StartTime>
5       <EndTime>12:00:00-05:00</EndTime>
6     </Hours>
7     <Hours>
8       <StartTime>13:00:00-05:00</StartTime>
9       <EndTime>19:00:00-05:00</EndTime>
10    </Hours>
11    <StartDate>2013-11-05</StartDate>
12    <EndDate>2013-11-05</EndDate>
13  </Schedule>
14 </HoursOpen>

```

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	single-xml-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	single-xml-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.
IsOnly-ByAppointment	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.
IsOr-ByAppointment	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.
IsSubject-ToChange	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.
Start-Date	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.
End-Date	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	Re- quired?	Re- peats?	Description	Error Handling
Start-Time	single-xml-time-with-zone	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.
End-Time	single-xml-time-with-zone	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
```

```

1 <HoursOpen id="hours0001">
2   <Schedule>
3     <Hours>
4       <StartTime>06:00:00-05:00</StartTime>
5       <EndTime>12:00:00-05:00</EndTime>
6     </Hours>
7     <Hours>
8       <StartTime>13:00:00-05:00</StartTime>
9       <EndTime>19:00:00-05:00</EndTime>
10    </Hours>
11    <StartDate>2013-11-05</StartDate>
12    <EndDate>2013-11-05</EndDate>
13  </Schedule>
14 </HoursOpen>

```

HtmlColorString

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

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

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 `single-xml-contest-base` * Any element that extends `single-xml-ballot-selection-base` * *Candidate* * `single-xml-contact-information` * *Election* * *ElectionAdministration* * *Office* * *Party* * *Person* * *PollingLocation* * *Source* NOTE: Internationalized Text is not currently supported for CSV submissions. “

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
Text	<i>LanguageString</i>	Required	Repeats	Contains the translations of a particular string of text.	At least one valid Text must be present for <code>InternationalizedText</code> to be valid. If no valid Text is present, the implementation is required to ignore the <code>InternationalizedText</code> 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.

```

1 <BallotTitle>
2   <Text language="en">Retention of Supreme Court Justice</Text>
3   <Text language="es">La retención de juez de la Corte Suprema</Text>
4 </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:double	Required	Single	The latitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
Longitude	xs:double	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.

Locality

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Election-AdministrationId	xs:IDREF	Optional	Single	Links to the locality's <i>ElectionAdministration</i> object.	If the field is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	<i>ExternalIdentifiers</i>	Optional	Single	Another identifier for a locality that links to another dataset (e.g. <i>OCD-ID</i>)	If the element is invalid or not present, then the implementation is required to ignore it.
IsMailOnly	xs:boolean	Optional	Single	Determines if the locality runs mail-only elections. If this is true, then all precincts a part of the locality will also run mail-only elections. Drop boxes may be used in addition to this flag using a <i>polling location</i> record configured as a Drop Box.	If the field is missing or invalid, the implementation is required to assume <i>IsMailOnly</i> is false.
Name	xs:string	Required	Single	Specifies the name of a locality.	If the field is invalid, then the implementation is required to ignore the <i>Locality</i> element containing it.
PollingLocations	xs:IDREFS	Optional	Single	Specifies a link to a set of the locality's <i>ref:polling locations</i> <code><single-xml-polling-location></code> '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 <i>State</i> .	If the field is invalid, then the implementation is required to ignore the <i>Locality</i> element containing it.
Type	<i>DistrictType</i>	Optional	Single	Defines the kind of locality (e.g. county, town, et al.), which is one of the various <i>DistrictType enumerations</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Other-Type	xs:string	Optional	Single	Allows for defining a type of locality that falls outside the options listed in <i>DistrictType</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

```

1 <Locality id="loc70001">
2   <ElectionAdministrationId>ea40001</ElectionAdministrationId>
3   <ExternalIdentifiers>
4     <ExternalIdentifier>
5       <Type>ocd-id</Type>
6       <Value>ocd-division/country:us/state:va/county:albemarle</Value>
7     </ExternalIdentifier>
8   </ExternalIdentifiers>
9   <IsMailOnly>true</IsMailOnly>
10  <Name>ALBEMARLE COUNTY</Name>
11  <StateId>st51</StateId>
12  <Type>county</Type>
13 </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	single-xml-contact-information	Optional	Repeats	Links to the single-xml-contact-information element associated with the office.	If the element is invalid or not present, then the implementation is required to ignore it.
Description	<i>InternationalizedText</i>	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.
ElectionDistrictId	<code>xs:IDREF</code>	Required	Single	Links to the <i>ElectionDistrict</i> element associated with the office.	If the field is invalid or not present, the implementation is required to ignore the <code>Office</code> element containing it.
ExternalIdentifiers	<i>ExternalIdentifiers</i>	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	<code>xs:date</code>	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	<code>xs:boolean</code>	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	<i>InternationalizedText</i>	Required	Single	The name of the office.	If the field is invalid or not present, the implementation is required to ignore the <code>Office</code> element containing it.
OfficeHolderPersonIds	<code>xs:IDREFS</code>	Optional	Single	Links to the <i>Person</i> 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	single-xml-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	<i>OfficeTermType</i>	Optional	Single	Specifies the type of office term (see <i>OfficeTermType</i> for valid values).	If the field is invalid or not present, the implementation is required to ignore the <code>Office</code> element containing it.
StartDate	<code>xs:date</code>	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	<code>xs:date</code>	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.

```
1 <Office id="off0000">
2   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
3   <FilingDeadline>2013-01-01</FilingDeadline>
4   <IsPartisan>>false</IsPartisan>
5   <Name>
6     <Text language="en">Governor</Text>
7   </Name>
8   <Term>
9     <Type>full-term</Type>
10  </Term>
11 </Office>
```

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	<i>Internationalized-Text</i>	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]	<i>Internationalized-Text</i>	Optional	Single	Contains the hours (in local time) that the location is open (NB: <i>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</i>).	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	xs:ID	Optional	Single	References an <i>HoursOpen</i> 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	single-xml-lat-lng	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.
2.1. XML Specification					45

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
```

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

OrderedContest

OrderedContest encapsulates links to the information that comprises a contest and potential ballot selections. OrderedContest elements can be collected within a *BallotStyle* to accurately 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 single-xml-contest-base.	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 single-xml-ballot-selection-base.	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 single-xml-ballot-selection-base-extended elements referenced by the underlying single-xml-contest-base-extended elements.

```

1 <OrderedContest id="oc20003abc">
2   <ContestId>cc20003</ContestId>
3   <OrderedBallotSelectionIds>cs10961 cs10962 cs10963</OrderedBallotSelectionIds>
4 </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
Ab- bre- via- tion	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	single-xml-html-color-string	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.
Ex- ter- nal- iden- ti- fiers	<i>External-identifiers</i>	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.
Lead- er- Per- son- Ids	xs:IDREFS	Optional	Single	A reference of <i>Person</i> elements which are leaders of the <i>Party</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Lo- goUri	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	<i>Internationalized-Text</i>	Optional	Single	The name of the party.	If the element is invalid or not present, then the implementation is required to ignore it.

```

1 <Party id="par0001">
2   <Abbreviation>REP</Abbreviation>
3   <Color>e91d0e</Color>
4   <IsWriteIn>>false</IsWriteIn>
5   <LeaderPersonIds>per01</LeaderPersonIds>
6   <Name>
7     <Text language="en">Republican</Text>
8   </Name>
9 </Party>

```

HtmlColorString

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

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

PartyContest

An extension of `single-xml-contest-base` 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).

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
Ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Se- lec- tion- Ids	xs:IDREFS	Optional	Single	References a set of <code>BallotSelections</code> , which could be of any selection type that extends <code>single-xml-ballot-selection-base</code> .	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Sub- Ti- tle	<i>Internationalized-Text</i>	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.
Bal- lot- Ti- tle	<i>Internationalized-Text</i>	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.
Elec- toral- Dis- tric- tId	xs:IDREF	Required	Single	References an <i>ElectoralDistrict</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the <code>ContestBase</code> element containing it.
Elec- torate- Spec- ifi- ca- tion	<i>Internationalized-Text</i>	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.
Ex- ter- nal- iden- ti- fiers	<i>ExternalIdentifiers</i>	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.
Has- Ro- ta- tion	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	Required	Single	Name of the contest, not necessarily how it appears on the ballot (NB: <code>BallotTitle</code> should be used for this purpose).	If the field is invalid, then the implementation is required to ignore the <code>ContestBase</code>

PartySelection

This element extends single-xml-ballot-selection-base 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 <i>Party</i> 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.

BallotSelectionBase

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Sequence-Order	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 <i>OrderedBallotSlectionIds</i> in <i>Ordered-Contest</i> .	If the field is invalid or not present, then the implementation is required to ignore 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
Contact-Information	single-xml-contact-information	Optional	Repeats	Refers to the associated single-xml-contact-information.	If the element is invalid or not present, then the implementation is required to ignore it.
Date-Of-Birth	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.
External-Identifiers	<i>External-Identifiers</i>	Optional	Single	Identifiers for this person.	If the element is invalid or not present, then the implementation is required to ignore it.
First-Name	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.
Full-Name	<i>Internationalized-Text</i>	Optional	Single	Specifies a person's full name (NB: this information is <i>InternationalizedText</i> 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.
Last-Name	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.
Middle-Name	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 <i>Party</i> . 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 <i>Candidate PartyId</i> .	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	<i>Internationalized-Text</i>	Optional	Single	Specifies a person's profession (NB: this information is <i>InternationalizedText</i> 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.
Suffix	xs:string	Optional	Single	Specifies a suffix associated with a person (e.g. Jr.).	If the field is invalid or not present, then


```
1 <Person id="per50001">
2   <ContactInformation label="ci60002">
3     <Email>rwashburne@albemarle.org</Email>
4     <Phone>4349724173</Phone>
5   </ContactInformation>
6   <FirstName>RICHARD</FirstName>
7   <LastName>WASHBURN</LastName>
8   <MiddleName>J.</MiddleName>
9   <Nickname>JAKE</Nickname>
10  <Title>
11    <Text language="en">General Registrar Physical</Text>
12  </Title>
13 </Person>
```

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	<i>Internationalized-Text</i>	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]	<i>Internationalized-Text</i>	Optional	Single	Contains the hours (in local time) that the location is open (NB: <i>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</i>).	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	xs:ID	Optional	Single	References an <i>HoursOpen</i> 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	single-xml-lat-lng	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
```

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

PollingLocation

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
AddressStructured	single-structured-simple-address-type	Optional	Single	Represents the various structured parts of an address to a polling location.	One of AddressStructured and AddressLine should be present for a given Polling Location. If none is present, the implementation is required to ignore the <code>PollingLocation</code> element containing it.
AddressLine	<code>xs:string</code>	Optional	Repeats	Represents the various parts of an address to a polling location.	One of <code>AddressStructured</code> and <code>AddressLine</code> should be present for a given Polling Location. If none is present, the implementation is required to ignore the <code>PollingLocation</code> element containing it.
Directions	<i>Internationalized-Text</i>	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]	<i>Internationalized-Text</i>	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 <i>HoursOpen</i> 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	<code>xs:IDRef</code>	Optional	Single	Links to an <i>HoursOpen</i> 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	<code>xs:boolean</code>	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	<code>xs:boolean</code>	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	single-xml-llng	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	<code>xs:string</code>	Optional	Single	Name of the polling location.	If the field is invalid or not present, then the implementation is required to ignore it.
PhotoUri	<code>xs:anyURI</code>	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.

```

1 <PollingLocation id="p100000">
2   <AddressLine>2775 Hydraulic Rd Charlottesville, VA 22901</AddressLine>
3   <HoursOpenId>hours0002</HoursOpenId>
4   <IsDropBox>true</IsDropBox>

```

(continues on next page)

(continued from previous page)

```

5     <IsEarlyVoting>true</IsEarlyVoting>
6     <LatLng>
7         <Latitude>38.009939</Latitude>
8         <Longitude>-78.506204</Longitude>
9     </LatLng>
10    <Name>ALBERMARLE HIGH SCHOOL</Name>
11</PollingLocation>
12<!-- Or: -->
13<PollingLocation id="pl00000">
14    <AddressStructured>
15        <Line1>2775 Hydraulic Rd</Line1>
16        <City>CHARLOTTESVILLE</City>
17        <State>VA</State>
18        <Zip>22901</Zip>
19    </AddressStructured>
20    <HoursOpenId>hours0002</HoursOpenId>
21    <IsDropBox>true</IsDropBox>
22    <IsEarlyVoting>true</IsEarlyVoting>
23    <LatLng>
24        <Latitude>38.009939</Latitude>
25        <Longitude>-78.506204</Longitude>
26    </LatLng>
27    <Name>ALBERMARLE HIGH SCHOOL</Name>
28</PollingLocation>

```

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:double	Required	Single	The latitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
Longitude	xs:double	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/Ing. 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.

SimpleAddressType

A SimpleAddressType represents a structured address.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Line1	xs:string	Required	Single	The address line for a structured address. Should include the street number, street name, and any prefix and suffix.	If no Line1 is provided, the implementation should ignore the SimpleAddressType.
Line2	xs:string	Optional	Single	Additional field for an address	If no Line2 is provided, the implementation should ignore it.
Line3	xs:string	Optional	Single	Additional field for an address	If no Line3 is provided, the implementation should ignore it.
City	xs:string	Required	Single	The City value of a structured address.	If City is not provided, the implementation should ignore the SimpleAddressType.
State	xs:string	Required	Single	The State value of a structured address.	If State is not provided, the implementation should ignore the SimpleAddressType.
Zip	xs:string	Optional	Single	The ZIP code of a structured address.	If Zip is not provided, the implementation should ignore the SimpleAddressType.

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).

Voters can be assigned to a precinct in two ways. A voter location modeled by StreetSegment is assigned to a precinct by StreetSegment.PrecinctId. Alternatively, a precinct's spatial boundary can be modeled with Precinct.SpatialBoundary. Any registered voter address contained within the spatial boundary of the precinct is assigned to that precinct.

Tag	Data Type	Required	Repeats?	Description	Error Handling
Ballot-StyleId	xs:ID	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.
Electoral-District-Ids	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.
External-Identifiers	External-Identifiers	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.
Is-MailOnly	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 <i>IsMailOnly</i> is false.
LocalityId	xs:IDREF	Required	Single	Links to the Locality that comprises the precinct.	If the field is invalid, then the implementation is required to ignore the <i>Precinct</i> element containing it.
Name	xs:string	Required	Single	Specifies the precinct's name (or number if no name exists).	If the field is invalid, then the implementation is required to ignore the <i>Precinct</i> 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 <i>Name</i> 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.
Polling-Location-Ids	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.
Precinct-Split-Name	xs:string	Optional	Single	If this field is empty, then this <i>Precinct</i> object represents a full precinct. If this field is present, then this <i>Precinct</i> object represents one portion of a split precinct. Each <i>Precinct</i> object that represents one portion of a split precinct must have the same <i>Name</i> value, but different <i>PrecinctSplitName</i> values. See the <i>sample_feed.xml</i> file for examples.	If the field is invalid or not present, then the implementation is required to ignore it.
Spatial	single	Optional	Single	Defines the spatial boundary of the precinct. All voter addresses	If the element

2.1. XML Specification

```
1 <Precinct id="pre90111">
2   <BallotStyleId>bs00010</BallotStyleId>
3   <ElectoralDistrictIds>ed60129 ed60311 ed60054</ElectoralDistrictIds>
4   <IsMailOnly>>false</IsMailOnly>
5   <LocalityId>loc70001</LocalityId>
6   <Name>203 - GEORGETOWN</Name>
7   <Number>0203</Number>
8   <PollingLocationIds>p181274</PollingLocationIds>
9   <SpatialBoundary>
10    <ExternalGeospatialFeature>
11     <ExternalFileId>ef1</ExternalFileId>
12     <FileFormat>shp</FileFormat>
13     <FeatureIdentifier>
14      <Index>3</Index>
15     </FeatureIdentifier>
16    </ExternalGeospatialFeature>
17   </SpatialBoundary>
18 </Precinct>
19 <!--
20   Precinct split. Name and PollingLocationIds are the same but
21   PrecinctSplitName is present, the ElectoralDistrictIds are different,
22   and the BallotStyleId is different.
23 -->
24 <Precinct id="pre90348sp0000">
25   <BallotStyleId>bs00002</BallotStyleId>
26   <ElectoralDistrictIds>ed60129 ed60054 ed60150</ElectoralDistrictIds>
27   <IsMailOnly>>false</IsMailOnly>
28   <LocalityId>loc70001</LocalityId>
29   <Name>201 - JACK JOUETT</Name>
30   <Number>0201</Number>
31   <PollingLocationIds>p100000 p181273 p181662</PollingLocationIds>
32   <PrecinctSplitName>0000</PrecinctSplitName>
33 </Precinct>
34 <Precinct id="pre90348sp0001">
35   <BallotStyleId>bs00015</BallotStyleId>
36   <ElectoralDistrictIds>ed60129 ed60054 ed60267</ElectoralDistrictIds>
37   <IsMailOnly>>false</IsMailOnly>
38   <LocalityId>loc70001</LocalityId>
39   <Name>201 - JACK JOUETT</Name>
40   <Number>0201</Number>
41   <PollingLocationIds>p100000 p181273 p181662</PollingLocationIds>
42   <PrecinctSplitName>0001</PrecinctSplitName>
43 </Precinct>
```

SpatialBoundary

The `SpatialBoundary` object defines a boundary in space. This boundary is usually defined by one or more discrete, closed polygonal shapes.

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
External-GeospatialFeature	single-xml-external-geospatial-feature	Required	Single	The spatial boundary defined by a geospatial feature that is external to the VIP feed.	If the element is invalid, then the implementation is required to ignore the SpatialBoundary element containing it.

```

1 <SpatialBoundary>
2   <ExternalGeospatialFeature>
3     <ExternalFileId>ef1</ExternalFileId>
4     <FileFormat>shp</FileFormat>
5     <FeatureIdentifier>
6       <Index>3</Index>
7     </FeatureIdentifier>
8   </ExternalGeospatialFeature>
9 </SpatialBoundary>

```

ExternalGeospatialFeature

The `ExternalGeospatialFeature` object contains a reference to a geospatial feature (one or more shapes) contained in a separate file external to the VIP feed.

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
External-FileId	<code>xs:IDREF</code>	Required	Single	Links to the ExternalFile containing the geospatial shape(s) that define the feature's boundary.	If the field is invalid, then the implementation is required to ignore the ExternalGeospatialFeature element containing it.
File-Format	Geospatial-Format	Required	Single	The format of the geospatial file.	If the field is invalid, then the implementation is required to ignore the ExternalGeospatialFeature element containing it.
FeatureIdentifier	single-xml-feature-identifier	Required	Repeats	Identifiers indicating which specific shape(s) to use from the geospatial file. These refer to identifiers within the referenced external file. This is a repeated field in the XML specification, but a scalar field in the CSV specification. If more than one identifier is required with the CSV specification, multiple values can be provided by delimited by space.	If the element is invalid, then the implementation is required to ignore the ExternalGeospatialFeature element containing it.

FeatureIdentifier

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
Index	<code>xs:int</code>	Optional	Single	The index value for the shapefile feature.	If the field is invalid or not present, then the implementation is required to ignore it.

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 <i>Candidate</i> being retained.	If the field is invalid or not present, the implementation is required to ignore the <code>RetentionContest</code> 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.

```
1 <RetentionContest id="rc40001">
2   <BallotSelectionIds>rc40001a rc40001b</BallotSelectionIds>
3   <BallotTitle>
4     <Text language="en">Retention of Supreme Court Justice</Text>
5     <Text language="es">La retención de juez de la Corte Suprema</Text>
6   </BallotTitle>
7   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
8   <Name>Judicial Retention, Supreme Court</Name>
9   <CandidateId>can14444</CandidateId>
10  <OfficeId>off20006</OfficeId>
11 </RetentionContest>
```

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	single-xml-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.
IsOnly-ByAppointment	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.
IsOr-ByAppointment	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.
IsSubject-ToChange	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.
Start-Date	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.
End-Date	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
Start-Time	single-xml-time-with-zone	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.
End-Time	single-xml-time-with-zone	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

```

1 <HoursOpen id="hours0001">
2   <Schedule>
3     <Hours>
4       <StartTime>06:00:00-05:00</StartTime>

```

(continues on next page)

(continued from previous page)

```

5      <EndTime>12:00:00-05:00</EndTime>
6    </Hours>
7    <Hours>
8      <StartTime>13:00:00-05:00</StartTime>
9      <EndTime>19:00:00-05:00</EndTime>
10   </Hours>
11   <StartDate>2013-11-05</StartDate>
12   <EndDate>2013-11-05</EndDate>
13 </Schedule>
14 </HoursOpen>

```

SimpleAddressType

A SimpleAddressType represents a structured address.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Line1	xs:string	Required	Single	The address line for a structured address. Should include the street number, street name, and any prefix and suffix.	If no Line1 is provided, the implementation should ignore the SimpleAddressType.
Line2	xs:string	Optional	Single	Additional field for an address	If no Line2 is provided, the implementation should ignore it.
Line3	xs:string	Optional	Single	Additional field for an address	If no Line3 is provided, the implementation should ignore it.
City	xs:string	Required	Single	The City value of a structured address.	If City is not provided, the implementation should ignore the SimpleAddressType.
State	xs:string	Required	Single	The State value of a structured address.	If State is not provided, the implementation should ignore the SimpleAddressType.
Zip	xs:string	Optional	Single	The ZIP code of a structured address.	If Zip is not provided, the implementation should ignore the SimpleAddressType.

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 <code>Source</code> 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 <code>Source</code> element containing it.
Date-Time	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 the <code>Source</code> element containing it.
Description	<i>InternationalizedText</i>	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.
Feed-ContactInformation	single-xml-contact-information	Optional	Single	Reference to the <i>Person</i> who will respond to inquiries about the information contained within the file.	If the element 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 the <code>Source</code> element containing it.

