

Package ‘readaec’

August 2, 2026

Title Access Australian Electoral Commission Data

Version 0.2.0

Description Provides clean, tidy access to Australian Electoral Commission (AEC) federal election data. Includes results for the House of Representatives and Senate from 2007 onwards, at both division and polling place level, plus by-elections from 2008 onwards, the 2023 referendum, and the full distribution of preferences. Data is downloaded directly from the AEC <<https://results.aec.gov.au>> on first use and cached locally for subsequent calls.

License MIT + file LICENSE

Encoding UTF-8

Language en-AU

RoxygenNote 7.3.3

Imports httr2, readr, dplyr, cli, glue, tools

Suggests ggplot2, purrr, tidyr, sf, knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

URL <https://charlescoverdale.github.io/readaec/>,
<https://github.com/charlescoverdale/readaec>

BugReports <https://github.com/charlescoverdale/readaec/issues>

NeedsCompilation no

Author Charles Coverdale [aut, cre]

Maintainer Charles Coverdale <charlesfcoverdale@gmail.com>

Repository CRAN

Date/Publication 2026-08-02 08:00:02 UTC

Contents

clear_cache	2
get_by_election_candidates	3
get_by_election_fp	4
get_by_election_tcp	4
get_by_election_tpp	5
get_candidates	6
get_dop	6
get_enrolment	7
get_fp	8
get_fp_by_booth	8
get_members_elected	9
get_polling_places	10
get_referendum_by_booth	10
get_referendum_turnout	11
get_senate	12
get_senators_elected	12
get_swing	13
get_tcp	14
get_tcp_by_booth	15
get_tpp	16
get_tpp_by_booth	16
get_turnout	17
list_by_elections	18
list_elections	18
list_referendums	19
Index	20

clear_cache	<i>Clear the local AEC data cache</i>
-------------	---------------------------------------

Description

Deletes all files downloaded and cached by readaec. The next function call will re-download fresh data from the AEC.

Usage

clear_cache()

Value

Invisibly returns NULL. Called for its side effect of deleting cached files.

Examples

```
op <- options(readaec.cache_dir = tempdir())
clear_cache()
options(op)
```

```
get_by_election_candidates
```

Get by-election candidates

Description

Get by-election candidates

Usage

```
get_by_election_candidates(division, year = NULL, refresh = FALSE)
```

Arguments

division	Division name (e.g. "Farrer").
year	By-election year. Only needed where a division has had more than one by-election (e.g. Mayo in 2008 and 2018).
refresh	If TRUE, re-download from the AEC even if a cached copy exists.

Value

A tidy data frame of candidates.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_by_election_candidates("Farrer")
options(op)
```

get_by_election_fp	<i>Get by-election first preference votes by polling place</i>
--------------------	--

Description

By-election results are published at polling place level. Use [list_by_elections\(\)](#) to see which by-elections are available.

Usage

```
get_by_election_fp(division, year = NULL, refresh = FALSE)
```

Arguments

division	Division name (e.g. "Farrer").
year	By-election year. Only needed where a division has had more than one by-election (e.g. Mayo in 2008 and 2018).
refresh	If TRUE, re-download from the AEC even if a cached copy exists.

Value

A tidy data frame of first preference votes by polling place.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_by_election_fp("Farrer")
options(op)
```

get_by_election_tcp	<i>Get by-election two-candidate preferred votes by polling place</i>
---------------------	---

Description

Get by-election two-candidate preferred votes by polling place

Usage

```
get_by_election_tcp(division, year = NULL, refresh = FALSE)
```

Arguments

division	Division name (e.g. "Farrer").
year	By-election year. Only needed where a division has had more than one by-election (e.g. Mayo in 2008 and 2018).
refresh	If TRUE, re-download from the AEC even if a cached copy exists.