```

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

```

SpatialBoundary

The `SpatialBoundary` object defines a boundary in space. This boundary is usually defined by one or more discrete, closed polygonal shapes.

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
External-GeospatialFeature	single-xml-external-geospatial-feature	Required	Single	The spatial boundary defined by a geospatial feature that is external to the VIP feed.	If the element is invalid, then the implementation is required to ignore the SpatialBoundary element containing it.

```

1 <SpatialBoundary>
2   <ExternalGeospatialFeature>
3     <ExternalFileId>ef1</ExternalFileId>
4     <FileFormat>shp</FileFormat>
5     <FeatureIdentifier>
6       <Index>3</Index>
7     </FeatureIdentifier>
8   </ExternalGeospatialFeature>
9 </SpatialBoundary>

```

ExternalGeospatialFeature

The `ExternalGeospatialFeature` object contains a reference to a geospatial feature (one or more shapes) contained in a separate file external to the VIP feed.

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
External-FileId	<code>xs:IDREF</code>	Required	Single	Links to the ExternalFile containing the geospatial shape(s) that define the feature's boundary.	If the field is invalid, then the implementation is required to ignore the ExternalGeospatialFeature element containing it.
File-Format	<i>Geospatial-Format</i>	Required	Single	The format of the geospatial file.	If the field is invalid, then the implementation is required to ignore the ExternalGeospatialFeature element containing it.
FeatureIdentifier	single-xml-feature-identifier	Required	Repeats	Identifiers indicating which specific shape(s) to use from the geospatial file. These refer to identifiers within the referenced external file. This is a repeated field in the XML specification, but a scalar field in the CSV specification. If more than one identifier is required with the CSV specification, multiple values can be provided by delimited by space.	If the element is invalid, then the implementation is required to ignore the ExternalGeospatialFeature element containing it.

FeatureIdentifier

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
Index	<code>xs:int</code>	Optional	Single	The index value for the shapefile feature.	If the field is invalid or not present, then the implementation is required to ignore it.

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
Election-AdministrationId	xs:ID	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.
External-identifiers	<i>External-identifiers</i>	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	Specifies the name of a state, such as Alabama.	If the field is invalid, then the implementation is required to ignore the State element containing it.
Polling-Location-Ids	xs:IDREFS	Optional	Single	Specifies a link to the state's <i>polling locations</i> . 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 <i>Precinct</i> element.	If the field is invalid or not present, then the implementation is required to ignore it.

```

1 <State id="st51">
2   <ElectionAdministrationId>ea40133</ElectionAdministrationId>
3   <ExternalIdentifiers>
4     <ExternalIdentifier>
5       <Type>ocd-id</Type>
6       <Value>ocd-division/country:us/state:va</Value>
7     </ExternalIdentifier>
8   </ExternalIdentifiers>
9   <Name>Virginia</Name>
10 </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?	Re-usable?	Description	Error Handling
Address-Direction	<code>xs:string</code>	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	<code>xs:string</code>	Required	Single	The city specifies the city or town of the address.	If the field is invalid, then the implementation is required to ignore the <code>StreetSegment</code> element containing it.
IncludesAllAddresses	<code>xs:boolean</code>	Optional	Single	Specifies if the segment covers every address on this street. If this is <i>true</i> , then the values of StartHouseNumber and EndHouseNumber should be ignored. The value of OddEvenBoth must be <i>both</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
IncludesAllStreets	<code>xs:boolean</code>	Optional	Single	Specifies if the segment covers every street in this city. If this is <i>true</i> , 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.
Odd-Even-Both	<i>OddEvenBoth</i>	Optional	Single	Specifies whether the odd side of the street (in terms of house numbers), the even side, or both are included in the street segment.	If the field is not present or invalid, the implementation is required to ignore the <code>StreetSegment</code> containing it.
PrecinctId	<code>xs:string</code>	Required	Single	References the <i>Precinct</i> that contains the entire street segment. If a precinct has a single-xml-spatial-boundary which also contains the entire street segment, then the precinct assignment from the segment will be preferred over the assignment defined by the spatial boundary.	If the field is invalid, then the implementation is required to ignore the <code>StreetSegment</code> element containing it.
StartHouseNumber	<code>xs:integer</code>	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 <code>StreetSegment</code> element containing it. If the StartHouseNumber is greater than the EndHouseNumber , the implementation should ignore the element containing them.
EndHouseNumber	<code>xs:integer</code>	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 <code>StreetSegment</code> element containing it. If the EndHouseNumber is less than the StartHouseNumber , the implementation should ignore the element containing it.
HouseNumber-Prefix	<code>xs:string</code>	Optional	Single	Part of a street address. It may contain letters or slashes (e.g., ‘B’ in ‘B22 Main St’). If this value is present then StartHouseNumber must be equal to EndHouseNumber . This field cannot be used if IncludesAllAddresses or IncludesAllStreets are true.	If the field is invalid or not present, then the implementation is required to ignore it.
68					Chapter 2. XML Documentation
HouseNumber	<code>xs:string</code>	Optional	Single	Part of a street address. It may contain letters or slashes (e.g., 1/2 in ‘22 1/2 Main St’). If this value is present then StartHouse-	If the field is invalid or not present, then the implementation is required to ignore it.


```

1 <StreetSegment id="ss999999">
2   <City>Charlottesville</City>
3   <IncludesAllAddresses>true</IncludesAllAddresses>
4   <OddEvenBoth>both</OddEvenBoth>
5   <PrecinctId>pre99999</PrecinctId>
6   <State>VA</State>
7   <StreetName>CHAPEL HILL</StreetName>
8   <StreetSuffix>RD</StreetSuffix>
9   <Zip>22901</Zip>
10 </StreetSegment>
11 <StreetSegment id="ss309904">
12   <City>GREENWOOD</City>
13   <OddEvenBoth>both</OddEvenBoth>
14   <PrecinctId>pre92145</PrecinctId>
15   <StartHouseNumber>1</StartHouseNumber>
16   <EndHouseNumber>201</EndHouseNumber>
17   <State>VA</State>
18   <StreetName>MISTY MOUNTAIN</StreetName>
19   <StreetSuffix>RD</StreetSuffix>
20   <Zip>22943</Zip>
21 </StreetSegment>
22 <StreetSegment id = "ss1"
23   <City>GREENWOOD</City>
24   <OddEvenBoth>both</OddEvenBoth>
25   <PrecinctId>pre92145</PrecinctId>
26   <StartHouseNumber>1</StartHouseNumber>
27   <EndHouseNumber>1</EndHouseNumber>
28   <HouseNumberPrefix>B</HouseNumberPrefix>
29   <HouseNumberSuffix>1/2</HouseNumberSuffix>
30   <State>VA</State>
31   <StreetName>MISTY MOUNTAIN</StreetName>
32   <StreetSuffix>RD</StreetSuffix>
33   <Zip>22943</Zip>
34 </StreetSegment>

```

Term

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Type	<i>OfficeTermType</i>	Optional	Single	Specifies the type of office term (see <i>OfficeTermType</i> for valid values).	If the field is invalid or not present, the implementation is required to ignore the <i>Office</i> element containing it.
Start-Date	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.
End-Date	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.

```

1 <Office id="off0000">
2   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
3   <FilingDeadline>2013-01-01</FilingDeadline>
4   <IsPartisan>false</IsPartisan>
5   <Name>
6     <Text language="en">Governor</Text>

```

(continues on next page)

(continued from previous page)

```

7   </Name>
8   <Term>
9     <Type>full-term</Type>
10  </Term>
11 </Office>

```

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

```

1 <HoursOpen id="hours0001">
2   <Schedule>
3     <Hours>
4       <StartTime>06:00:00-05:00</StartTime>
5       <EndTime>12:00:00-05:00</EndTime>
6     </Hours>
7     <Hours>
8       <StartTime>13:00:00-05:00</StartTime>
9       <EndTime>19:00:00-05:00</EndTime>
10    </Hours>
11    <StartDate>2013-11-05</StartDate>
12    <EndDate>2013-11-05</EndDate>
13  </Schedule>
14 </HoursOpen>

```

VoterService

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
Contact- Informa- tion	single-xml- contact- information	Op- tional	Sin- gle	The contact for a particular voter service.	If the element is invalid or not present, then the implementation is required to ignore it.
Descrip- tion	<i>InternationalizedText</i>	Op- tional	Sin- gle	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
Elec- tionOffi- cialPer- sonId	xs:IDREF	Op- tional	Sin- gle	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
Type	<i>VoterServiceType</i>	Op- tional	Sin- gle	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Other- Type	xs:string	Op- tional	Sin- gle	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.

Enumerations

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	

ChecksumAlgorithm

Tag	Description
sha-256	256-bit cryptographic hash algorithm of the SHA-2 family
sha-512	512-bit cryptographic hash algorithm of the SHA-2 family

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 <i>OtherType</i> field to describe it.

GeospatialFormat

Geospatial file formats that are supported by the VIP specification.

Tag	Description
shp	ESRI Shapefile (reference)

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.
ap-proval	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).
super-majority	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 department handles the selection and management of polling places.
voter-registration	The department that manages voter registration.
other	Any other service not covered by the above descriptions.

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
Con-State-ment	<i>Internationalized-Text</i>	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	<i>Internationalized-Text</i>	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.
Full-Text	<i>Internationalized-Text</i>	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	<code>xs:anyURI</code>	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	<i>Internationalized-Text</i>	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 <i>competing initiatives</i> , information about their effect on the passage of the <i>BallotMeasureContest</i> would go here.	If the element is invalid or not present, then the implementation is required to ignore it.
ProStatement	<i>Internationalized-Text</i>	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.
Summary-Text	<i>Internationalized-Text</i>	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	<i>Ballot-Measure-Type</i>	Optional	Single	Specifies the particular type of ballot measure. Must be one of the valid <i>BallotMeasureType</i> options.	If the field is invalid or not present, then the implementation is required to ignore it.
2.1. XML Specification					75
Other-Type	<code>xs:string</code>	Optional	Single	Allows for cataloging a new <i>BallotMeasureType</i> option, when Type is specified as “other.”	If the field is invalid or not present, then the

```
1 <BallotMeasureContest id="bmc30001">
2   <BallotSelectionIds>bms30001a bms30001b</BallotSelectionIds>
3   <BallotTitle>
4     <Text language="en">State of the State</Text>
5     <Text language="es">Estado del Estado.</Text>
6   </BallotTitle>
7   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
8   <Name>Referendum on Virginia</Name>
9   <ConStatement label="bmc30001con">
10    <Text language="en">This is no good.</Text>
11    <Text language="es">Esto no es bueno.</Text>
12  </ConStatement>
13  <EffectOfAbstain label="bmc30001abs">
14    <Text language="en">Nothing will happen.</Text>
15    <Text language="es">Nada pasará.</Text>
16  </EffectOfAbstain>
17  <ProStatement label="bmc30001pro">
18    <Text language="en">Everything will be great.</Text>
19    <Text language="es">Todo va a estar bien.</Text>
20  </ProStatement>
21  <Type>referendum</Type>
22 </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
Ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Se- lec- tion- Ids	xs:IDREFS	Optional	Single	References a set of <code>BallotSelections</code> , which could be of any selection type that extends <i>BallotSelectionBase</i> .	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Sub- Ti- tle	<i>Internationalized-Text</i>	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.
Bal- lot- Ti- tle	<i>Internationalized-Text</i>	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.
Elec- toral- Dis- tric- tId	xs:IDREF	Required	Single	References an <i>ElectoralDistrict</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the <code>ContestBase</code> element containing it.
Elec- torate- Spec- ifi- ca- tion	<i>Internationalized-Text</i>	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.
Ex- ter- nal- iden- ti- fiers	<i>External-identifiers</i>	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.
Has- Ro- ta- tion	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	Required	Single	Name of the contest, not necessarily how it appears on the ballot (NB: <code>BallotTitle</code> should be used for this purpose).	If the field is invalid, then the implementation is required to ignore the

2.1. XML Specification

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	Re- quired?	Re- peats?	Description	Error Handling
Se- lec- tion	<i>InternationalizedText</i>	Re- quired	Single	Selection text for a <i>BallotMeasureContest</i>	If the element is invalid or not present, the implementation is required to ignore the <i>BallotMeasureSelection</i> containing it.

```

1 <BallotMeasureSelection id="bms30001a">
2   <Selection label="bms30001at">
3     <Text language="en">Yes</Text>
4     <Text language="es">Sí</Text>
5   </Selection>
6 </BallotMeasureSelection>
7 <BallotMeasureSelection id="bms30001b">
8   <Selection label="bms30001bt">
9     <Text language="en">No</Text>
10    <Text language="es">No</Text>
11  </Selection>
12 </BallotMeasureSelection>

```

BallotSelectionBase

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

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
Se- quence- Order	<i>xs:integer</i>	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 <i>OrderedBallotSlectionIds</i> in <i>OrderedContest</i> .	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	Re- quired?	Re- peats?	Description	Error Handling
ImageUri	<i>xs:anyURI</i>	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.
Ordered- Con- testIds	<i>xs:IDREF</i>	Optional	Single	Reference to a set of <i>OrderedContest</i>	If the field is invalid or not present, then the implementation is required to ignore it.
PartyIds	<i>xs:IDREF</i>	Optional	Single	Reference to a set of <i>:ref:multi-xml-party</i> 's.	If the field is invalid or not present, then the implementation is required to ignore it.

```
1 <BallotStyle id="bs00000">
2   <OrderedContestIds>oc20003 oc20004 oc20005 oc20025 oc20355 oc20449</
   ↳OrderedContestIds>
3 </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
Ballot-Name	<i>Internationalized-Text</i>	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, then the implementation is required to ignore the <code>Candidate</code> element containing it.
Contact-Information	<i>ContactInformation</i>	Optional	Single	Contact and physical address information for this Candidate and/or their campaign (see <i>ContactInformation</i>).	If the element is invalid or not present, then the implementation is required to ignore it.
External-Identifiers	<i>External-Identifiers</i>	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.
File-Date	<code>xs:date</code>	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	<code>xs:boolean</code>	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	<code>xs:boolean</code>	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	<code>xs:IDREF</code>	Optional	Single	Reference to a <i>Party</i> 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	<code>xs:IDREF</code>	Optional	Single	Reference to a <i>Person</i> element with additional information about the candidate.	If the field is invalid or not present, then the implementation is required to ignore it.
Post-Election-Status	<i>CandidatePost-Election-Status</i>	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.
Pre-Election-Status	<i>CandidatePre-Election-Status</i>	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.

```

1 <Candidate id="can10961">
2   <BallotName>
3     <Text language="en">Ken T. Cuccinelli II</Text>
4   </BallotName>
5   <PartyId>par0001</PartyId>
6   <PersonId>per10961</PersonId>

```

(continues on next page)

(continued from previous page)

</Candidate>

CandidateContest

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

Tag	Data Type	Required	Repeats?	Description	Error Handling
Number-Elected	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 <i>Office</i> 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 <i>BallotSelectionIds</i> . E.g., if the various <i>BallotSelectionIds</i> reference <i>CandidateSelection</i> elements which reference the candidate for President first and Vice-President second, the <i>OfficeIds</i> 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.
Primary-PartyIds	xs:IDREFS	Optional	Single	References <i>Party</i> 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.

```

1 <CandidateContest id="cc20003">
2   <BallotSelectionIds>cs10961 cs10962 cs10963</BallotSelectionIds>
3   <BallotTitle>
4     <Text language="en">Governor of Virginia</Text>
5   </BallotTitle>
6   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
7   <Name>Governor</Name>
8   <NumberElected>1</NumberElected>
9   <OfficeId>off0000</OfficeId>

```

(continues on next page)

(continued from previous page)

```
10 <VotesAllowed>1</VotesAllowed>
11 </CandidateContest>
```

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
Ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Se- lec- tion- Ids	xs:IDREFS	Optional	Single	References a set of <code>BallotSelections</code> , which could be of any selection type that extends <code>BallotSelectionBase</code> .	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Sub- Ti- tle	<i>Internationalized-Text</i>	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.
Bal- lot- Ti- tle	<i>Internationalized-Text</i>	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.
Elec- toral- Dis- tric- tId	xs:IDREF	Required	Single	References an <code>ElectoralDistrict</code> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the <code>ContestBase</code> element containing it.
Elec- torate- Spec- ifi- ca- tion	<i>Internationalized-Text</i>	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.
Ex- ter- nal- den- ti- fiers	<i>External-identifiers</i>	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.
Has- Ro- ta- tion	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	Required	Single	Name of the contest, not necessarily how it appears on the ballot (NB: <code>BallotTitle</code> should be used for this purpose).	If the field is invalid, then the implementation is required to ignore the

2.1. XML Specification

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 <i>Candidate</i> 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 <i>Party</i> 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.

```

1 <CandidateSelection id="cs10861">
2   <CandidateIds>can10861a can10861b</CandidateIds>
3   <EndorsementPartyIds>par0001</EndorsementPartyIds>
4 </CandidateSelection>

```

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 <i>OrderedBallotSlectionIds</i> in <i>OrderedContest</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

Election

The Election object represents an Election Day, which usually consists of many individual contests and/or referenda. A feed may **not** contain multiple Election objects. 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 <i>Date</i> 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.
Election-Type	Internationalized-Text	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 <i>State</i> element where the election is being held.	If the field is invalid, then the implementation is required to ignore the Election element containing it.
Is-Statewide	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	Internationalized-Text	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	Internationalized-Text	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.
Absentee-Ballot-Info	Internationalized-Text	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.
2.1. XML Specification					85
PollingHours	Hours	Optional	Single	Contains the hours (in local time) that Election Day polling loca-	If the element

```
1 <Election id="ele30000">
2   <AbsenteeRequestDeadline>2013-10-30</AbsenteeRequestDeadline>
3   <Date>2013-11-05</Date>
4   <ElectionType>
5     <Text language="en">General</Text>
6     <Text language="es">Generales</Text>
7   </ElectionType>
8   <HasElectionDayRegistration>>false</HasElectionDayRegistration>
9   <HoursOpenId>hours0001</HoursOpenId>
10  <IsStatewide>true</IsStatewide>
11  <Name>
12    <Text language="en">2013 Statewide General</Text>
13  </Name>
14  <RegistrationDeadline>2013-10-15</RegistrationDeadline>
15  <ResultsUri>http://www.sbe.virginia.gov/ElectionResults.html</ResultsUri>
16  <StateId>st51</StateId>
17 </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.
BallotTrackingUri	xs:anyURI	Optional	Single	Specifies the web address for tracking information for a ballot cast by mail	If the field is invalid or not present, then the implementation is required to ignore it.
BallotProvisionalTrackingUri	xs:anyURI	Optional	Single	Specifies the web address for tracking information for a provisional ballot. To support EAC guidelines for “Processing Provisional Ballots” (https://www.eac.gov/research-and-data/provisional-voting/)	If the field is invalid or not present, then the implementation is required to ignore it.
Department	<i>Department</i>	Required	Repeats	Describes the administrative body for a particular voter service.	There must be at least one valid <i>Department</i> in each <i>ElectionAdministration</i> element. If no valid <i>Department</i> objects are present, the implementation is required to ignore the <i>ElectionAdministration</i> object that contains it/them.
ElectionNotice	<i>ElectionNotice</i>	Optional	Single	A place for election administrators to post last minute and emergency notifications pertaining to the election.	If the element is invalid or not present, then the implementation is required to ignore it.
ElectionUri	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.

```

1 <ElectionAdministration id="ea40133">
2   <AbsenteeUri>http://www.sbe.virginia.gov/absenteevoting.html</AbsenteeUri>
3   <AmIRegisteredUri>https://www.vote.virginia.gov/</AmIRegisteredUri>

```

(continues on next page)

(continued from previous page)

```

4    <BallotTrackingUri>https://www.vote.virginia.gov/</BallotTrackingUri>
5    <BallotProvisionalTrackingUri>https://www.vote.virginia.gov/</
6    <BallotProvisionalTrackingUri>
7    <Department>
8      <ContactInformation label="ci60000">
9        <AddressLine>Washington Building First Floor</AddressLine>
10       <AddressLine>1100 Bank Street</AddressLine>
11       <AddressLine>Richmond, VA 23219</AddressLine>
12       <Name>State Board of Elections</Name>
13     </ContactInformation>
14   </Department>
15   <ElectionNotice>
16     <NoticeText>
17       <Text language="en">This is an emergency notification for this election.</Text>
18     </NoticeText>
19     <NoticeURI>https://www.yadayada.gov</NoticeURI>
20   </ElectionNotice>
21   <ElectionsUri>http://www.sbe.virginia.gov/</ElectionsUri>
22   <RegistrationUri>https://www.vote.virginia.gov/</RegistrationUri>
23   <RulesUri>http://www.sbe.virginia.gov/</RulesUri>
24   <WhatIsOnMyBallotUri>https://www.vote.virginia.gov/</WhatIsOnMyBallotUri>
25   <WhereDoIVoteUri>https://www.vote.virginia.gov/</WhereDoIVoteUri>
</ElectionAdministration>

```

Department

Tag	Data Type	Re-quired	Re-peats?	Description	Error Handling
Contact-Information	<i>ContactInformation</i>	Optional	Single	Contact and physical address information for the election administration body (see <i>ContactInformation</i>).	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 <i>election official</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
VoterService	multi-xml-voter-service	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
Contact-Information	<i>Contact-Information</i>	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	<i>InternationalizedText</i>	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	<code>xs:IDREF</code>	Optional	Single	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
Type	<i>VoterServiceType</i>	Optional	Single	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Other-Type	<code>xs:string</code>	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.

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. <i>See usage note.</i>	If the field is invalid or not present, then the implementation is required to ignore it.
Directions	<i>Internationalized-Text</i>	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]	<i>Internationalized-Text</i>	Optional	Single	Contains the hours (in local time) that the location is open (NB: <i>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).</i>	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	xs:ID	Optional	Single	References an <i>HoursOpen</i> 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	<i>LatLng</i>	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. <i>See usage note.</i>	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
```

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

ElectionNotice

Tag	Data Type	Required?	Repeats?	Description	Error Handling
No- tice- Text	<i>Inter- nation- alized- Text</i>	Re- quired	Single	The last minute or emergency notification text should be placed here.	If the element is invalid, then the implementation is required to ignore the ElectionNotice element containing it.
No- ticeUri	xs:anyURI	Optional	Single	Optional URL for additional information related to the last minute or emergency notification.	If the field is invalid or not present, then the implementation is required to ignore it.

VoterService

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Contact-Information	<i>Contact-Information</i>	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	<i>InternationalizedText</i>	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	<code>xs:IDREF</code>	Optional	Single	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
Type	<i>VoterServiceType</i>	Optional	Single	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Other-Type	<code>xs:string</code>	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.

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
External-identifiers	<i>External-identifiers</i>	Optional	Single	Other identifiers that link to external datasets (e.g. <i>OCD-IDs</i>)	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 <i>ElectoralDistrict</i> 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	<i>District-Type</i>	Required	Single	Specifies the type of electoral area.	If the field is invalid or not present, then the implementation is required to ignore the <i>ElectoralDistrict</i> object containing it.
Other-Type	xs:string	Optional	Single	Allows for cataloging a new <i>DistrictType</i> option when Type is specified as "other".	If the field is invalid or not present, then the implementation is required to ignore it.

```

1 <ElectoralDistrict id="ed60129">
2   <ExternalIdentifiers>
3     <ExternalIdentifier>
4       <Type>ocd-id</Type>
5       <Value>ocd-division/country:us/state:va</Value>
6     </ExternalIdentifier>
7     <ExternalIdentifier>
8       <Type>fips</Type>
9       <Value>51</Value>
10    </ExternalIdentifier>
11  </ExternalIdentifiers>
12  <Name>Commonwealth of Virginia</Name>
13  <Type>state</Type>
14 </ElectoralDistrict>

```

ExternalFile

The *ExternalFile* object holds a reference to a file external to the feed itself. External files are packaged along with the VIP feed into a single, archived file.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
FileUri	xs:anyURI	Required	Single	The URI of the external file.	If the field is invalid, then the implementation is required to ignore the ExternalFile element containing it.
Checksum	<i>Checksum</i>	Required	Single	The cryptographic checksum of the referenced external file.	If the element is invalid, then the implementation is required to ignore the ExternalFile element containing it.

```

1 <ExternalFile id="ef1">
2   <FileUri>precinct_shapes.zip</FileUri>
3   <Checksum>
4     <Algorithm>sha-256</Algorithm>
5     <Value>65b634c5037f8a344616020d8060d233daa37b0f032a71d0d15ad7a5d3afa68e</Value>
6   </Checksum>
7 </State>

```

Checksum

The Checksum object contains information about a cryptographic checksum, including the raw checksum value and the cryptographic hash algorithm used to compute it.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Algorithm	<i>ChecksumAlgorithm</i>	Required	Single	The cryptographic hash algorithm used to compute the checksum value.	If the field is invalid, then the implementation is required to ignore the Checksum element containing it.
Value	xs:string	Required	Single	The raw cryptographic checksum value encoded as a non-delimited, lowercase hexadecimal string.	If the field is invalid, then the implementation is required to ignore the Checksum element containing it.

```

1 <Checksum>
2   <Algorithm>sha-256</Algorithm>
3   <Value>65b634c5037f8a344616020d8060d233daa37b0f032a71d0d15ad7a5d3afa68e</Value>
4 </Checksum>

```

ExternalIdentifier

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Type	<i>IdentifierType</i>	Required	Single	Specifies the type of identifier. Must be one of the valid types as defined by <i>IdentifierType</i> .	If the field is invalid or not present, the implementation is required to ignore the <code>ElectionIdentifier</code> containing it.
Other-Type	xs:string	Optional	Single	Allows for cataloging an <code>ExternalIdentifier</code> type that falls outside the options listed in <i>IdentifierType</i> . 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 <code>ElectionIdentifier</code> containing it.

```

1 <ExternalIdentifiers>
2   <ExternalIdentifier>
3     <Type>ocd-id</Type>
4     <Value>ocd-division/country:us/state:nc/county:durham</Value>
5   </ExternalIdentifier>
6   <ExternalIdentifier>
7     <Type>FIPS</Type>
8     <Value>37063</Value>
9   </ExternalIdentifier>
10  <ExternalIdentifier>
11    <Type>OTHER</Type>
12    <OtherType>GNIS</OtherType>
13    <Value>1008550</Value>
14  </ExternalIdentifier>
15  <external_identifier>
16    <Type>OTHER</Type>
17    <OtherType>census</OtherType>
18    <Value>99063</Value>
19  </external_identifier>
20 </ExternalIdentifiers>

```

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	<i>ExternalIdentifier</i>	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.

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	<i>Schedule</i>	Required	Repeats	Defines a block of days and hours that a place will be open.	At least one valid <i>Schedule</i> must be present for HoursOpen to be valid. If no valid <i>Schedule</i> 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	<i>Hours</i>	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.
IsOnly-ByAppointment	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.
IsOr-ByAppointment	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.
IsSubject-ToChange	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.
Start-Date	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.
End-Date	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
Start-Time	<i>Time-With-Zone</i>	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.
End-Time	<i>Time-With-Zone</i>	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

```

1 <HoursOpen id="hours0001">
2   <Schedule>
3     <Hours>
4       <StartTime>06:00:00-05:00</StartTime>

```

(continues on next page)

(continued from previous page)

```

5      <EndTime>12:00:00-05:00</EndTime>
6    </Hours>
7    <Hours>
8      <StartTime>13:00:00-05:00</StartTime>
9      <EndTime>19:00:00-05:00</EndTime>
10   </Hours>
11   <StartDate>2013-11-05</StartDate>
12   <EndDate>2013-11-05</EndDate>
13 </Schedule>
14 </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* NOTE: Internationalized Text is not currently supported for CSV submissions.

Tag	Data Type	Required	Repeats?	Description	Error Handling
Text	<i>LanguageString</i>	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.

```

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

```

Locality

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Election-AdministrationId	xs:IDREF	Optional	Single	Links to the locality's <i>ElectionAdministration</i> object.	If the field is invalid or not present, then the implementation is required to ignore it.
ExternalIdentifiers	<i>ExternalIdentifiers</i>	Optional	Single	Another identifier for a locality that links to another dataset (e.g. <i>OCD-ID</i>)	If the element is invalid or not present, then the implementation is required to ignore it.
IsMailOnly	xs:boolean	Optional	Single	Determines if the locality runs mail-only elections. If this is true, then all precincts a part of the locality will also run mail-only elections. Drop boxes may be used in addition to this flag using a <i>polling location</i> record configured as a Drop Box.	If the field is missing or invalid, the implementation is required to assume <i>IsMailOnly</i> is false.
Name	xs:string	Required	Single	Specifies the name of a locality.	If the field is invalid, then the implementation is required to ignore the <i>Locality</i> element containing it.
PollingLocations	xs:IDREFS	Optional	Single	Specifies a link to a set of the locality's <i>ref:polling locations</i> <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 <i>State</i> .	If the field is invalid, then the implementation is required to ignore the <i>Locality</i> element containing it.
Type	<i>DistrictType</i>	Optional	Single	Defines the kind of locality (e.g. county, town, et al.), which is one of the various <i>DistrictType enumerations</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Other-Type	xs:string	Optional	Single	Allows for defining a type of locality that falls outside the options listed in <i>DistrictType</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

```

1 <Locality id="loc70001">
2   <ElectionAdministrationId>ea40001</ElectionAdministrationId>
3   <ExternalIdentifiers>
4     <ExternalIdentifier>
5       <Type>ocd-id</Type>
6       <Value>ocd-division/country:us/state:va/county:albemarle</Value>
7     </ExternalIdentifier>
8   </ExternalIdentifiers>
9   <IsMailOnly>true</IsMailOnly>
10  <Name>ALBEMARLE COUNTY</Name>
11  <StateId>st51</StateId>
12  <Type>county</Type>
13 </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	<i>ContactInformation</i>	Optional	Repeats	Links to the <i>ContactInformation</i> element associated with the office.	If the element is invalid or not present, then the implementation is required to ignore it.
Description	<i>Internationalized-Text</i>	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	<code>xs:IDREF</code>	Required	Single	Links to the <i>ElectoralDistrict</i> element associated with the office.	If the field is invalid or not present, the implementation is required to ignore the <i>Office</i> element containing it.
ExternalIdentifiers	<i>ExternalIdentifiers</i>	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	<code>xs:date</code>	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	<code>xs:boolean</code>	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	<i>Internationalized-Text</i>	Required	Single	The name of the office.	If the field is invalid or not present, the implementation is required to ignore the <i>Office</i> element containing it.
OfficeHolderPersonIds	<code>xs:IDREF</code>	Optional	Single	Links to the <i>Person</i> 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	<i>Term</i>	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	<i>OfficeTermType</i>	Optional	Single	Specifies the type of office term (see <i>OfficeTermType</i> for valid values).	If the field is invalid or not present, the implementation is required to ignore the <i>Office</i> element containing it.
Start-Date	<code>xs:date</code>	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.
End-Date	<code>xs:date</code>	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.


```
1 <Office id="off0000">
2   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
3   <FilingDeadline>2013-01-01</FilingDeadline>
4   <IsPartisan>>false</IsPartisan>
5   <Name>
6     <Text language="en">Governor</Text>
7   </Name>
8   <Term>
9     <Type>full-term</Type>
10  </Term>
11 </Office>
```

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. <i>See usage note.</i>	If the field is invalid or not present, then the implementation is required to ignore it.
Directions	<i>Internationalized-Text</i>	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]	<i>Internationalized-Text</i>	Optional	Single	Contains the hours (in local time) that the location is open (NB: <i>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.</i>)	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	xs:ID	Optional	Single	References an <i>HoursOpen</i> 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	<i>LatLng</i>	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. <i>See usage note.</i>	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.
102				Chapter 2. XML Documentation	

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
```

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

OrderedContest

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

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

```

1 <OrderedContest id="oc20003abc">
2   <ContestId>cc20003</ContestId>
3   <OrderedBallotSelectionIds>cs10961 cs10962 cs10963</OrderedBallotSelectionIds>
4 </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
Ab- bre- via- tion	<code>xs:string</code>	Op- tional	Single	An abbreviation for the party name.	If the field is invalid or not present, then the implementation is required to ignore it.
Color	<i>Html- Col- orString</i>	Op- tional	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.
Ex- ter- nal- iden- ti- fiers	<i>Ex- ter- nal- iden- ti- fiers</i>	Op- tional	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	<code>xs:boolean</code>	Op- tional	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.
Lead- er- Per- son- Ids	<code>xs:IDREFS</code>	Op- tional	Single	A reference of <i>Person</i> elements which are leaders of the <i>Party</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
Lo- goUri	<code>xs:anyURI</code>	Op- tional	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	<i>In- ter- na- tion- al- ized- Text</i>	Op- tional	Single	The name of the party.	If the element is invalid or not present, then the implementation is required to ignore it.

```

1 <Party id="par0001">
2   <Abbreviation>REP</Abbreviation>
3   <Color>e91d0e</Color>
4   <IsWriteIn>>false</IsWriteIn>
5   <LeaderPersonIds>per01</LeaderPersonIds>
6   <Name>
7     <Text language="en">Republican</Text>
8   </Name>
9 </Party>

```

HtmlColorString

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

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

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).

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
Ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Se- lec- tion- Ids	xs:IDREFS	Optional	Single	References a set of <code>BallotSelections</code> , which could be of any selection type that extends <code>BallotSelectionBase</code> .	If the field is invalid or not present, then the implementation should ignore it.
Bal- lot- Sub- Ti- tle	<i>Internationalized-Text</i>	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.
Bal- lot- Ti- tle	<i>Internationalized-Text</i>	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.
Elec- toral- Dis- tric- tId	xs:IDREF	Required	Single	References an <code>ElectoralDistrict</code> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the <code>ContestBase</code> element containing it.
Elec- torate- Spec- ifi- ca- tion	<i>Internationalized-Text</i>	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.
Ex- ter- nal- den- ti- fiers	<i>External-identifiers</i>	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.
Has- Ro- ta- tion	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	Required	Single	Name of the contest, not necessarily how it appears on the ballot (NB: <code>BallotTitle</code> should be used for this purpose).	If the field is invalid, then the implementation is required to ignore the

2.1. XML Specification

PartySelection

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

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
PartyIds	xs:IDREFS	Re- quired	Single	One or more <i>Party</i> 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.

BallotSelectionBase

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

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
Sequence- Order	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 <i>OrderedBallotSlectionIds</i> in <i>Ordered-Contest</i> .	If the field is invalid or not present, then the implementation is required to ignore 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
Contact-Information	<i>Contact-Information</i>	Optional	Repeats	Refers to the associated <i>ContactInformation</i> .	If the element is invalid or not present, then the implementation is required to ignore it.
Date-Of-Birth	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.
External-Identifiers	<i>External-Identifiers</i>	Optional	Single	Identifiers for this person.	If the element is invalid or not present, then the implementation is required to ignore it.
First-Name	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.
Full-Name	<i>Internationalized-Text</i>	Optional	Single	Specifies a person's full name (NB: this information is <i>InternationalizedText</i> 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.
Last-Name	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.
Middle-Name	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 <i>Party</i> . 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 <i>Candidate PartyId</i> .	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	<i>Internationalized-Text</i>	Optional	Single	Specifies a person's profession (NB: this information is <i>InternationalizedText</i> 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.

```
1 <Person id="per50001">
2   <ContactInformation label="ci60002">
3     <Email>rwashburne@albemarle.org</Email>
4     <Phone>4349724173</Phone>
5   </ContactInformation>
6   <FirstName>RICHARD</FirstName>
7   <LastName>WASHBURN</LastName>
8   <MiddleName>J.</MiddleName>
9   <Nickname>JAKE</Nickname>
10  <Title>
11    <Text language="en">General Registrar Physical</Text>
12  </Title>
13 </Person>
```

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	<code>xs:string</code>	Optional	Repeats	The “location” portion of a mailing address. <i>See usage note.</i>	If the field is invalid or not present, then the implementation is required to ignore it.
Directions	<i>Internationalized-Text</i>	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	<code>xs:string</code>	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	<code>xs:string</code>	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]	<i>Internationalized-Text</i>	Optional	Single	Contains the hours (in local time) that the location is open (NB: <i>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.</i>)	If the element is invalid or not present, then the implementation is required to ignore it.
HoursOpenId	<code>xs:ID</code>	Optional	Single	References an <i>HoursOpen</i> 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	<i>LatLng</i>	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	<code>xs:string</code>	Optional	Single	The name of the location or contact. <i>See usage note.</i>	If the field is invalid or not present, then the implementation is required to ignore it.
Phone	<code>xs:string</code>	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	<code>xs:anyURI</code>	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.
2.1. XML Specification					111

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
```

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

PollingLocation

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
AddressStructured	<i>Sim- Struct- AddressType</i>	Optional	Single	Represents the various structured parts of an address to a polling location.	One of AddressStructured and AddressLine should be present for a given Polling Location. If none is present, the implementation is required to ignore the <code>PollingLocation</code> element containing it.
AddressLine	<code>xs:string</code>	Optional	Repeats	Represents the various parts of an address to a polling location.	One of <code>AddressStructured</code> and <code>AddressLine</code> should be present for a given Polling Location. If none is present, the implementation is required to ignore the <code>PollingLocation</code> element containing it.
Directions	<i>Inter- national- alized- Text</i>	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 [depre- cated]	<i>Inter- national- alized- Text</i>	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 <i>HoursOpen</i> 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	<code>xs:IDREF</code>	Optional	Single	Links to an <i>HoursOpen</i> 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	<code>xs:boolean</code>	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	<code>xs:boolean</code>	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	<i>LatLng</i>	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	<code>xs:string</code>	Optional	Single	Name of the polling location.	If the field is invalid or not present, then the implementation is required to ignore it.
PhotoUri	<code>xs:anyURI</code>	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.

```

1 <PollingLocation id="pl00000">
2   <AddressLine>2775 Hydraulic Rd Charlottesville, VA 22901</AddressLine>
3   <HoursOpenId>hours0002</HoursOpenId>

```

(continues on next page)

(continued from previous page)

```

4    <IsDropBox>true</IsDropBox>
5    <IsEarlyVoting>true</IsEarlyVoting>
6    <LatLng>
7        <Latitude>38.009939</Latitude>
8        <Longitude>-78.506204</Longitude>
9    </LatLng>
10   <Name>ALBERMARLE HIGH SCHOOL</Name>
11 </PollingLocation>
12 <!-- Or: -->
13 <PollingLocation id="p100000">
14     <AddressStructured>
15         <Line1>2775 Hydraulic Rd</Line1>
16         <City>CHARLOTTESVILLE</City>
17         <State>VA</State>
18         <Zip>22901</Zip>
19     </AddressStructured>
20     <HoursOpenId>hours0002</HoursOpenId>
21     <IsDropBox>true</IsDropBox>
22     <IsEarlyVoting>true</IsEarlyVoting>
23     <LatLng>
24         <Latitude>38.009939</Latitude>
25         <Longitude>-78.506204</Longitude>
26     </LatLng>
27     <Name>ALBERMARLE HIGH SCHOOL</Name>
28 </PollingLocation>

```

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:double	Required	Single	The latitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
Longitude	xs:double	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.

SimpleAddressType

A SimpleAddressType represents a structured address.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Line1	xs:string	Required	Single	The address line for a structured address. Should include the street number, street name, and any prefix and suffix.	If no Line1 is provided, the implementation should ignore the SimpleAddressType.
Line2	xs:string	Optional	Single	Additional field for an address	If no Line2 is provided, the implementation should ignore it.
Line3	xs:string	Optional	Single	Additional field for an address	If no Line3 is provided, the implementation should ignore it.
City	xs:string	Required	Single	The City value of a structured address.	If City is not provided, the implementation should ignore the SimpleAddressType.
State	xs:string	Required	Single	The State value of a structured address.	If State is not provided, the implementation should ignore the SimpleAddressType.
Zip	xs:string	Optional	Single	The ZIP code of a structured address.	If Zip is not provided, the implementation should ignore the SimpleAddressType.

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).

Voters can be assigned to a precinct in two ways. A voter location modeled by *StreetSegment* is assigned to a precinct by *StreetSegment.PrecinctId*. Alternatively, a precinct's spatial boundary can be modeled with *Precinct.SpatialBoundary*. Any registered voter address contained within the spatial boundary of the precinct is assigned to that precinct.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
Ballot-StyleId	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:IDREF	Optional	Single	Links to the 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.
Is-MailOnly	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 <i>IsMailOnly</i> is false.
LocalityId	xs:IDREF	Required	Single	Links to the Locality that comprises the precinct.	If the field is invalid, then the implementation is required to ignore the <code>Precinct</code> element containing it.
Name	xs:string	Required	Single	Specifies the precinct's name (or number if no name exists).	If the field is invalid, then the implementation is required to ignore the <code>Precinct</code> 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 <i>Name</i> 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:IDREF	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 <i>Precinct</i> object represents a full precinct. If this field is present, then this <i>Precinct</i> object represents one portion of a split precinct. Each <i>Precinct</i> object that represents one portion of a split precinct must have the same <i>Name</i> value, but different <i>PrecinctSplitName</i> values. See the <i>sample_feed.xml</i> file for examples.	If the field is invalid or not present, then the implementation is required to ignore it.
SpatialBoundary	SpatialBoundary	Optional	Single	Defines the spatial boundary of the precinct. All voter addresses contained within this boundary are assigned to the precinct. If a voter address also maps to a StreetSegment , then the precinct assignment from the <i>StreetSegment</i> will be preferred over the assignment from the spatial boundary.	If the element 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 in-


```

1 <Precinct id="pre90111">
2   <BallotStyleId>bs00010</BallotStyleId>
3   <ElectoralDistrictIds>ed60129 ed60311 ed60054</ElectoralDistrictIds>
4   <IsMailOnly>>false</IsMailOnly>
5   <LocalityId>loc70001</LocalityId>
6   <Name>203 - GEORGETOWN</Name>
7   <Number>0203</Number>
8   <PollingLocationIds>p181274</PollingLocationIds>
9   <SpatialBoundary>
10    <ExternalGeospatialFeature>
11     <ExternalFileId>ef1</ExternalFileId>
12     <FileFormat>shp</FileFormat>
13     <FeatureIdentifier>
14      <Index>3</Index>
15     </FeatureIdentifier>
16    </ExternalGeospatialFeature>
17   </SpatialBoundary>
18 </Precinct>
19 <!--
20   Precinct split. Name and PollingLocationIds are the same but
21   PrecinctSplitName is present, the ElectoralDistrictIds are different,
22   and the BallotStyleId is different.
23 -->
24 <Precinct id="pre90348sp0000">
25   <BallotStyleId>bs00002</BallotStyleId>
26   <ElectoralDistrictIds>ed60129 ed60054 ed60150</ElectoralDistrictIds>
27   <IsMailOnly>>false</IsMailOnly>
28   <LocalityId>loc70001</LocalityId>
29   <Name>201 - JACK JOUETT</Name>
30   <Number>0201</Number>
31   <PollingLocationIds>p100000 p181273 p181662</PollingLocationIds>
32   <PrecinctSplitName>0000</PrecinctSplitName>
33 </Precinct>
34 <Precinct id="pre90348sp0001">
35   <BallotStyleId>bs00015</BallotStyleId>
36   <ElectoralDistrictIds>ed60129 ed60054 ed60267</ElectoralDistrictIds>
37   <IsMailOnly>>false</IsMailOnly>
38   <LocalityId>loc70001</LocalityId>
39   <Name>201 - JACK JOUETT</Name>
40   <Number>0201</Number>
41   <PollingLocationIds>p100000 p181273 p181662</PollingLocationIds>
42   <PrecinctSplitName>0001</PrecinctSplitName>
43 </Precinct>

```

SpatialBoundary

The `SpatialBoundary` object defines a boundary in space. This boundary is usually defined by one or more discrete, closed polygonal shapes.

Tag	Data Type	Required	Repeats?	Description	Error Handling
External-GeospatialFeature	<i>External-GeospatialFeature</i>	Required	Single	The spatial boundary defined by a geospatial feature that is external to the VIP feed.	If the element is invalid, then the implementation is required to ignore the <code>SpatialBoundary</code> element containing it.

```

1 <SpatialBoundary>
2   <ExternalGeospatialFeature>
3     <ExternalFileId>ef1</ExternalFileId>
4     <FileFormat>shp</FileFormat>
5     <FeatureIdentifier>
6       <Index>3</Index>
7     </FeatureIdentifier>
8   </ExternalGeospatialFeature>
9 </SpatialBoundary>

```

ExternalGeospatialFeature

The `ExternalGeospatialFeature` object contains a reference to a geospatial feature (one or more shapes) contained in a separate file external to the VIP feed.

Tag	Data Type	Re-quired?	Re-peats?	Description	Error Handling
External-FileId	<code>xs:string</code>	Re-quired	Single	Links to the <i>ExternalFile</i> containing the geospatial shape(s) that define the feature's boundary.	If the field is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.
File-Format	<i>Geospatial-Format</i>	Re-quired	Single	The format of the geospatial file.	If the field is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.
FeatureIdentifier	<i>FeatureIdentifier</i>	Re-quired	Re-peats	Identifiers indicating which specific shape(s) to use from the geospatial file. These refer to identifiers within the referenced external file. This is a repeated field in the XML specification, but a scalar field in the CSV specification. If more than one identifier is required with the CSV specification, multiple values can be provided by delimited by space.	If the element is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.

FeatureIdentifier

Tag	Data Type	Re-quired?	Re-peats?	Description	Error Handling
Index	<code>xs:int</code>	Optional	Single	The index value for the shapefile feature.	If the field is invalid or not present, then the implementation is required to ignore it.

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 <i>Candidate</i> 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.

```

1 <RetentionContest id="rc40001">
2   <BallotSelectionIds>rc40001a rc40001b</BallotSelectionIds>
3   <BallotTitle>
4     <Text language="en">Retention of Supreme Court Justice</Text>
5     <Text language="es">La retención de juez de la Corte Suprema</Text>
6   </BallotTitle>
7   <ElectoralDistrictId>ed60129</ElectoralDistrictId>
8   <Name>Judicial Retention, Supreme Court</Name>
9   <CandidateId>can14444</CandidateId>
10  <OfficeId>off20006</OfficeId>
11 </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 <code>Source</code> 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 <code>Source</code> element containing it.
Date-Time	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 the <code>Source</code> element containing it.
Description	<i>Internationalized-Text</i>	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.
Feed-ContactInformation	<i>ContactInformation</i>	Optional	Single	Reference to the Person who will respond to inquiries about the information contained within the file.	If the element 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 the <code>Source</code> element containing it.

```

1 <Source id="src1">
2   <DateTime>2013-10-24T14:25:28</DateTime>
3   <Description>
4     <Text language="en">SBE is the official source for Virginia data</Text>
5   </Description>
6   <Name>State Board of Elections, Commonwealth of Virginia</Name>
7   <OrganizationUri>http://www.sbe.virginia.gov/</OrganizationUri>
8   <VipId>51</VipId>
9   <Version>5.0</Version>
10 </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
Election-AdministrationId	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	<i>ExternalIdentifiers</i>	Optional	Single	Other identifier for the state that relates to another dataset (e.g. <i>OCD-ID</i>).	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 state, such as Alabama.	If the field is invalid, then the implementation is required to ignore the <i>State</i> element containing it.
PollingLocations	xs:IDREFS	Optional	Single	Specifies a link to the state's <i>polling locations</i> . 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 <i>Precinct</i> element.	If the field is invalid or not present, then the implementation is required to ignore it.

```

1 <State id="st51">
2   <ElectionAdministrationId>ea40133</ElectionAdministrationId>
3   <ExternalIdentifiers>
4     <ExternalIdentifier>
5       <Type>ocd-id</Type>
6       <Value>ocd-division/country:us/state:va</Value>
7     </ExternalIdentifier>
8   </ExternalIdentifiers>
9   <Name>Virginia</Name>
10 </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
Address-Direction	<code>xs:string</code>	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	<code>xs:string</code>	Required	Single	The city specifies the city or town of the address.	If the field is invalid, then the implementation is required to ignore the <code>StreetSegment</code> element containing it.
IncludesAllAddresses	<code>xs:boolean</code>	Optional	Single	Specifies if the segment covers every address on this street. If this is <i>true</i> , then the values of StartHouseNumber and EndHouseNumber should be ignored. The value of OddEvenBoth must be <i>both</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
IncludesAllStreets	<code>xs:boolean</code>	Optional	Single	Specifies if the segment covers every street in this city. If this is <i>true</i> , 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.
Odd-Even-Both	<i>OddEvenBoth</i>	Optional	Single	Specifies whether the odd side of the street (in terms of house numbers), the even side, or both are included in the street segment.	If the field is not present or invalid, the implementation is required to ignore the <code>StreetSegment</code> containing it.
PrecinctId	<code>xs:string</code>	Required	Single	References the <i>Precinct</i> that contains the entire street segment. If a precinct has a <i>SpatialBoundary</i> which also contains the entire street segment, then the precinct assignment from the segment will be preferred over the assignment defined by the spatial boundary.	If the field is invalid, then the implementation is required to ignore the <code>StreetSegment</code> element containing it.
StartHouseNumber	<code>xs:integer</code>	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 <code>StreetSegment</code> element containing it. If the StartHouseNumber is greater than the EndHouseNumber , the implementation should ignore the element containing them.
EndHouseNumber	<code>xs:integer</code>	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 <code>StreetSegment</code> element containing it. If the EndHouseNumber is less than the StartHouseNumber , the implementation should ignore the element containing it.
HouseNumber-Prefix	<code>xs:string</code>	Optional	Single	Part of a street address. It may contain letters or slashes (e.g., ‘B’ in ‘B22 Main St’). If this value is present then StartHouseNumber must be equal to EndHouseNumber . This field cannot be used if IncludesAllAddresses or IncludesAllStreets are true.	If the field is invalid or not present, then the implementation is required to ignore it.
HouseNumber-Suffix	<code>xs:string</code>	Optional	Single	Part of a street address. It may contain letters or slashes (e.g., 1/2 in ‘22 1/2 Main St’). If this value is present then StartHouseNumber must be equal to EndHouseNumber .	If the field is invalid or not present, then the implementation is required to ignore it.

```

1 <StreetSegment id="ss999999">
2   <City>Charlottesville</City>
3   <IncludesAllAddresses>true</IncludesAllAddresses>
4   <OddEvenBoth>both</OddEvenBoth>
5   <PrecinctId>pre99999</PrecinctId>
6   <State>VA</State>
7   <StreetName>CHAPEL HILL</StreetName>
8   <StreetSuffix>RD</StreetSuffix>
9   <Zip>22901</Zip>
10 </StreetSegment>
11 <StreetSegment id="ss309904">
12   <City>GREENWOOD</City>
13   <OddEvenBoth>both</OddEvenBoth>
14   <PrecinctId>pre92145</PrecinctId>
15   <StartHouseNumber>1</StartHouseNumber>
16   <EndHouseNumber>201</EndHouseNumber>
17   <State>VA</State>
18   <StreetName>MISTY MOUNTAIN</StreetName>
19   <StreetSuffix>RD</StreetSuffix>
20   <Zip>22943</Zip>
21 </StreetSegment>
22 <StreetSegment id = "ss1"
23   <City>GREENWOOD</City>
24   <OddEvenBoth>both</OddEvenBoth>
25   <PrecinctId>pre92145</PrecinctId>
26   <StartHouseNumber>1</StartHouseNumber>
27   <EndHouseNumber>1</EndHouseNumber>
28   <HouseNumberPrefix>B</HouseNumberPrefix>
29   <HouseNumberSuffix>1/2</HouseNumberSuffix>
30   <State>VA</State>
31   <StreetName>MISTY MOUNTAIN</StreetName>
32   <StreetSuffix>RD</StreetSuffix>
33   <Zip>22943</Zip>
34 </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	

ChecksumAlgorithm

Tag	Description
sha-256	256-bit cryptographic hash algorithm of the SHA-2 family
sha-512	512-bit cryptographic hash algorithm of the SHA-2 family

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 <i>OtherType</i> field to describe it.

GeospatialFormat

Geospatial file formats that are supported by the VIP specification.

Tag	Description
shp	ESRI Shapefile (reference)

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.
ap-proval	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).
super-majority	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 department handles the selection and management of polling places.
voter-registration	The department that manages voter registration.
other	Any other service not covered by the above descriptions.

3.1 CSV Specification

- *Getting Started*
- *Elements & Enumerations*
 - *Single-page Format*
 - *Elements (Separate Pages)*
 - *Enumerations (Separate Pages)*

3.1.1 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
- department.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).

3.1.2 Elements & Enumerations

Single-page Format

CSV Elements & Enumerations (Single Page)

- *Elements*
 - *ballot_measure_contest*
 - * *contest_base*
 - *ballot_measure_selection*
 - * *ballot_selection_base*
 - *ballot_selection_base*
 - *ballot_style*
 - *candidate_selection*
 - * *ballot_selection_base*
 - *candidate*
 - *candidate_contest*

- * *contest_base*
- *checksum*
- *contact_information*
- *contest_base*
- *department*
 - * *voter_service*
 - * *contact_information*
- *election*
- *election_administration*
 - * *department*
 - *voter_service*
 - *contact_information*
 - * *election_notice*
 - * *voter_service*
- *election_notice*
- *electoral_district*
- *external_file*
 - * *checksum*
- *external_geospatial_feature*
 - * *feature_identifier*
- *external_identifier*
- *external_identifiers*
- *feature_identifier*
- *hours*
 - * *time_with_zone*
- *hours_open*
 - * *schedule*
 - *hours*
- *time_with_zone*
- *html_color_string*
- *internationalized_text*
- *language_string*
- *lat_long*
- *locality*
- *office*

- * *term*
 - * *contact_information*
 - *ordered_contest*
 - *party*
 - * *html_color_string*
 - *party_contest*
 - * *contest_base*
 - *party_selection*
 - * *ballot_selection_base*
 - *person*
 - * *contact_information*
 - *polling_location*
 - * *lat_long*
 - * *simple_address_type*
 - *precinct*
 - * *spatial_boundary*
 - *external_geospatial_feature*
 - * *feature_identifier*
 - *retention_contest*
 - *schedule*
 - * *hours*
 - *time_with_zone*
 - *simple_address_type*
 - *source*
 - *spatial_boundary*
 - * *external_geospatial_feature*
 - *feature_identifier*
 - *state*
 - *street_segment*
 - *term*
 - *time_with_zone*
 - *voter_service*
- *Enumerations*
 - *ballot_measure_type*
 - *candidate_post_election_status*
 - *candidate_pre_election_status*

- *checksum_algorithm*
- *district_type*
- *geospatial_format*
- *identifier_type*
- *oeb_enum*
- *office_term_type*
- *vote_variation*
- *voter_service_type*

Elements

ballot_measure_contest

The BallotMeasureContest provides information about a ballot measure before the voters, including summary statements on each side. Extends single-csv-contest-base.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
con_statement	xs:string	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.
effect_of_abstain	xs:string	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.
full_text	xs:string	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.
info_uri	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.
passage_threshold	xs:string	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.
pro_statement	xs:string	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.
summary_text	xs:string	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	xs:string	Optional	Single	Specifies the particular type of ballot measure. Must be one of the valid <i>ballot_measure_type</i> options.	If the field is invalid or not present, then the implementation is required to ignore it.
other_type	xs:string	Optional	Single	Allows for cataloging a new <i>ballot_measure_type</i> option, when Type is specified as “other.”	If the field is invalid or not present, then the implementation is required to ignore it.
132				Chapter 3. CSV Documentation	


```

1  id,abbreviation,ballot_selection_ids,ballot_sub_title,ballot_title,elecral_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
2  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,

```

contest_base

A base model for all Contest types: *ballot_measure_contest*, *candidate_contest*, *party_contest*, and *retention_contest* (NB: the latter because it extends *ballot_measure_contest*).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_selection_ids	xs:IDRefs	Optional	Single	References a set of BallotSelections, which could be of any selection type that extends single-csv-ballot-selection-base.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_sub_title	xs:string	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.
bal- lot_title	xs:string	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.
elec- toral_district_id	xs:IDRef	Required	Single	References an <i>electoral_district</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the ContestBase element containing it.
elec- torate_specification	xs:string	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.
ex- ter- nal_identifiers	xs:string	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.
has_rotations	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	Required	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, then the implementation is required to ignore the ContestBase element containing it.
sequence_order	xs:integer	Optional	Single	Order in which the contests are listed on the ballot. This is the default ordering, and can be overrides by data in a <i>ballot_style</i> element.	If the field is invalid or not present, then the implementation should ignore it.

ballot_measure_selection

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. BallotMeasureSelection extends single-csv-ballot-selection-base.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
selection	xs:string	Required	Single	Selection text for a <i>ballot_measure_contest</i>	If the element is invalid or not present, the implementation is required to ignore the BallotMeasureSelection containing it.

```

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

```

ballot_selection_base

A base model for all ballot selection types: *ballot_measure_selection*, *candidate_selection*, and *party_selection*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
sequence_order	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 <i>OrderedBallotSlectionIds</i> in <i>ordered_contest</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

ballot_selection_base

A base model for all ballot selection types: *ballot_measure_selection*, *candidate_selection*, and *party_selection*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
sequence_order	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 <i>OrderedBallotSlectionIds</i> in <i>ordered_contest</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

ballot_style

A container for the contests/measures on the ballot.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
image_uri	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.
ordered_contest_ids	xs:IDREFS	Optional	Single	Reference to a set of <i>ordered_contest</i>	If the field is invalid or not present, then the implementation is required to ignore it.
party_ids	xs:IDREFS	Optional	Single	Reference to a set of <i>single-csv-party</i> 's.	If the field is invalid or not present, then the implementation is required to ignore it.

```

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

```

candidate_selection

CandidateSelection extends single-csv-ballot-selection-base and represents a ballot selection for a candidate contest.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
candidate_ids	xs:IDREFS	Optional	Single	References a set of <i>candidate</i> 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.
endorsement_party_ids	xs:IDREFS	Optional	Single	References a set of <i>party</i> 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.
is_write_in	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.

```

1 id,sequence_order,candidate_ids,endorsement_party_ids,is_write_in
2 cs001,3,can004,par01,false
3 cs002,2,can001 can002,par03 par02,false
4 cs003,1,can003,par02 par03,true

```

ballot_selection_base

A base model for all ballot selection types: *ballot_measure_selection*, *candidate_selection*, and *party_selection*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
sequence_order	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 <i>OrderedBallotSlectionIds</i> in <i>ordered_contest</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

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
ballot_name	xs:string	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, then the implementation is required to ignore the Candidate element containing it.
contact_information	xs:string	Optional	Single	Contact and physical address information for this Candidate and/or their campaign (see single-csv-contact-information).	If the element is invalid or not present, then the implementation is required to ignore it.
external_identifiers	<i>external_identifier</i>	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.
file_date	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.
is_incumbent	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.
is_top_ticket	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.
party_id	xs:IDREF	Optional	Single	Reference to a <i>party</i> 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.
person_id	xs:IDREF	Optional	Single	Reference to a <i>person</i> element with additional information about the candidate.	If the field is invalid or not present, then the implementation is required to ignore it.
post_election_date	xs:date	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.
pre_election_date	xs:date	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.

```

1 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
2 can001,Jude Fawley,,,,,2016-12-01,true,false,par01,per50001,,filed
3 can002,Arabella Donn,,,,,2016-12-01,false,false,par02,per50002,,qualified
4 can003,John Coltrane,,,,,2016-09-23,false,false,par02,per50003,,qualified
5 can004,Miles Davis,,,,,2016-05-26,false,false,par01,per50004,,qualified

```

candidate_contest

CandidateContest extends single-csv-contest-base and represents a contest among candidates.

Tag	Data Type	Required?	Re-usable?	Description	Error Handling
number_elected	xs:int	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.
office_ids	xs:IDREFS	Optional	Single	References a set of <i>office</i> 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 <i>BallotSelectionIds</i> . E.g., if the various <i>BallotSelectionIds</i> reference <i>candidate_selection</i> elements which reference the candidate for President first and Vice-President second, the <i>OfficeIds</i> 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.
primary_party_ids	xs:IDREFS	Optional	Single	References <i>party</i> 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.
votes_allowed	xs:int	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.

```

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,
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

```

contest_base

A base model for all Contest types: *ballot_measure_contest*, *candidate_contest*, *party_contest*, and *retention_contest* (NB: the latter because it extends *ballot_measure_contest*).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_selection_ids	xs:IDRefs	Optional	Single	References a set of BallotSelections, which could be of any selection type that extends single-csv-ballot-selection-base.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_sub_title	xs:string	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.
bal- lot_title	xs:string	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.
elec- toral_district_id	xs:IDRef	Required	Single	References an <i>electoral_district</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the ContestBase element containing it.
elec- torate_specification	xs:string	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.
ex- ter- nal_identifiers	xs:string	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.
has_rotations	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	Required	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, then the implementation is required to ignore the ContestBase element containing it.
sequence_order	xs:integer	Optional	Single	Order in which the contests are listed on the ballot. This is the default ordering, and can be overridden by data in a <i>ballot_style</i> element.	If the field is invalid or not present, then the implementation should ignore it.

checksum

The `Checksum` object contains information about a cryptographic checksum, including the raw checksum value and the cryptographic hash algorithm used to compute it.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
algorithm	<i>checksum_algorithm</i>	Required	Single	The cryptographic hash algorithm used to compute the checksum value.	If the field is invalid, then the implementation is required to ignore the <code>Checksum</code> element containing it.
value	<code>xs:string</code>	Required	Single	The raw cryptographic checksum value encoded as a non-delimited, lowercase hexadecimal string.	If the field is invalid, then the implementation is required to ignore the <code>Checksum</code> element containing it.

```
1 id,algorithm,value
2 chl,sha-256,65b634c5037f8a344616020d8060d233daa37b0f032a71d0d15ad7a5d3afa68e
```

contact_information

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*). `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
address_line	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	xs:string	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	xs:string	Optional	Single	Contains the hours (in local time) that the location is open (<i>NB: this element is deprecated in favor of the more structured :ref: ‘single-csv-hours-open’ element. It is strongly encouraged that data providers move toward contributing hours in this format.</i>)	If the element is invalid or not present, then the implementation is required to ignore it.
hours_open	xs:string	Optional	Single	References an hours_open 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.
lat_long_csv_lat_lng	xs:string	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.

```
1 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  
2 ci0827,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,  
  ↳,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,off001  
3 ci0828,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,  
  ↳,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,vs01
```

contest_base

A base model for all Contest types: *ballot_measure_contest*, *candidate_contest*, *party_contest*, and *retention_contest* (NB: the latter because it extends *ballot_measure_contest*).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_selection_id	xs:IDREF	Optional	Single	References a set of BallotSelections, which could be of any selection type that extends single-csv-ballot-selection-base.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_sub_title	xs:string	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.
bal- lot_title	xs:string	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.
elec- toral_district_id	xs:IDREF	Required	Single	References an <i>electoral_district</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the ContestBase element containing it.
elec- torate_specification	xs:string	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.
ex- ter- nal_identifiers	xs:string	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.
has_rotations	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	Required	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, then the implementation is required to ignore the ContestBase element containing it.
sequence_order	xs:integer	Optional	Single	Order in which the contests are listed on the ballot. This is the default ordering, and can be overridden by data in a <i>ballot_style</i> element.	If the field is invalid or not present, then the implementation should ignore it.

department

Tag	Data Type	Required?	Repeats?	Description	Error Handling
contact_information	single-contact-information	Optional	Single	Contact and physical address information for the election administration body (see single-csv-contact-information).	If the element is invalid or not present, then the implementation is required to ignore it.
election_official_person_id	xs:IDREF	Optional	Single	The individual to contact at the election administration office. The specified person should be the <i>election official</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
voter_services	single-csv-voter-service	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.

```

1 id,election_official_person_id,election_administration_id
2 dep01,per50002,ea123
3 dep02,per50002,ea345
4 dep03,per50002,ea625
5 dep04,per50002,ea625

```

voter_service

Tag	Data Type	Required?	Repeats?	Description	Error Handling
contact_information	single-csv-contact-information	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	xs:string	Optional	Single	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
election_official_person_id	xs:IDREF	Optional	Single	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
type	<i>voter_service</i>	Optional	Single	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
other_type	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.

```

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

```

contact_information

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*). `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
address_line	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	xs:string	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	xs:string	Optional	Single	Contains the hours (in local time) that the location is open (<i>NB: this element is deprecated in favor of the more structured :ref: ‘single-csv-hours-open’ element. It is strongly encouraged that data providers move toward contributing hours in this format.</i>)	If the element is invalid or not present, then the implementation is required to ignore it.
hours_opening	xs:string	Optional	Single	References an <i>hours_open</i> 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.
lat_long_single-csv-lat-lng	xs:string	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.

```
1 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  
2 ci0827,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,  
  ↳,,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,off001  
3 ci0828,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,  
  ↳,,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,vs01
```

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	<code>xs:date</code>	Required	Single	Specifies when the election is being held. The <i>Date</i> 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.
election_type	<code>xs:string</code>	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.
state_id	<code>xs:IDREF</code>	Required	Single	Specifies a link to the <i>State</i> element where the election is being held.	If the field is invalid, then the implementation is required to ignore the Election element containing it.
is_statewide	<code>xs:boolean</code>	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	<code>xs:string</code>	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.
registration_info	<code>xs:string</code>	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.
absentee_ballot_info	<code>xs:string</code>	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.
results_uri	<code>xs:anyURI</code>	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.
polling_hours	<code>xs:string</code>	Optional	Single	Contains the hours (in local time) that Election Day polling locations are open. If polling hours differ in specific polling locations, alternative hours may be specified in the <i>polling_location</i> object (NB: <i>this element is deprecated in favor of the more structured :ref:‘single-csv-hours-open‘ element. It is strongly encouraged</i>	If the element is invalid or not present, then the implementation is required to


```
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

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

Tag	Data Type	Re-quired?	Re-peats?	Description	Error Handling
ab-sen-tee_uri	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.
am_i_registered	xs:boolean	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.
bal-lot_tracking_uri	xs:anyURI	Optional	Single	Specifies the web address for tracking information for a ballot cast by mail	If the field is invalid or not present, then the implementation is required to ignore it.
bal-lot_tracking_provisional_uri	xs:anyURI	Optional	Single	Specifies the web address for tracking information for a provisional ballot. To support EAC guidelines for "Processing Provisional Ballots" (https://www.eac.gov/research-and-data/provisional-voting/)	If the field is invalid or not present, then the implementation is required to ignore it.
de-part-ment	single-csv-department	Required	Re-peats	Describes the administrative body for a particular voter service.	There must be at least one valid <i>Department</i> in each <i>ElectionAdministration</i> element. If no valid <i>Department</i> objects are present, the implementation is required to ignore the <i>ElectionAdministration</i> object that contains it/them.
elec-tion_notices	single-csv-election-notice	Optional	Single	A place for election administrators to post last minute and emergency notifications pertaining to the election.	If the element is invalid or not present, then the implementation is required to ignore it.
elec-tions_uri	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.
reg-istra-tion_uri	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.
rules_uri	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.
what_is_on_my_ballot	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.
where_does_i_vote	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.

```

1 id,absentee_uri,am_i_registered_uri,ballot_tracking_uri,ballot_tracking_provisional_
  ↪uri,election_notice_text,election_notice_uri,elections_uri,registration_uri,rules_
  ↪uri,what_is_on_my_ballot_uri,where_do_i_vote_uri
2 ea123,https://example.com/absentee,https://example.com/am-i-registered,https://www.
  ↪vote.virginia.gov/,https://www.vote.virginia.gov/,This is an emergency notification_
  ↪for this election.,https://www.yadayada.gov,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
3 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
4 ea625,https://example.com/absentee3,https://example.com/am-i-registered3,https://
  ↪example.com/elections3,https://example.com/registration3,This is an emergency_
  ↪notification for this election.,https://example.com/rules3,https://example.com/
  ↪what-is-on-my-ballot3,https://example.com/where-do-i-vote3

```

department

Tag	Data Type	Re-quired	Re-peats?	Description	Error Handling
con- tact_infor- mation	single- contact- information	Op- tional	Sin- gle	Contact and physical address informa- tion for the election administration body (see single-csv-contact-information).	If the element is invalid or not present, then the implementa- tion is required to ignore it.
elec- tion_official_	xs:IDREF person_id	Op- tional	Sin- gle	The individual to contact at the election administration office. The specified per- son should be the <i>election official</i> .	If the field is invalid or not present, then the implementa- tion is required to ignore it.
voter_service	single- csv-voter- service	Op- tional	Re- peats	The types of services and appropriate contact individual available to voters.	If the element is invalid or not present, then the implementa- tion is required to ignore it.

```

1 id,election_official_person_id,election_administration_id
2 dep01,per50002,ea123
3 dep02,per50002,ea345
4 dep03,per50002,ea625
5 dep04,per50002,ea625

```

voter_service

Tag	Data Type	Required?	Repeats?	Description	Error Handling
contact_information	single-csv-contact-information	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	xs:string	Optional	Single	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
election_official_person_id	xs:IDREF	Optional	Single	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
type	<i>voter_service</i>	Optional	Single	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
other_type	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.

```

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

```

contact_information

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*). `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
address_line	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	xs:string	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	xs:string	Optional	Single	Contains the hours (in local time) that the location is open (<i>NB: this element is deprecated in favor of the more structured :ref: ‘single-csv-hours-open’ element. It is strongly encouraged that data providers move toward contributing hours in this format.</i>)	If the element is invalid or not present, then the implementation is required to ignore it.
hours_opening	xs:string	Optional	Single	References an <i>hours_open</i> 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.
lat_long_single-csv-lat-lng	xs:string	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.

```

1 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
2 ci0827,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,
  ↳,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,off001
3 ci0828,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,
  ↳,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,vs01

```

election_notice

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
elec- tion_notice_text	xs:string	Re- quired	Single	The last minute or emergency notification text should be placed here.	If the element is invalid, then the implementation is required to ignore the ElectionNotice element containing it.
elec- tion_notice_uri	xs:string	Op- tional	Single	Optional URL for additional information related to the last minute or emergency notification.	If the field is invalid or not present, then the implementation is required to ignore it.

voter_service

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
con- tact_information	single-csv- contact- information	Op- tional	Single	The contact for a particular voter service.	If the element is invalid or not present, then the implementation is required to ignore it.
descrip- tion	xs:string	Op- tional	Single	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
elec- tion_official_person_id	xs:IDREF	Op- tional	Single	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
type	<i>voter_service_type</i>	Op- tional	Single	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
other_type	xs:string	Op- tional	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.

```

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

```

election_notice

Tag	Data Type	Required?	Repeats?	Description	Error Handling
election_notice_text	xs:string	Required	Single	The last minute or emergency notification text should be placed here.	If the element is invalid, then the implementation is required to ignore the ElectionNotice element containing it.
election_notice_uri	xs:string	Optional	Single	Optional URL for additional information related to the last minute or emergency notification.	If the field is invalid or not present, then the implementation is required to ignore it.

electoral_district

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
external_identifier	external_identifier	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	district_type	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.
other_type	xs:string	Optional	Single	Allows for cataloging a new district_type option when Type is specified as “other”.	If the field is invalid or not present, then the implementation is required to ignore it.

```

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

```

external_file

The `ExternalFile` object holds a reference to a file external to the feed itself. External files are packaged along with the VIP feed into a single, archived file.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
file_uri	xs:anyURI	Required	Single	The URI of the external file.	If the field is invalid, then the implementation is required to ignore the <code>ExternalFile</code> element containing it.
checksum_id	xs:IDREF	Required	Single	The cryptographic checksum of the referenced external file.	If the element is invalid, then the implementation is required to ignore the <code>ExternalFile</code> element containing it.

```
1 id,file_uri,checksum_id
2 efl,precinct_shapes.zip,ch1
```

checksum

The `Checksum` object contains information about a cryptographic checksum, including the raw checksum value and the cryptographic hash algorithm used to compute it.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
algorithm	<i>checksum_algorithm</i>	Required	Single	The cryptographic hash algorithm used to compute the checksum value.	If the field is invalid, then the implementation is required to ignore the <code>Checksum</code> element containing it.
value	xs:string	Required	Single	The raw cryptographic checksum value encoded as a non-delimited, lowercase hexadecimal string.	If the field is invalid, then the implementation is required to ignore the <code>Checksum</code> element containing it.

```
1 id,algorithm,value
2 ch1,sha-256,65b634c5037f8a344616020d8060d233daa37b0f032a71d0d15ad7a5d3afa68e
```

external_geospatial_feature

The `ExternalGeospatialFeature` object contains a reference to a geospatial feature (one or more shapes) contained in a separate file external to the VIP feed.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
external_file_id	xs:IDRef	Required	Single	Links to the <i>external_file</i> containing the geospatial shape(s) that define the feature's boundary.	If the field is invalid, then the implementation is required to ignore the ExternalGeospatialFeature element containing it.
file_format	xs:string	Required	Single	The format of the geospatial file.	If the field is invalid, then the implementation is required to ignore the ExternalGeospatialFeature element containing it.
feature_identifier	single-identifier	Required	Repeats	Identifiers indicating which specific shape(s) to use from the geospatial file. These refer to identifiers within the referenced external file. This is a repeated field in the XML specification, but a scalar field in the CSV specification. If more than one identifier is required with the CSV specification, multiple values can be provided by delimited by space.	If the element is invalid, then the implementation is required to ignore the ExternalGeospatialFeature element containing it.

```

1 id,external_file_id,file_format,shape_identifiers
2 egfl,efl,shp,0 7 9

```

feature_identifier

Tag	Data Type	Required?	Repeats?	Description	Error Handling
index	xs:int	Optional	Single	The index value for the shapefile feature.	If the field is invalid or not present, then the implementation is required to ignore it.

external_identifier

Tag	Data Type	Required	Repeats?	Description	Error Handling
type	identifier	Required	Single	Specifies the type of identifier. Must be one of the valid types as defined by <i>identifier_type</i> .	If the field is invalid or not present, the implementation is required to ignore the <code>ElectionIdentifier</code> containing it.
other_type	string	Optional	Single	Allows for cataloging an <code>ExternalIdentifier</code> type that falls outside the options listed in <i>identifier_type</i> . 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 <code>ElectionIdentifier</code> containing it.

external_identifiers

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 `single-csv-contest-base`
- *electoral_district*
- *locality*
- *office*
- *party*
- *precinct*
- *state*

Tag	Data Type	Required	Repeats?	Description	Error Handling
external_identifier	<i>external_identifier</i>	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 <code>ExternalIdentifiers</code> to be valid. If no valid ‘ExternalIdentifier’_ is present, the implementation is required to ignore the <code>ExternalIdentifiers</code> element.

feature_identifier

Tag	Data Type	Re-quired?	Re-peats?	Description	Error Handling
in-index	xs:int	Optional	Single	The index value for the shapefile feature.	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	Re-quired?	Re-peats?	Description	Error Handling
start_time	single-csv-time-with-zone	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.
end_time	single-csv-time-with-zone	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.

time_with_zone

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

```

1 <HoursOpen id="hours0001">
2   <Schedule>
3     <Hours>
4       <StartTime>06:00:00-05:00</StartTime>
5       <EndTime>12:00:00-05:00</EndTime>
6     </Hours>
7     <Hours>
8       <StartTime>13:00:00-05:00</StartTime>
9       <EndTime>19:00:00-05:00</EndTime>
10    </Hours>
11    <StartDate>2013-11-05</StartDate>
12    <EndDate>2013-11-05</EndDate>
13  </Schedule>
14 </HoursOpen>

```

hours_open

A structured way of describing the days and hours that a place such as a *office* or *polling_location* 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	single-csv-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	single-csv-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.
is_only_by_appointment	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.
is_or_by_appointment	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.
is_subject_to_change	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.
start_date	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.
end_date	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.

```

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

```

hours

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

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
start_time	single-csv-time-with-zone	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.
end_time	single-csv-time-with-zone	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.

time_with_zone

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
```

```

1 <HoursOpen id="hours0001">
2   <Schedule>
3     <Hours>
4       <StartTime>06:00:00-05:00</StartTime>
5       <EndTime>12:00:00-05:00</EndTime>
6     </Hours>
7     <Hours>
8       <StartTime>13:00:00-05:00</StartTime>
9       <EndTime>19:00:00-05:00</EndTime>
10    </Hours>
11    <StartDate>2013-11-05</StartDate>
12    <EndDate>2013-11-05</EndDate>
13  </Schedule>
14 </HoursOpen>

```

html_color_string

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

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

internationalized_text

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 single-csv-contest-base * Any element that extends single-csv-ballot-selection-base * *candidate* * single-csv-contact-information * *election* * *election_administration* * *office* * *party* * *person* * *polling_location* * *source* NOTE: Internationalized Text is not currently supported for CSV submissions. “

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
text	xs:string	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.

language_string

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.

```

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

```

lat_long

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	Re-quired	Re-peats?	Description	Error Handling
lati-tude	xs:double	Re-quired	Single	The latitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
lon-gi-tude	xs:double	Re-quired	Single	The longitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
latlng_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.

locality

The Locality object represents the jurisdiction below the [state](#) (e.g. county).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
election_administration_id	xs:IDREF	Optional	Single	Links to the locality's <i>election_administration</i> object.	If the field is invalid or not present, then the implementation is required to ignore it.
external_identifier	external_identifiers	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.
is_mail_only	boolean	Optional	Single	Determines if the locality runs mail-only elections. If this is true, then all precincts a part of the locality will also run mail-only elections. Drop boxes may be used in addition to this flag using a <i>polling location</i> record configured as a Drop Box.	If the field is missing or invalid, the implementation is required to assume <i>IsMailOnly</i> is false.
name	xs:string	Required	Single	Specifies the name of a locality.	If the field is invalid, then the implementation is required to ignore the <i>Locality</i> element containing it.
polling_location_ids	xs:string	Optional	Single	Specifies a link to a set of the locality's :ref:polling locations <single-csv-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.
state_id	xs:IDREF	Required	Single	References the locality's <i>state</i> .	If the field is invalid, then the implementation is required to ignore the <i>Locality</i> element containing it.
type	district_type	Optional	Single	Defines the kind of locality (e.g. county, town, et al.), which is one of the various <i>DistrictType</i> enumerations.	If the field is invalid or not present, then the implementation is required to ignore it.
other_type	xs:string	Optional	Single	Allows for defining a type of locality that falls outside the options listed in <i>DistrictType</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

```

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

```

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
contact_information_id	xs:IDREF	Optional	Repeats	Links to the single-csv-contact-information element associated with the office.	If the element is invalid or not present, then the implementation is required to ignore it.
description	xs:string	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.
electoral_district_id	xs:IDREF	Required	Single	Links to the <i>electoral_district</i> element associated with the office.	If the field is invalid or not present, the implementation is required to ignore the <i>Office</i> element containing it.
external_identifiers	xs:IDREF	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.
filing_deadline	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.
is_partisan	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	xs:string	Required	Single	The name of the office.	If the field is invalid or not present, the implementation is required to ignore the <i>Office</i> element containing it.
office_holder_person_ids	xs:IDREFS	Optional	Single	Links to the <i>person</i> 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	single-csv-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.

```

1 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
2 off001,ed001,,,,,true,Deputy Chief of Staff,per50003,full-term,2002-01-21,
3 off002,ed001,,,,,true,Deputy Deputy Chief of Staff,per50001,unexpired-term,2002-01-
  ↪21,
4 off003,ed001,,,,,false,General Secretary of Secretaries,per50004,full-term,2002-01-
  ↪21,

```

term

Tag	Data Type	Required?	Repeats?	Description	Error Handling
term_type	office_term_type	Optional	Single	Specifies the type of office term (see <i>office_term_type</i> for valid values).	If the field is invalid or not present, the implementation is required to ignore the <code>Office</code> element containing it.
term_start_date	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.
term_end_date	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.

contact_information

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*). `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
address_line	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	xs:string	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	xs:string	Optional	Single	Contains the hours (in local time) that the location is open (<i>NB: this element is deprecated in favor of the more structured :ref: ‘single-csv-hours-open’ element. It is strongly encouraged that data providers move toward contributing hours in this format.</i>)	If the element is invalid or not present, then the implementation is required to ignore it.
hours_open	xs:string	Optional	Single	References an hours_open 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.
lat_long_csv_lat_lng	xs:string	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.

```

1 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
2 ci0827,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,
  ↪,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,off001
3 ci0828,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,
  ↪,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,vs01

```

ordered_contest

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
con- test_id	xs:IDREF	Required	Single	Links to elements that extend single-csv-contest-base.	If the field is invalid or not present, the implementation is required to ignore the <code>OrderedContest</code> element containing it.
or- dered_ballot_selection_ids	IDREF	Optional	Single	Links to elements that extend single-csv-ballot-selection-base.	If the field is invalid or not present, the implementation is required to ignore it. If an <code>OrderedBallotSelectionIds</code> element is not present, the presumed order of the selection will be the order of single-csv-ballot-selection-base-extended elements referenced by the underlying single-csv-contest-base-extended elements.

```

1 id,contest_id,ordered_ballot_selection_ids
2 oc2025,con001,bs001 bs002 bs003
3 oc3000,con002,bs001

```

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 *candidate_contest* 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
ab- bre- via- tion	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	single-csv-html-color-string	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.
ex- ter- nal_ identifiers	<i>external_identifier_type</i>	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.
is_write_in	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.
leader_person_ids	xs:string	Optional	Single	A reference of <i>person</i> elements which are leaders of the <i>Party</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
logo_uri	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	xs:string	Optional	Single	The name of the party.	If the element is invalid or not present, then the implementation is required to ignore it.

```

1 id,abbreviation,color,external_identifier_type,external_identifier_othertype,
2 external_identifier_value,is_write_in,leader_person_ids,logo_uri,name
3 par01,REP,ff0000,,,,true,,http://example.com/elephant.png,Republican
4 par02,DEM,0000ff,,,,false,per01,http://example.com/donkey.png,Democrat
5 par03,GRN,efefef,,,,,http://example.com/tree.png,Green
par04,WFP,ee99aa,,,,,http://example.com/worker.png,Working Families Party

```

html_color_string

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

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

party_contest

An extension of `single-csv-contest-base` which describes a contest in which the possible ballot selections are of type *party_selection*. 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).

```
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  
2 pcon001,PC1071,bs001 bs002,,Party Election,ed001,all registered voters,,,,false,  
  ↳Straight Party Vote,3,,
```

contest_base

A base model for all Contest types: *ballot_measure_contest*, *candidate_contest*, *party_contest*, and *retention_contest* (NB: the latter because it extends *ballot_measure_contest*).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_selection_ids	xs:IDREFS	Optional	Single	References a set of BallotSelections, which could be of any selection type that extends single-csv-ballot-selection-base.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_sub_title	xs:string	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.
bal- lot_title	xs:string	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.
elec- toral_district_id	xs:IDREF	Required	Single	References an <i>electoral_district</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the ContestBase element containing it.
elec- torate_specification	xs:string	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.
ex- ter- nal_identifiers	xs:string	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.
has_rotations	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	Required	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, then the implementation is required to ignore the ContestBase element containing it.
sequence_order	xs:integer	Optional	Single	Order in which the contests are listed on the ballot. This is the default ordering, and can be overridden by data in a <i>ballot_style</i> element.	If the field is invalid or not present, then the implementation should ignore it.

party_selection

This element extends single-csv-ballot-selection-base to support contests in which the selections can be groups of one or more parties.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
party_ids	IDREFS	Required	Single	One or more <i>party</i> 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.

```

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

```

ballot_selection_base

A base model for all ballot selection types: *ballot_measure_selection*, *candidate_selection*, and *party_selection*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
sequence_order	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 <i>OrderedBallotSlectionIds</i> in <i>ordered_contest</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

person

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*

Tag	Data Type	Required?	Repeats?	Description	Error Handling
contact_information	xs:IDREF	Optional	Repeats	Refers to the associated single-csv-contact-information.	If the element is invalid or not present, then the implementation is required to ignore it.
date_of_birth	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.
external_identifiers	external_identifiers	Optional	Single	Identifiers for this person.	If the element is invalid or not present, then the implementation is required to ignore it.
first_name	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.
full_name	xs:string	Optional	Single	Specifies a person's full name (NB: this information is <i>internationalized_text</i> 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.
last_name	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.
middle_name	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.
party_id	xs:IDREF	Optional	Single	Refers to the associated <i>party</i> . 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 <i>candidate PartyId</i> .	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	xs:string	Optional	Single	Specifies a person's profession (NB: this information is <i>internationalized_text</i> 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.
suffix	xs:string	Optional	Single	Specifies a suffix associated with a person (e.g. Jr.).	If the field is invalid or not present, then the implementation is required to ignore it.
title	xs:string	Optional	Single	A title associated with a person (NB: this information is <i>internationalized_text</i> 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.

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

contact_information

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](#)*). `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
address_line	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	xs:string	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	xs:string	Optional	Single	Contains the hours (in local time) that the location is open (<i>NB: this element is deprecated in favor of the more structured :ref: ‘single-csv-hours-open’ element. It is strongly encouraged that data providers move toward contributing hours in this format.</i>)	If the element is invalid or not present, then the implementation is required to ignore it.
hours_open	xs:string	Optional	Single	References an hours_open 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.
lat_long_csv_lat_lng	xs:string	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.

```
1 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  
2 ci0827,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,  
  ↳,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,off001  
3 ci0828,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,  
  ↳,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,vs01
```

polling_location

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
single-csv-simple-address-type	simple	Optional	Single	Represents the various structured parts of an address to a polling location.	One of AddressStructured and AddressLine should be present for a given Polling Location. If none is present, the implementation is required to ignore the <code>PollingLocation</code> element containing it.
address_line	<code>xs:string</code>	Optional	Repeats	Represents the various parts of an address to a polling location.	One of <code>AddressStructured</code> and <code>AddressLine</code> should be present for a given Polling Location. If none is present, the implementation is required to ignore the <code>PollingLocation</code> element containing it.
directions	<code>xs:string</code>	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	<code>xs:string</code>	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 <code>hours_open</code> 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.
hours_open_id	<code>xs:string</code>	Optional	Single	Links to an <code>hours_open</code> 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.
is_drop_box	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.
is_early_voting	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.
lat_lng	single-csv-lat-lng	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	<code>xs:string</code>	Optional	Single	Name of the polling location.	If the field is invalid or not present, then the implementation is required to ignore it.
photo_uri	<code>xs:string</code>	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.

```

1 id,name,address_line,structured_line_1,structured_city,structured_state,structured_
  ↳zip,directions,hours,photo_uri,hours_open_id,is_drop_box,is_early_voting,latitude,
  ↳longitude,latlng_source
2 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
3 poll002,Public Library,Main St Denver CO,,,,,,next to the checkout counter,7am-8pm,
  ↳www.picture.com,,false,true,38.0754627,78.5014875,Google Maps

```

lat_long

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:double	Required	Single	The latitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
longitude	xs:double	Required	Single	The longitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
latlng_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.

simple_address_type

A SimpleAddressType represents a structured address.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
structured_line_1	xs:string	Required	Single	The address line for a structured address. Should include the street number, street name, and any prefix and suffix.	If no Line1 is provided, the implementation should ignore the SimpleAddressType.
structured_line_2	xs:string	Optional	Single	Additional field for an address	If no Line2 is provided, the implementation should ignore it.
structured_line_3	xs:string	Optional	Single	Additional field for an address	If no Line3 is provided, the implementation should ignore it.
structured_city	xs:string	Required	Single	The City value of a structured address.	If City is not provided, the implementation should ignore the SimpleAddressType.
structured_state	xs:string	Required	Single	The State value of a structured address.	If State is not provided, the implementation should ignore the SimpleAddressType.
structured_zip	xs:string	Optional	Single	The ZIP code of a structured address.	If Zip is not provided, the implementation should ignore the SimpleAddressType.

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).

Voters can be assigned to a precinct in two ways. A voter location modeled by StreetSegment is assigned to a precinct by StreetSegment.PrecinctId. Alternatively, a precinct’s spatial boundary can be modeled with Precinct.SpatialBoundary. Any registered voter address contained within the spatial boundary of the precinct is assigned to that precinct.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ballot_style_id	xs:integer	Optional	Single	Links to the ballot_style , 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.
electoral_district_id	xs:integer	Optional	Single	Links to the single-csv-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.
external_identifier	xs:string	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.
is_mail_only	boolean	Optional	Single	Determines if the precinct runs mail-only elections.	If the field is missing or invalid, the implementation is required to assume <i>IsMailOnly</i> is false.
locality_id	xs:integer	Required	Single	Links to the locality that comprises the precinct.	If the field is invalid, then the implementation is required to ignore the <code>Precinct</code> element containing it.
name	xs:string	Required	Single	Specifies the precinct's name (or number if no name exists).	If the field is invalid, then the implementation is required to ignore the <code>Precinct</code> 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 <i>Name</i> 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.
polling_location	xs:string	Optional	Single	Specifies a link to the precinct's polling_location object(s).	If the field is invalid or not present, then the implementation is required to ignore it.
precinct_split_name	xs:string	Optional	Single	If this field is empty, then this <i>Precinct</i> object represents a full precinct. If this field is present, then this <i>Precinct</i> object represents one portion of a split precinct. Each <i>Precinct</i> object that represents one portion of a split precinct must have the same <i>Name</i> value, but different <i>PrecinctSplitName</i> values. See the <i>sample_feed.xml</i> file for examples.	If the field is invalid or not present, then the implementation is required to ignore it.
spatial_boundary	xs:integer	Optional	Single	Defines the spatial boundary of the precinct. All voter addresses contained within this boundary are assigned to the precinct. If a voter address also maps to a <i>StreetSegment</i> , then the precinct assignment from the <i>StreetSegment</i> will be preferred over the assignment from the spatial boundary.	If the element is invalid or not present, then the implementation is required to ignore it.
3.1. CSV Specification					
ward	xs:string	Optional	Single	Specifies the ward the precinct is contained within.	If the field is invalid or not present,

```

1 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,spatial_boundary_id,ward
2 pre90111,bs00010,ed001,ocd-id,,ocd-division/country:us,false,loc001,203 - GEORGETOWN,
  ↪ 0203,poll001 poll002,split13,sb1,,5
3 pre90112,bs00011,ed002,fips,,42,false,loc001,203 - GEORGETOWN,0203,poll003,split26,,6
4 pre90113,bs00010,ed003,,,,false,loc002,203 - GEORGETOWN,0203,poll004,split54,sb1,7

```

spatial_boundary

The `SpatialBoundary` object defines a boundary in space. This boundary is usually defined by one or more discrete, closed polygonal shapes.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
external_geospatial_feature_id	xs:IDREF	Required	Single	The spatial boundary defined by a geospatial feature that is external to the VIP feed.	If the element is invalid, then the implementation is required to ignore the <code>SpatialBoundary</code> element containing it.

```

1 id,external_geospatial_feature_id
2 sb1,egf1

```

external_geospatial_feature

The `ExternalGeospatialFeature` object contains a reference to a geospatial feature (one or more shapes) contained in a separate file external to the VIP feed.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
external_file_id	xs:IDREF	Required	Single	Links to the <i>external_file</i> containing the geospatial shape(s) that define the feature's boundary.	If the field is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.
file_format	xs:string	Required	Single	The format of the geospatial file.	If the field is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.
feature_identifier	single- text	Required	Repeats	Identifiers indicating which specific shape(s) to use from the geospatial file. These refer to identifiers within the referenced external file. This is a repeated field in the XML specification, but a scalar field in the CSV specification. If more than one identifier is required with the CSV specification, multiple values can be provided by delimited by space.	If the element is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.

```

1 id,external_file_id,file_format,shape_identifiers
2 egfl,efl,shp,0 7 9

```

feature_identifier

Tag	Data Type	Required?	Repeats?	Description	Error Handling
index	xs:int	Optional	Single	The index value for the shapefile feature.	If the field is invalid or not present, then the implementation is required to ignore it.

retention_contest

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
candidate_id	xs:IDREF	Required	Single	Links to the <i>candidate</i> being retained.	If the field is invalid or not present, the implementation is required to ignore the <code>RetentionContest</code> element containing it.
office_id	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.

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	single-csv-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.
is_only_by_appointment	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.
is_or_by_appointment	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.
is_subject_to_change	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.
start_date	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.
end_date	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.

```

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

```

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
start_time	single-csv-time-with-zone	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.
end_time	single-csv-time-with-zone	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.

time_with_zone

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

```

1 <HoursOpen id="hours0001">
2   <Schedule>
3     <Hours>
4       <StartTime>06:00:00-05:00</StartTime>
5       <EndTime>12:00:00-05:00</EndTime>
6     </Hours>
7     <Hours>
8       <StartTime>13:00:00-05:00</StartTime>
9       <EndTime>19:00:00-05:00</EndTime>
10    </Hours>
11    <StartDate>2013-11-05</StartDate>
12    <EndDate>2013-11-05</EndDate>
13  </Schedule>
14 </HoursOpen>

```

simple_address_type

A SimpleAddressType represents a structured address.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
structured_line_1	xs:string	Required	Single	The address line for a structured address. Should include the street number, street name, and any prefix and suffix.	If no Line1 is provided, the implementation should ignore the SimpleAddressType.
structured_line_2	xs:string	Optional	Single	Additional field for an address	If no Line2 is provided, the implementation should ignore it.
structured_line_3	xs:string	Optional	Single	Additional field for an address	If no Line3 is provided, the implementation should ignore it.
structured_city	xs:string	Required	Single	The City value of a structured address.	If City is not provided, the implementation should ignore the SimpleAddressType.
structured_state	xs:string	Required	Single	The State value of a structured address.	If State is not provided, the implementation should ignore the SimpleAddressType.
structured_zip	xs:string	Optional	Single	The ZIP code of a structured address.	If Zip is not provided, the implementation should ignore the SimpleAddressType.

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 <code>Source</code> element containing it.
vip_id	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 <code>Source</code> element containing it.
date_time	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 the <code>Source</code> element containing it.
description	xs:string	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.
organization_uri	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.
feed_contact_information_id	xs:string	Optional	Single	Reference to the <i>person</i> who will respond to inquiries about the information contained within the file.	If the element is invalid or not present, then the implementation is required to ignore it.
terms_of_use_uri	xs:string	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 the <code>Source</code> element containing it.

```

1 id,date_time,description,name,organization_uri,terms_of_use_uri,vip_id,version
2 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

```

spatial_boundary

The `SpatialBoundary` object defines a boundary in space. This boundary is usually defined by one or more discrete, closed polygonal shapes.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
external_geospatial_feature_id	xs:IDREF	Required	Single	The spatial boundary defined by a geospatial feature that is external to the VIP feed.	If the element is invalid, then the implementation is required to ignore the <code>SpatialBoundary</code> element containing it.

```

1 id,external_geospatial_feature_id
2 sb1,egf1

```

external_geospatial_feature

The `ExternalGeospatialFeature` object contains a reference to a geospatial feature (one or more shapes) contained in a separate file external to the VIP feed.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
external_file_id	<code>xs:ID</code>	Required	Single	Links to the <i>external_file</i> containing the geospatial shape(s) that define the feature's boundary.	If the field is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.
file_format	<code>geospatial_format</code>	Required	Single	The format of the geospatial file.	If the field is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.
feature_identifier	single-identifier feature-identifier	Required	Repeats	Identifiers indicating which specific shape(s) to use from the geospatial file. These refer to identifiers within the referenced external file. This is a repeated field in the XML specification, but a scalar field in the CSV specification. If more than one identifier is required with the CSV specification, multiple values can be provided by delimited by space.	If the element is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.

```

1 id,external_file_id,file_format,shape_identifiers
2 egfl,efl,shp,0 7 9

```

feature_identifier

Tag	Data Type	Required?	Repeats?	Description	Error Handling
index	<code>xs:int</code>	Optional	Single	The index value for the shapefile feature.	If the field is invalid or not present, then the implementation is required to ignore it.

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
election_administration_id	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.
external_identifier	external_identifier_type	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	Specifies the name of a state, such as Alabama.	If the field is invalid, then the implementation is required to ignore the State element containing it.
polling_location_ids	xs:string	Optional	Single	Specifies a link to the state's <i>polling locations</i> . 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 <i>precinct</i> element.	If the field is invalid or not present, then the implementation is required to ignore it.

```

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

```

street_segment

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
address_directional	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 the <code>StreetSegment</code> element containing it.
includes_all_addresses	xs:boolean	Optional	Single	Specifies if the segment covers every address on this street. If this is <i>true</i> , then the values of StartHouseNumber and EndHouseNumber should be ignored. The value of OddEvenBoth must be <i>both</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
includes_all_streets	xs:boolean	Optional	Single	Specifies if the segment covers every street in this city. If this is <i>true</i> , 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.
odd_even_both	xs:string	Optional	Single	Specifies whether the odd side of the street (in terms of house numbers), the even side, or both are included in the street segment.	If the field is not present or invalid, the implementation is required to ignore the <code>StreetSegment</code> containing it.
precinct_id	xs:string	Required	Single	References the <i>precinct</i> that contains the entire street segment. If a precinct has a single-csv-spatial-boundary which also contains the entire street segment, then the precinct assignment from the segment will be preferred over the assignment defined by the spatial boundary.	If the field is invalid, then the implementation is required to ignore the <code>StreetSegment</code> element containing it.
start_house_number	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 <code>StreetSegment</code> element containing it. If the StartHouseNumber is greater than the EndHouseNumber , the implementation should ignore the element containing them.
end_house_number	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 <code>StreetSegment</code> element containing it. If the EndHouseNumber is less than the StartHouseNumber , the implementation should ignore the element containing it.
house_number_prefix	xs:string	Optional	Single	Part of a street address. It may contain letters or slashes (e.g., ‘B’ in ‘B22 Main St’). If this value is present then StartHouseNumber must be equal to EndHouseNumber . This field cannot be used if IncludesAllAddresses or IncludesAllStreets are true.	If the field is invalid or not present, then the implementation is required to ignore it.
house_number_suffix	xs:string	Optional	Single	Part of a street address. It may contain letters or slashes (e.g., 1/2 in ‘22 1/2 Main St’). If this value is present then StartHouseNumber must be equal to EndHouseNumber . This field cannot be used if IncludesAllAddresses or IncludesAllStreets are true.	If the field is invalid or not present, then the implementation is required to ignore it.

3.1. CSV Specification

```

1 id,address_direction,city,includes_all_addresses,includes_all_streets,odd_even_both,
  ↳precinct_id,start_house_number,end_house_number,house_number_prefix,house_number_
  ↳suffix,state,street_direction,street_name,street_suffix,unit_number,zip
2 ss000001,N,Washington,false,false,odd,pre90113,101,199,,,DC,NW,Delaware,St,,20001
3 ss000002,S,Washington,true,false,both,pre90112,,,,,DC,SE,Wisconsin,Ave,,20002
4 ss000003,N,Washington,false,false,even,pre90113,100,100,A,1/2,DC,NW,Delaware,St,,
  ↳20001

```

term

Tag	Data Type	Required?	Repeats?	Description	Error Handling
term_type	<i>office_term_type</i>	Optional	Single	Specifies the type of office term (see <i>office_term_type</i> for valid values).	If the field is invalid or not present, the implementation is required to ignore the Office element containing it.
term_start_date	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.
term_end_date	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.

time_with_zone

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
```

```

1 <HoursOpen id="hours0001">
2   <Schedule>
3     <Hours>
4       <StartTime>06:00:00-05:00</StartTime>
5       <EndTime>12:00:00-05:00</EndTime>
6     </Hours>
7     <Hours>
8       <StartTime>13:00:00-05:00</StartTime>
9       <EndTime>19:00:00-05:00</EndTime>
10    </Hours>
11    <StartDate>2013-11-05</StartDate>
12    <EndDate>2013-11-05</EndDate>
13  </Schedule>
14 </HoursOpen>

```

voter_service

Tag	Data Type	Required?	Repeats?	Description	Error Handling
contact_information	single-csv-contact-information	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	xs:string	Optional	Single	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
election_official_person_id	xs:IDREF	Optional	Single	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
type	<i>voter_service</i>	Optional	Single	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
other_type	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.

```

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

```

Enumerations**ballot_measure_type**

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.

candidate_post_election_status

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.

candidate_pre_election_status

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	

checksum_algorithm

Tag	Description
sha-256	256-bit cryptographic hash algorithm of the SHA-2 family
sha-512	512-bit cryptographic hash algorithm of the SHA-2 family

district_type

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 <i>OtherType</i> field to describe it.

geospatial_format

Geospatial file formats that are supported by the VIP specification.

Tag	Description
shp	ESRI Shapefile (reference)

identifier_type

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.

oeb_enum

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.

office_term_type

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.

vote_variation

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.
ap-proval	‘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).
super-majority	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.

voter_service_type

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 department handles the selection and management of polling places.
voter-registration	The department that manages voter registration.
other	Any other service not covered by the above descriptions.

Elements (Separate Pages)

ballot_measure_contest

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
con_statement	xs:string	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.
effect_of_abstain	xs:string	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.
full_text	xs:string	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.
info_uri	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.
passage_threshold	xs:string	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.
pro_statement	xs:string	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.
summary_text	xs:string	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	xs:string	Optional	Single	Specifies the particular type of ballot measure. Must be one of the valid <i>ballot_measure_type</i> options.	If the field is invalid or not present, then the implementation is required to ignore it.
other_type	xs:string	Optional	Single	Allows for cataloging a new <i>ballot_measure_type</i> option, when Type is specified as “other.”	If the field is invalid or not present, then the implementation is required to ignore it.
192				Chapter 3. CSV Documentation	

```

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
2 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,

```

contest_base

A base model for all Contest types: *ballot_measure_contest*, *candidate_contest*, *party_contest*, and *retention_contest* (NB: the latter because it extends *ballot_measure_contest*).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_selection_ids	xs:IDRefs	Optional	Single	References a set of BallotSelections, which could be of any selection type that extends <i>ballot_selection_base</i> .	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_sub_title	xs:string	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.
bal- lot_title	xs:string	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.
elec- toral_district_id	xs:IDRef	Required	Single	References an <i>electoral_district</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the ContestBase element containing it.
elec- torate_specification	xs:string	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.
ex- ter- nal_identifiers	xs:string	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.
has_rotations	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	Required	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, then the implementation is required to ignore the ContestBase element containing it.
sequence_order	xs:integer	Optional	Single	Order in which the contests are listed on the ballot. This is the default ordering, and can be overridden by data in a <i>ballot_style</i> element.	If the field is invalid or not present, then the implementation should ignore it.

ballot_measure_selection

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. BallotMeasureSelection extends *ballot_selection_base*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
selection	xs:string	Required	Single	Selection text for a <i>ballot_measure_contest</i>	If the element is invalid or not present, the implementation is required to ignore the BallotMeasureSelection containing it.

```

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

```

ballot_selection_base

A base model for all ballot selection types: *ballot_measure_selection*, *candidate_selection*, and *party_selection*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
sequence_order	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 <i>OrderedBallotSelectionIds</i> in <i>ordered_contest</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

ballot_style

A container for the contests/measures on the ballot.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
image_uri	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.
ordered_contests_ids	xs:IDREFS	Optional	Single	Reference to a set of <i>ordered_contest</i>	If the field is invalid or not present, then the implementation is required to ignore it.
party_ids	xs:IDREFS	Optional	Single	Reference to a set of :ref:'multi-csv-party's.	If the field is invalid or not present, then the implementation is required to ignore it.

```

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

```

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
ballot_name	xs:string	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, then the implementation is required to ignore the Candidate element containing it.
contact_information	xs:string	Optional	Single	Contact and physical address information for this Candidate and/or their campaign (see <i>contact_information</i>).	If the element is invalid or not present, then the implementation is required to ignore it.
external_identifiers	<i>external_identifier_type</i>	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.
file_date	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.
is_incumbent	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.
is_top_ticket	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.
party_id	xs:IDREF	Optional	Single	Reference to a <i>party</i> 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.
person_id	xs:IDREF	Optional	Single	Reference to a <i>person</i> element with additional information about the candidate.	If the field is invalid or not present, then the implementation is required to ignore it.
post_election_status	<i>election_date_post_election_status</i>	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.
pre_election_status	<i>election_date_pre_election_status</i>	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.

```

1 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
2 can001,Jude Fawley,,,,,2016-12-01,true,false,par01,per50001,,filed
3 can002,Arabella Donn,,,,,2016-12-01,false,false,par02,per50002,,qualified
4 can003,John Coltrane,,,,,2016-09-23,false,false,par02,per50003,,qualified
5 can004,Miles Davis,,,,,2016-05-26,false,false,par01,per50004,,qualified

```


candidate_contest

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
number_elected	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.
office_ids	xs:IDREFS	Optional	Single	References a set of <i>office</i> 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 <i>BallotSelectionIds</i> . E.g., if the various <i>BallotSelectionIds</i> reference <i>candidate_selection</i> elements which reference the candidate for President first and Vice-President second, the <i>OfficeIds</i> 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.
primary_party_ids	xs:IDREFS	Optional	Single	References <i>party</i> 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.
votes_allowed	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.

```

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

```

contest_base

A base model for all Contest types: *ballot_measure_contest*, *candidate_contest*, *party_contest*, and *retention_contest* (NB: the latter because it extends *ballot_measure_contest*).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_selection_ids	xs:IDRefs	Optional	Single	References a set of BallotSelections, which could be of any selection type that extends <i>ballot_selection_base</i> .	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_sub_title	xs:string	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.
bal- lot_title	xs:string	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.
elec- toral_district_id	xs:IDRef	Required	Single	References an <i>electoral_district</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the ContestBase element containing it.
elec- torate_specification	xs:string	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.
ex- ter- nal_identifiers	xs:string	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.
has_rotations	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	Required	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, then the implementation is required to ignore the ContestBase element containing it.
sequence_order	xs:integer	Optional	Single	Order in which the contests are listed on the ballot. This is the default ordering, and can be overridden by data in a <i>ballot_style</i> element.	If the field is invalid or not present, then the implementation should ignore it.

candidate_selection

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
candidate_ids	xs:IDREFS	Optional	Single	References a set of <i>candidate</i> 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.
endorsement_party_ids	xs:IDREFS	Optional	Single	References a set of <i>party</i> 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.
is_write_in	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.

```

1 id,sequence_order,candidate_ids,endorsement_party_ids,is_write_in
2 cs001,3,can004,par01,false
3 cs002,2,can001 can002,par03 par02,false
4 cs003,1,can003,par02 par03,true

```

ballot_selection_base

A base model for all ballot selection types: *ballot_measure_selection*, *candidate_selection*, and *party_selection*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
sequence_order	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 <i>OrderedBallotSlectionIds</i> in <i>ordered_contest</i> .	If the field is invalid or not present, then the implementation is required to 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	<code>xs:date</code>	Required	Single	Specifies when the election is being held. The <i>Date</i> 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.
election_type	<code>xs:string</code>	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.
state_id	<code>xs:IDREF</code>	Required	Single	Specifies a link to the <i>State</i> element where the election is being held.	If the field is invalid, then the implementation is required to ignore the Election element containing it.
is_statewide	<code>xs:boolean</code>	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	<code>xs:string</code>	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.
registration_info	<code>xs:string</code>	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.
absentee_ballot_info	<code>xs:string</code>	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.
results_uri	<code>xs:anyURI</code>	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.
polling_hours	<code>xs:string</code>	Optional	Single	Contains the hours (in local time) that Election Day polling locations are open. If polling hours differ in specific polling locations, alternative hours may be specified in the <i>polling_location</i> object (NB: this element is deprecated in favor of the more structured <i>ref:‘multi-csv-hours-open‘</i> element. It is strongly encouraged to use the <i>ref:‘multi-csv-hours-open‘</i> element instead.)	If the element is invalid or not present, then the implementation is required to ignore it.

```
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

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
absentee_uri	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.
am_i_registered	xs:boolean	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.
ballot_tracking_uri	xs:anyURI	Optional	Single	Specifies the web address for tracking information for a ballot cast by mail	If the field is invalid or not present, then the implementation is required to ignore it.
ballot_tracking_provisional_uri	xs:anyURI	Optional	Single	Specifies the web address for tracking information for a provisional ballot. To support EAC guidelines for "Processing Provisional Ballots" (https://www.eac.gov/research-and-data/provisional-voting/)	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 <i>Department</i> in each <i>ElectionAdministration</i> element. If no valid <i>Department</i> objects are present, the implementation is required to ignore the <i>ElectionAdministration</i> object that contains it/them.
election_notice	election_notice	Optional	Single	A place for election administrators to post last minute and emergency notifications pertaining to the election.	If the element is invalid or not present, then the implementation is required to ignore it.
elections_uri	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.
registration_uri	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.
rules_uri	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.
what_is_on_my_ballot	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.
where_does_my_ballot_go	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.

```

1 id,absentee_uri,am_i_registered_uri,ballot_tracking_uri,ballot_tracking_provisional_
  ↪uri,election_notice_text,election_notice_uri,elections_uri,registration_uri,rules_
  ↪uri,what_is_on_my_ballot_uri,where_do_i_vote_uri
2 ea123,https://example.com/absentee,https://example.com/am-i-registered,https://www.
  ↪vote.virginia.gov/,https://www.vote.virginia.gov/,This is an emergency notification_
  ↪for this election.,https://www.yadayada.gov,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
3 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
4 ea625,https://example.com/absentee3,https://example.com/am-i-registered3,https://
  ↪example.com/elections3,https://example.com/registration3,This is an emergency_
  ↪notification for this election.,https://example.com/rules3,https://example.com/
  ↪what-is-on-my-ballot3,https://example.com/where-do-i-vote3

```

department

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
con- tact_information	<i>con- tact_informational</i>	Op- tional	Single	Contact and physical address information for the election administration body (see <i>contact_information</i>).	If the element is invalid or not present, then the implementation is required to ignore it.
elec- tion_official_person_id	xs:IDREF	Op- tional	Single	The individual to contact at the election administration office. The specified person should be the <i>election official</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
voter_service	multi- csv- voter- service	Op- tional	Re- peats	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.

```

1 id,election_official_person_id,election_administration_id
2 dep01,per50002,ea123
3 dep02,per50002,ea345
4 dep03,per50002,ea625
5 dep04,per50002,ea625

```

voter_service

Tag	Data Type	Required?	Repeats?	Description	Error Handling
contact_information	<i>contact_information</i>	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	xs:string	Optional	Single	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
election_official_person_id	xs:IDREF	Optional	Single	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
type	<i>voter_service_type</i>	Optional	Single	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
other_type	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.

```

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

```

contact_information

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*). `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
address_line	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	xs:string	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	xs:string	Optional	Single	Contains the hours (in local time) that the location is open (<i>NB: this element is deprecated in favor of the more structured :ref: ‘multi-csv-hours-open’ element. It is strongly encouraged that data providers move toward contributing hours in this format).</i>	If the element is invalid or not present, then the implementation is required to ignore it.
hours_open_id	IDREF	Optional	Single	References an <i>hours_open</i> 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.
lat_lng	lat_lng	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.

```

1 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
2 ci0827,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,
  ↪,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,off001
3 ci0828,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,
  ↪,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,vs01

```


election_notice

Tag	Data Type	Required?	Repeats?	Description	Error Handling
election_notice_text	xs:string	Required	Single	The last minute or emergency notification text should be placed here.	If the element is invalid, then the implementation is required to ignore the ElectionNotice element containing it.
election_notice_uri	xs:string	Optional	Single	Optional URL for additional information related to the last minute or emergency notification.	If the field is invalid or not present, then the implementation is required to ignore it.

voter_service

Tag	Data Type	Required?	Repeats?	Description	Error Handling
contact_information	contact_information	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	xs:string	Optional	Single	Long description of the services available.	If the element is invalid or not present, then the implementation is required to ignore it.
election_official_person_id	xs:IDREF	Optional	Single	The <i>authority</i> for a particular voter service.	If the field is invalid or not present, then the implementation is required to ignore it.
type	voter_service_type	Optional	Single	The type of <i>voter service</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
other_type	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.

```

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

```

electoral_district

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
external_identifier	external_identifier	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	district_type	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.
other_type	xs:string	Optional	Single	Allows for cataloging a new <i>district_type</i> option when Type is specified as "other".	If the field is invalid or not present, then the implementation is required to ignore it.

```

1 id,external_identifier_type,external_identifier_othertype,external_identifier_value,
2 ↪name,number,type,other_type
3 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

```

external_file

The `ExternalFile` object holds a reference to a file external to the feed itself. External files are packaged along with the VIP feed into a single, archived file.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
file_uri	xs:anyURI	Required	Single	The URI of the external file.	If the field is invalid, then the implementation is required to ignore the <code>ExternalFile</code> element containing it.
checksum_id	xs:IDREF	Required	Single	The cryptographic checksum of the referenced external file.	If the element is invalid, then the implementation is required to ignore the <code>ExternalFile</code> element containing it.

```

1 id,file_uri,checksum_id
2 efl,precinct_shapes.zip,ch1

```

checksum

The `Checksum` object contains information about a cryptographic checksum, including the raw checksum value and the cryptographic hash algorithm used to compute it.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
algorithm	<i>checksum_algorithm</i>	Required	Single	The cryptographic hash algorithm used to compute the checksum value.	If the field is invalid, then the implementation is required to ignore the <code>Checksum</code> element containing it.
value	<code>xs:string</code>	Required	Single	The raw cryptographic checksum value encoded as a non-delimited, lowercase hexadecimal string.	If the field is invalid, then the implementation is required to ignore the <code>Checksum</code> element containing it.

```
1 id,algorithm,value
```

```
2 chl,sha-256,65b634c5037f8a344616020d8060d233daa37b0f032a71d0d15ad7a5d3afa68e
```

external_identifier

Tag	Data Type	Required?	Repeats?	Description	Error Handling
type	<i>identifier_type</i>	Required	Single	Specifies the type of identifier. Must be one of the valid types as defined by <i>identifier_type</i> .	If the field is invalid or not present, the implementation is required to ignore the <code>ElectionIdentifier</code> containing it.
other_type	<code>string</code>	Optional	Single	Allows for cataloging an <code>ExternalIdentifier</code> type that falls outside the options listed in <i>identifier_type</i> . 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	<code>xs:string</code>	Required	Single	Specifies the identifier.	If the field is invalid or not present, the implementation is required to ignore the <code>ElectionIdentifier</code> containing it.

external_identifiers

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 *contest_base*
- *electoral_district*
- *locality*
- *office*

- *party*
- *precinct*
- *state*

Tag	Data Type	Required?	Repeats?	Description	Error Handling
external_identifier	<i>external_identifier</i>	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.

hours_open

A structured way of describing the days and hours that a place such as a *office* or *polling_location* 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	<i>schedule</i>	Required	Repeats	Defines a block of days and hours that a place will be open.	At least one valid <i>Schedule</i> must be present for HoursOpen to be valid. If no valid <i>Schedule</i> 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.
is_only_by_appointment	is_only_by_appointment	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.
is_or_by_appointment	is_or_by_appointment	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.
is_subject_to_change	is_subject_to_change	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.
start_date	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.
end_date	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.

```

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

```

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
start_time	time_with_zone	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.
end_time	time_with_zone	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.

time_with_zone

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
```

```

1 <HoursOpen id="hours0001">
2   <Schedule>
3     <Hours>
4       <StartTime>06:00:00-05:00</StartTime>
5       <EndTime>12:00:00-05:00</EndTime>
6     </Hours>
7     <Hours>
8       <StartTime>13:00:00-05:00</StartTime>
9       <EndTime>19:00:00-05:00</EndTime>
10    </Hours>
11    <StartDate>2013-11-05</StartDate>
12    <EndDate>2013-11-05</EndDate>
13  </Schedule>
14 </HoursOpen>

```

internationalized_text

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 *contest_base* * Any element that extends *ballot_selection_base* * *candidate* * *contact_information* * *election* * *election_administration* * *office* * *party* * *person* * *polling_location* * *source*

NOTE: Internationalized Text is not currently supported for CSV submissions.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
text	xs:string	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.

language_string

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.

```

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

```

locality

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
election_administration_id	xs:IDREF	Optional	Single	Links to the locality's <i>election_administration</i> object.	If the field is invalid or not present, then the implementation is required to ignore it.
external_identifier	external_identifier	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.
is_mail_only	xs:boolean	Optional	Single	Determines if the locality runs mail-only elections. If this is true, then all precincts a part of the locality will also run mail-only elections. Drop boxes may be used in addition to this flag using a <i>polling location</i> record configured as a Drop Box.	If the field is missing or invalid, the implementation is required to assume <i>IsMailOnly</i> is false.
name	xs:string	Required	Single	Specifies the name of a locality.	If the field is invalid, then the implementation is required to ignore the <i>Locality</i> element containing it.
polling_location_ids	xs:string	Optional	Single	Specifies a link to a set of the locality's :ref:polling locations <multi-csv-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.
state_id	xs:IDREF	Required	Single	References the locality's <i>state</i> .	If the field is invalid, then the implementation is required to ignore the <i>Locality</i> element containing it.
type	district_type	Optional	Single	Defines the kind of locality (e.g. county, town, et al.), which is one of the various <i>DistrictType enumerations</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
other_type	xs:string	Optional	Single	Allows for defining a type of locality that falls outside the options listed in <i>DistrictType</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

```

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

```

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
contact_information_id	<code>xs:IDREF</code>	Optional	Repeats	Links to the contact_information element associated with the office.	If the element is invalid or not present, then the implementation is required to ignore it.
description	<code>xs:string</code>	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.
electoral_district_id	<code>xs:IDREF</code>	Required	Single	Links to the electoral_district element associated with the office.	If the field is invalid or not present, the implementation is required to ignore the <code>Office</code> element containing it.
external_identifiers	<code>xs:IDREF</code>	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.
filing_deadline	<code>xs:date</code>	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.
is_partisan	<code>xs:boolean</code>	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	<code>xs:string</code>	Required	Single	The name of the office.	If the field is invalid or not present, the implementation is required to ignore the <code>Office</code> element containing it.
office_holder_person_ids	<code>xs:IDREFS</code>	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.

```

1 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
2 off001,ed001,,,,,true,Deputy Chief of Staff,per50003,full-term,2002-01-21,
3 off002,ed001,,,,,true,Deputy Deputy Chief of Staff,per50001,unexpired-term,2002-01-
  ↪21,
4 off003,ed001,,,,,false,General Secretary of Secretaries,per50004,full-term,2002-01-
  ↪21,

```


term

Tag	Data Type	Required?	Repeats?	Description	Error Handling
term_type	office_term_type	Optional	Single	Specifies the type of office term (see <i>office_term_type</i> for valid values).	If the field is invalid or not present, the implementation is required to ignore the <code>Office</code> element containing it.
term_start_date	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.
term_end_date	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.

contact_information

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*). `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
address_line	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	xs:string	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	xs:string	Optional	Single	Contains the hours (in local time) that the location is open (<i>NB: this element is deprecated in favor of the more structured :ref: ‘multi-csv-hours-open’ element. It is strongly encouraged that data providers move toward contributing hours in this format).</i>	If the element is invalid or not present, then the implementation is required to ignore it.
hours_open_id	IDREF	Optional	Single	References an <i>hours_open</i> 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.
lat_lng	lat_lng	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.

```

1 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
2 ci0827,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,
  ↪,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,off001
3 ci0828,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,
  ↪,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,vs01

```

ordered_contest

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

Tag	Data Type	Required	Repeats	Description	Error Handling
contest_id	xs:IDREF	Required	Single	Links to elements that extend <i>contest_base</i> .	If the field is invalid or not present, the implementation is required to ignore the <code>OrderedContest</code> element containing it.
ordered_ballot_selection_ids	IDREF	Optional	Single	Links to elements that extend <i>ballot_selection_base</i> .	If the field is invalid or not present, the implementation is required to ignore it. If an <code>OrderedBallotSelectionIds</code> element is not present, the presumed order of the selection will be the order of <i>ballot_selection_base</i> -extended elements referenced by the underlying <i>contest_base</i> -extended elements.

```

1 id,contest_id,ordered_ballot_selection_ids
2 oc2025,con001,bs001 bs002 bs003
3 oc3000,con002,bs001

```

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 *candidate_contest* 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
ab- bre- via- tion	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	html_color_string	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.
ex- ter- nal_iden- tifiers	external_identifier_type	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.
is_write_in	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.
leader_person_ids	list of person_ids	Optional	Single	A reference of <i>person</i> elements which are leaders of the <i>Party</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
logo_uri	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	xs:string	Optional	Single	The name of the party.	If the element is invalid or not present, then the implementation is required to ignore it.

```

1 id,abbreviation,color,external_identifier_type,external_identifier_othertype,
2 external_identifier_value,is_write_in,leader_person_ids,logo_uri,name
3 par01,REP,ff0000,,,true,,http://example.com/elephant.png,Republican
4 par02,DEM,0000ff,,,false,per01,http://example.com/donkey.png,Democrat
5 par03,GRN,efefef,,,,,http://example.com/tree.png,Green
par04,WFP,ee99aa,,,,,http://example.com/worker.png,Working Families Party

```

html_color_string

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

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

party_contest

An extension of *contest_base* which describes a contest in which the possible ballot selections are of type *party_selection*. 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).

```

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
2  pcon001,PC1071,bs001 bs002,,Party Election,ed001,all registered voters,,,,false,
   ↳Straight Party Vote,3,,

```

contest_base

A base model for all Contest types: *ballot_measure_contest*, *candidate_contest*, *party_contest*, and *retention_contest* (NB: the latter because it extends *ballot_measure_contest*).

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ab- bre- via- tion	xs:string	Optional	Single	An abbreviation for the contest.	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_selection_ids	xs:IDRefs	Optional	Single	References a set of BallotSelections, which could be of any selection type that extends <i>ballot_selection_base</i> .	If the field is invalid or not present, then the implementation should ignore it.
bal- lot_sub_title	xs:string	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.
bal- lot_title	xs:string	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.
elec- toral_district_id	xs:IDRef	Required	Single	References an <i>electoral_district</i> element that represents the geographical scope of the contest.	If the field is invalid, then the implementation is required to ignore the ContestBase element containing it.
elec- torate_specification	xs:string	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.
ex- ter- nal_identifiers	xs:string	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.
has_rotations	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	Required	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, then the implementation is required to ignore the ContestBase element containing it.
sequence_order	xs:integer	Optional	Single	Order in which the contests are listed on the ballot. This is the default ordering, and can be overrides by data in a <i>ballot_style</i> element.	If the field is invalid or not present, then the implementation should ignore it.

party_selection

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
party_ids	IDREFS	Required	Single	One or more <i>party</i> 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.

```

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

```

ballot_selection_base

A base model for all ballot selection types: *ballot_measure_selection*, *candidate_selection*, and *party_selection*.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
sequence_order	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 <i>OrderedBallotSlectionIds</i> in <i>ordered_contest</i> .	If the field is invalid or not present, then the implementation is required to ignore it.

person

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*

Tag	Data Type	Re-quired?	Re-peats?	Description	Error Handling
contact_information	xs:IDREF	Optional	Repeats	Refers to the associated <i>contact_information</i> .	If the element is invalid or not present, then the implementation is required to ignore it.
date_of_birth	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.
external_identifiers	external_identifiers	Optional	Single	Identifiers for this person.	If the element is invalid or not present, then the implementation is required to ignore it.
first_name	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.
full_name	xs:string	Optional	Single	Specifies a person's full name (NB: this information is <i>internationalized_text</i> 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.
last_name	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.
middle_name	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.
party_id	xs:IDREF	Optional	Single	Refers to the associated <i>party</i> . 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 <i>candidate PartyId</i> .	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	xs:string	Optional	Single	Specifies a person's profession (NB: this information is <i>internationalized_text</i> 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.
suffix	xs:string	Optional	Single	Specifies a suffix associated with a person (e.g. Jr.).	If the field is invalid or not present, then the implementation is required to ignore it.
title	xs:string	Optional	Single	A title associated with a person (NB: this information is <i>internationalized_text</i> 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.


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

contact_information

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](#)*). `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
address_line	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	xs:string	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	xs:string	Optional	Single	Contains the hours (in local time) that the location is open (<i>NB: this element is deprecated in favor of the more structured :ref: ‘multi-csv-hours-open’ element. It is strongly encouraged that data providers move toward contributing hours in this format).</i>	If the element is invalid or not present, then the implementation is required to ignore it.
hours_open_id	IDREF	Optional	Single	References an <i>hours_open</i> 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.
lat_lng	lat_lng	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.

```

1 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
2 ci0827,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,
  ↪,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,off001
3 ci0828,The White House,1600 Pennsylvania Ave,,,josh@example.com,,Early to very late,,
  ↪,,,Josh Lyman,555-111-2222,http://lemonlyman.example.com,vs01

```

polling_location

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

Tag	Data Type	Required?	Repeats?	Description	Error Handling
<i>simple_address_line</i>	simple	Optional	Single	Represents the various structured parts of an address to a polling location.	One of AddressStructured and AddressLine should be present for a given Polling Location. If none is present, the implementation is required to ignore the PollingLocation element containing it.
address_line	xs:string	Optional	Repeats	Represents the various parts of an address to a polling location.	One of AddressStructured and AddressLine should be present for a given Polling Location. If none is present, the implementation is required to ignore the PollingLocation element containing it.
directions	xs:string	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	xs:string	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 <i>hours_open</i> 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.
hours_open_id	xs:string	Optional	Single	Links to an <i>hours_open</i> 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.
is_drop_box	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.
is_early_voting	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.
lat_lng	lat_long	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.
photo_uri	xs:string	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.

```

1 id,name,address_line,structured_line_1,structured_city,structured_state,structured_
  ↪zip,directions,hours,photo_uri,hours_open_id,is_drop_box,is_early_voting,latitude,
  ↪longitude,latlng_source
2 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

```

(continues on next page)

(continued from previous page)

```
poll002,Public Library,Main St Denver CO,,,,,next to the checkout counter,7am-8pm,
↪www.picture.com,,false,true,38.0754627,78.5014875,Google Maps
```

lat_long

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:double	Required	Single	The latitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
longitude	xs:double	Required	Single	The longitude of the polling location.	If the field is invalid, then the implementation is required to ignore it.
latlng_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.

simple_address_type

A SimpleAddressType represents a structured address.

Tag	Data Type	Required	Repeats?	Description	Error Handling
structured_line_1	xs:string	Required	Single	The address line for a structured address. Should include the street number, street name, and any prefix and suffix.	If no Line1 is provided, the implementation should ignore the SimpleAddressType.
structured_line_2	xs:string	Optional	Single	Additional field for an address	If no Line2 is provided, the implementation should ignore it.
structured_line_3	xs:string	Optional	Single	Additional field for an address	If no Line3 is provided, the implementation should ignore it.
structured_city	xs:string	Required	Single	The City value of a structured address.	If City is not provided, the implementation should ignore the SimpleAddressType.
structured_state	xs:string	Required	Single	The State value of a structured address.	If State is not provided, the implementation should ignore the SimpleAddressType.
structured_zip	xs:string	Optional	Single	The ZIP code of a structured address.	If Zip is not provided, the implementation should ignore the SimpleAddressType.

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).

Voters can be assigned to a precinct in two ways. A voter location modeled by *StreetSegment* is assigned to a precinct by *StreetSegment.PrecinctId*. Alternatively, a precinct's spatial boundary can be modeled with *Precinct.SpatialBoundary*. Any registered voter address contained within the spatial boundary of the precinct is assigned to that precinct.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
ballot_style_id	xs:integer	Optional	Single	Links to the ballot_style , 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.
electoral_district_id	xs:integer	Optional	Single	Links to the ref:multi-csv-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.
external_identifiers	external identifiers	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.
is_mail_only	boolean	Optional	Single	Determines if the precinct runs mail-only elections.	If the field is missing or invalid, the implementation is required to assume <i>IsMailOnly</i> is false.
locality_id	xs:integer	Required	Single	Links to the locality that comprises the precinct.	If the field is invalid, then the implementation is required to ignore the <code>Precinct</code> element containing it.
name	xs:string	Required	Single	Specifies the precinct's name (or number if no name exists).	If the field is invalid, then the implementation is required to ignore the <code>Precinct</code> 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 <i>Name</i> 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.
polling_location	xs:string	Optional	Single	Specifies a link to the precinct's polling_location object(s).	If the field is invalid or not present, then the implementation is required to ignore it.
precinct_split_name	xs:string	Optional	Single	If this field is empty, then this <i>Precinct</i> object represents a full precinct. If this field is present, then this <i>Precinct</i> object represents one portion of a split precinct. Each <i>Precinct</i> object that represents one portion of a split precinct must have the same <i>Name</i> value, but different <i>PrecinctSplitName</i> values. See the <i>sample_feed.xml</i> file for examples.	If the field is invalid or not present, then the implementation is required to ignore it.
spatial_boundary	xs:integer	Optional	Single	Defines the spatial boundary of the precinct. All voter addresses contained within this boundary are assigned to the precinct. If a voter address also maps to a StreetSegment , then the precinct assignment from the <code>StreetSegment</code> will be preferred over the assignment from the spatial boundary.	If the element 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,

```

1 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,spatial_boundary_id,ward
2 pre90111,bs00010,ed001,ocd-id,,ocd-division/country:us,false,loc001,203 - GEORGETOWN,
  ↪ 0203,poll001 poll002,split13,sb1,,5
3 pre90112,bs00011,ed002,fips,,42,false,loc001,203 - GEORGETOWN,0203,poll003,split26,,6
4 pre90113,bs00010,ed003,,,,false,loc002,203 - GEORGETOWN,0203,poll004,split54,sb1,7

```

spatial_boundary

The `SpatialBoundary` object defines a boundary in space. This boundary is usually defined by one or more discrete, closed polygonal shapes.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
external_geospatial_feature_id	xs:IDREF	Required	Single	The spatial boundary defined by a geospatial feature that is external to the VIP feed.	If the element is invalid, then the implementation is required to ignore the <code>SpatialBoundary</code> element containing it.

```

1 id,external_geospatial_feature_id
2 sb1,egf1

```

external_geospatial_feature

The `ExternalGeospatialFeature` object contains a reference to a geospatial feature (one or more shapes) contained in a separate file external to the VIP feed.

Tag	Data Type	Required?	Repeats?	Description	Error Handling
external_file_id	xs:IDREF	Required	Single	Links to the <i>external_file</i> containing the geospatial shape(s) that define the feature's boundary.	If the field is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.
file_format	geospatial_format	Required	Single	The format of the geospatial file.	If the field is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.
feature_id	feature_id	Required	Repeats	Identifiers indicating which specific shape(s) to use from the geospatial file. These refer to identifiers within the referenced external file. This is a repeated field in the XML specification, but a scalar field in the CSV specification. If more than one identifier is required with the CSV specification, multiple values can be provided by delimited by space.	If the element is invalid, then the implementation is required to ignore the <code>ExternalGeospatialFeature</code> element containing it.

```
1 id,external_file_id,file_format,shape_identifiers
2 egfl,efl,shp,0 7 9
```

feature_identifier

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
in- dex	xs:int	Op- tional	Single	The index value for the shapefile feature.	If the field is invalid or not present, then the implementation is required to ignore it.

retention_contest

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

Tag	Data Type	Re- quired?	Re- peats?	Description	Error Handling
can- di- date_id	xs:IDREF	Re- quired	Single	Links to the <i>can- didate</i> being re- tained.	If the field is invalid or not present, the implementation is required to ignore the RetentionContest element containing it.
of- fice_id	xs:IDREF	Op- tional	Single	Links to the infor- mation about the office.	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 <code>Source</code> element containing it.
vip_id	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 <code>Source</code> element containing it.
date_time	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 the <code>Source</code> element containing it.
description	xs:string	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.
organization_uri	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.
feed_contact_information	xs:string	Optional	Single	Reference to the <i>person</i> who will respond to inquiries about the information contained within the file.	If the element is invalid or not present, then the implementation is required to ignore it.
terms_of_use_uri	xs:string	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 the <code>Source</code> element containing it.

```

1 id,date_time,description,name,organization_uri,terms_of_use_uri,vip_id,version
2 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

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
election_administration_id	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.
external_identifier	external_identifier_type	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	Specifies the name of a state, such as Alabama.	If the field is invalid, then the implementation is required to ignore the State element containing it.
polling_location_ids	xs:string	Optional	Single	Specifies a link to the state's <i>polling locations</i> . 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 <i>precinct</i> element.	If the field is invalid or not present, then the implementation is required to ignore it.

```

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

```

street_segment

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
address	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 the <code>StreetSegment</code> element containing it.
includes_all_addresses	xs:boolean	Optional	Single	Specifies if the segment covers every address on this street. If this is <i>true</i> , then the values of StartHouseNumber and EndHouseNumber should be ignored. The value of OddEvenBoth must be <i>both</i> .	If the field is invalid or not present, then the implementation is required to ignore it.
includes_all_streets	xs:boolean	Optional	Single	Specifies if the segment covers every street in this city. If this is <i>true</i> , 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.
odd_even_both	xs:string	Optional	Single	Specifies whether the odd side of the street (in terms of house numbers), the even side, or both are included in the street segment.	If the field is not present or invalid, the implementation is required to ignore the <code>StreetSegment</code> containing it.
precinct_id	xs:string	Required	Single	References the <i>precinct</i> that contains the entire street segment. If a precinct has a <i>spatial_boundary</i> which also contains the entire street segment, then the precinct assignment from the segment will be preferred over the assignment defined by the spatial boundary.	If the field is invalid, then the implementation is required to ignore the <code>StreetSegment</code> element containing it.
start_house_number	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 <code>StreetSegment</code> element containing it. If the StartHouseNumber is greater than the EndHouseNumber , the implementation should ignore the element containing them.
end_house_number	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 <code>StreetSegment</code> element containing it. If the EndHouseNumber is less than the StartHouseNumber , the implementation should ignore the element containing it.
house_number_prefix	xs:string	Optional	Single	Part of a street address. It may contain letters or slashes (e.g., ‘B’ in ‘B22 Main St’). If this value is present then StartHouseNumber must be equal to EndHouseNumber . This field cannot be used if IncludesAllAddresses or IncludesAllStreets are true.	If the field is invalid or not present, then the implementation is required to ignore it.
house_number_suffix	xs:string	Optional	Single	Part of a street address. It may contain letters or slashes (e.g., 1/2 in ‘22 1/2 Main St’). If this value is present then StartHouseNumber must be equal to EndHouseNumber . This field cannot be used if IncludesAllAddresses or IncludesAllStreets are true.	If the field is invalid or not present, then the implementation is required to ignore it.
3.1. CSV Specification				StartHouseNumber must be equal to EndHouseNumber . This field cannot be used if IncludesAllAddresses or IncludesAllStreets are true.	231
state	xs:string	Required	Single	Specifies the two-letter state abbreviation of	If the field is invalid, then the im-

```

1 id,address_direction,city,includes_all_addresses,includes_all_streets,odd_even_both,
  ↳precinct_id,start_house_number,end_house_number,house_number_prefix,house_number_
  ↳suffix,state,street_direction,street_name,street_suffix,unit_number,zip
2 ss000001,N,Washington,false,false,odd,pre90113,101,199,,,DC,NW,Delaware,St,,20001
3 ss000002,S,Washington,true,false,both,pre90112,,,,,DC,SE,Wisconsin,Ave,,20002
4 ss000003,N,Washington,false,false,even,pre90113,100,100,A,1/2,DC,NW,Delaware,St,,
  ↳20001

```

Enumerations (Separate Pages)

ballot_measure_type

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.

candidate_post_election_status

Tag	Description
advanced-to-runoff	For contests in which the top <i>N</i> 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.

candidate_pre_election_status

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	

checksum_algorithm

Tag	Description
sha-256	256-bit cryptographic hash algorithm of the SHA-2 family
sha-512	512-bit cryptographic hash algorithm of the SHA-2 family

district_type

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 <i>OtherType</i> field to describe it.

geospatial_format

Geospatial file formats that are supported by the VIP specification.

Tag	Description
shp	ESRI Shapefile (reference)

identifier_type

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.

oeb_enum

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.

office_term_type

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.

vote_variation

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.
ap-proval	‘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).
super-majority	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.

voter_service_type

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 department handles the selection and management of polling places.
voter-registration	The department that manages voter registration.
other	Any other service not covered by the above descriptions.

4.1 Data Best Practices

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

4.1.1 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.

4.1.2 Element Identifiers

Most elements within the VIP feed require unique identifiers, `xs:ID` data types. Conformance to `xs:ID` requires the identifying record to:

- begin with a letter or underscore
- only contain letters, digits, hyphens and periods
- be unique across the VIP data set

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, where 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

4.1.3 File Structure

All XML and CSV files should be encoded UTF-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.

4.1.4 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.

4.1.5 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:

```

1 <BallotMeasureContest id="bm390616670907">
2   <BallotSelectionId>bms390616670907</BallotSelectionId>
3   <ElectoralDistrictId>ed3906177703103</ElectoralDistrictId>
4   <Name>Proposed Tax Levy School District</Name>
5   <SequenceOrder>34</SequenceOrder>
6   <FullText>
7     <Text language="en">An additional tax for the benefit of the Lockland Local
8     ↪ School District, County of Hamilton,
9     ↪ Ohio, for the purpose of CURRENT EXPENSES at a rate not exceeding eleven and two-
10    ↪ tenths (11.2) mills for each
11    ↪ one dollar of valuation, which amounts to one dollar and twelve cents ($1.12)
12    ↪ for each one hundred dollars of
13    ↪ valuation, for a continuing period of time, commencing in 2015, first due in
14    ↪ calendar year 2016.</Text>
15   </FullText>
16   <SummaryText>
17     <Text language="en">4 Proposed Tax Levy</Text>
18   </SummaryText>
19   <Type>other</Type>
20   <OtherType>bond</OtherType>
21 </BallotMeasureContest>

```

4.1.6 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.

4.1.7 Geospatial Data

The following sections provide guidance and best practices on using geospatial data with a VIP feed. Geospatial data represents the geographic modeling of a shape on the Earth's surface (i.e. a polygon on a map), and within the context of a VIP feed is primarily used to model the boundary of voter precincts. In places where voter precinct shapes are available, this capability is intended to be straightforward and lightweight to integrate with existing GIS tooling.

Geodetic Datum

VIP exclusively uses the 84 revision of the World Geodetic System (WGS 84) as the geodetic reference system by which geospatial coordinates are defined. This applies to geospatial coordinates provided within the VIP feed itself (e.g. `PollingLocation.LatLng`) as well as coordinates provided in an external geospatial file.

Assigning Voters to Precincts

Voter precincts are the atomic unit of electoral districts in the U.S., and provide the mechanism by which voters are mapped to their polling places, ballot information and more. It is critical that voter locations are mapped to the correct precinct in a VIP feed in order to provide the most accurate and reliable voting information.

There are two mechanisms by which voters can be assigned to precincts in a VIP feed:

1. **Street segments** A voter address that maps to a street segment is assigned to the precinct given by `StreetSegment.PrecinctId`. This is the traditional approach of assigning voters to precincts that has been in use since the beginning of the VIP specification.
2. **Containment within a precinct boundary** A geocoded voter address that is contained within the geographic boundary of a precinct is considered assigned to that precinct. The geographic boundary of each precinct is defined by `Precinct.SpatialBoundary`. This is a newer approach of assigning voters to precincts that was supported starting with version 6.0 of the VIP specification. Compared to the approach of using street segments, geospatial data offers a far more accurate and reliable solution to mapping voters to precincts, and thus should be the preferred approach whenever possible.

Providing both street segments and precinct shapes

There are some cases where precinct boundaries alone are not sufficient to accurately map all voters to their precinct. For example, if a precinct boundary divides an apartment building, there is no way to distinguish the correct precinct for voters that live in this building using a two-dimensional shape on a map.

To mitigate this limitation of geospatial data, it is possible to provide **both** street segments and precinct shapes in the same VIP feed. If a voter's location is determined to map to a street segment *and* is also contained within a precinct shape, the precinct assignment from the street segment will be preferred.

Using the above example where a precinct boundary runs through an apartment building, this scenario could be handled by providing street segments in the VIP feed, in addition to precinct shapes, to specify the mapping of apartment numbers to precinct. For N apartments in the building, there could be N street segments provided in the feed, each with a distinct `StreetSegment.UnitNumber` and precinct assignment. If a voter address maps to one of these street segments, the precinct assignment from the segment will supersede a precinct assignment given by containment with a precinct shape.

Exporting and packaging geospatial files with a VIP feed

Geospatial data files are provided in a native geospatial format. Each geospatial file should adhere to standard industry conventions and requirements of the corresponding `GeospatialFormat`. In most cases, these files will be exported directly from a GIS tool or an Election Management System and delivered alongside a VIP feed.

All referenced external files need to be packaged with the VIP feed file, and archived together within a single ZIP file. The following is an example of the file structure for the case of an XML feed file that includes ESRI shapefiles.

Example file structure:

- vipfeed-19-2012-11-06-Arizona.zip
 - vipfeed-19-2012-11-06-Arizona.xml
 - precinct_shapes1.zip
 - * precinct_shapes1.shp
 - * precinct_shapes1.shx
 - * precinct_shapes1.dbf
 - precinct_shapes2.zip
 - * precinct_shapes2.shp
 - * precinct_shapes2.shx
 - * precinct_shapes2.dbf

The expected file type and structure of each individual geospatial file will depend on the GeospatialFormat being used. The external file referenced from the VIP feed may be a flat file, but it could also be a ZIP file containing multiple relevant files, as the geospatial format requires. The following provides geospatial data file requirements by format.

Expected file type and structure by Geospatial Format:

Geospatial-Format	Expected file type	Description
shp (ESRI shapefile)	.zip	<p>The referenced external file should be a ZIP archive containing, at a minimum, all files required by the ESRI Shapefile format. The filename referenced from the VIP feed should be the name of the ZIP archive for the shapefile. Required files within the archive include a main <i>.shp</i> geometry file, a <i>.shx</i> index file, and a <i>.dbf</i> attributes file. Other optional files as part of the ESRI Shapefile specification are permitted, but may be ignored.</p> <p>Individual files within the shapefile archive are identified by file extension. For example, the main geometry file is identified by the file within the archive with a <i>.shp</i> file extension, regardless of the file name. It is therefore required that there is only one file per expected file type within the archive.</p>

Referencing specific shapes within a geospatial data file

Geospatial data files will usually contain many shapes. For example, depending on how the source shape data is managed, it may be easiest to export a single file containing all precinct shapes represented in the VIP feed. A precinct boundary is modeled in the feed as a reference to the external geospatial data file, but in most cases only one or a few shapes contained in that file are relevant to the precinct.

The specific shape(s) within the external file that comprise the spatial boundary of the geometric feature are captured by the field **ExternalGeospatialFeature.ShapeIdentifier**. ShapeIdentifier is a repeated string field, but the expected value will depend on the geospatial format of the external file. For example, an integer type is expected when using the ESRI shapefile format, so the string value of ShapeIdentifier should be parsable as an integer.

Expected type of ShapeIdentifier by GeospatialFormat:

Geospatial-Format	ShapeIdentifier expected type	Description
shp (ESRI shapefile)	32-bit integer	ShapeIdentifier should be parsable as a 32-bit integer. Geometric features in an ESRI shapefile are ordered in sequence, and the ShapeIdentifier value corresponds to the zero-based index of a record within the file. For example, a ShapeIdentifier value of “35” is a reference to the 36th sequential record in the shapefile.

External file checksums

Geospatial data is provided in the form of supplemental files external to the VIP feed itself. Since this effectively divides the full set of information of the feed across multiple files, it’s essential that references between files be reliable. This is ensured by verifiable cryptographic checksums.

Each external file reference includes the name of the file and its checksum. A checksum includes both the raw cryptographic hash of the file’s contents, as well as information about which cryptographic algorithm was used to compute the value. A consumer of a VIP feed should be able to compute a checksum value of an external file using the same algorithm and independently verify it matches the checksum value in the feed.

It’s worth noting that having a file checksum also introduces an opportunity for consumers of the data to optimize their processing of it. If the contents of a VIP feed are updated, but the checksum for an external geospatial data file has not changed, then the consumer could omit having to reprocess the geospatial aspects of the feed.

Requirements for precinct shapes

The following are practical requirements when defining the spatial boundary of a Precinct element (whether a precinct or precinct split) with geospatial shapes.

- The resolution of a polygon for a precinct shape can be as coarse or fine as needed, so long as the shape accurately represents the boundary of the precinct.
- No two shapes should overlap.
- All polygons must form a closed loop. That is, the polygon should start and end from the same point.
- The border of a polygon must not intersect itself.
- The spatial boundary of a Precinct is defined by a single geospatial feature. That feature, however, may contain one or more discrete and non-overlapping polygons as necessary to define the full extent of the boundary.
- ShapeIdentifier must be a valid, existing reference in the external file.
- ShapeIdentifier must have an expected type according to the table above.