Value

A tidy data frame of TCP votes by candidate by polling place.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_by_election_tcp("Farrer")
options(op)
```

get_by_election_tpp	<i>Get by-election two-party preferred votes by polling place</i>
---------------------	---

Description

Note that TPP figures are only meaningful where the final two candidates were ALP and Coalition. In by-elections decided between other candidates (such as Farrer in 2026), use [get_by_election_tcp\(\)](#) instead.

Usage

```
get_by_election_tpp(division, year = NULL, refresh = FALSE)
```

Arguments

division	Division name (e.g. "Farrer").
year	By-election year. Only needed where a division has had more than one by-election (e.g. Mayo in 2008 and 2018).
refresh	If TRUE, re-download from the AEC even if a cached copy exists.

Value

A tidy data frame of TPP votes by polling place.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_by_election_tpp("Dunkley")
options(op)
```

get_candidates	<i>Get all candidates for an election</i>
----------------	---

Description

Get all candidates for an election

Usage

```
get_candidates(year, chamber = "house", refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
chamber	"house" or "senate".
refresh	If TRUE, re-download from the AEC even if a cached copy exists.

Value

A tidy data frame of candidates.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_candidates(2022)
get_candidates(2022, chamber = "senate")
options(op)
```

get_dop	<i>Get the full distribution of preferences by division</i>
---------	---

Description

Returns the count-by-count distribution of preferences for every division: each exclusion round, the candidate excluded, and where their preferences flowed. This is the dataset for analysing seats won from second or third place on preference flows, which division-level TPP and TCP figures cannot show.

Usage

```
get_dop(year, refresh = FALSE)
```

Arguments

year	Election year. Use <code>list_elections()</code> to see available years.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame with one row per candidate per count per division, including countnumber, calculationtype (preference count, transfer count, and percentages), and calculationvalue.

Examples

```
op <- options(readaec.cache_dir = tempdir())
dop <- get_dop(2025)

# Final count in a single seat
mel <- subset(dop, division == "Melbourne")
subset(mel, countnumber == max(countnumber))
options(op)
```

get_enrolment

*Get enrolment by division***Description**

Get enrolment by division

Usage

```
get_enrolment(year, refresh = FALSE)
```

Arguments

year	Election year. Use <code>list_elections()</code> to see available years.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame of enrolment figures by division.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_enrolment(2022)
options(op)
```

get_fp	<i>Get first preference votes by division</i>
--------	---

Description

Get first preference votes by division

Usage

```
get_fp(year, refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame of first preference votes by division.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_fp(2022)
options(op)
```

get_fp_by_booth	<i>Get first preference votes by polling place</i>
-----------------	--

Description

Get first preference votes by polling place

Usage

```
get_fp_by_booth(year, state = NULL, refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
state	Filter to a specific state (e.g. "VIC"). NULL returns all states.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame of first preference votes by polling place.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_fp_by_booth(2022, state = "VIC")
options(op)
```

get_members_elected	<i>Get members elected to the House of Representatives</i>
---------------------	--

Description

Get members elected to the House of Representatives

Usage

```
get_members_elected(year, refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame of elected members.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_members_elected(2022)
options(op)
```

get_polling_places	<i>Get polling place locations</i>
--------------------	------------------------------------

Description

Returns all polling place addresses and coordinates for a given election.

Usage

```
get_polling_places(year, division = NULL, refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
division	Filter to a specific division name. NULL returns all.
refresh	If TRUE, re-download from the AEC even if a cached copy exists.

Value

A tidy data frame of polling places with lat/lon coordinates.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_polling_places(2022)
get_polling_places(2022, division = "Kooyong")
options(op)
```

get_referendum_by_booth	<i>Get referendum results by polling place</i>
-------------------------	--

Description

Returns Yes and No votes for every polling place, including formal and informal counts.

Usage

```
get_referendum_by_booth(year = 2023, state = NULL, refresh = FALSE)
```

Arguments

year	Referendum year. Use list_referendums() to see available years. Defaults to 2023 (the Voice referendum).
state	Filter to a specific state (e.g. "VIC"). NULL returns all states.
refresh	If TRUE, re-download from the AEC even if a cached copy exists.

Value

A tidy data frame of Yes/No votes by polling place.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_referendum_by_booth(2023, state = "TAS")
options(op)
```

```
get_referendum_turnout
```

Get referendum turnout

Description

Get referendum turnout

Usage

```
get_referendum_turnout(year = 2023, by = "division", refresh = FALSE)
```

Arguments

year	Referendum year. Use list_referendums() to see available years. Defaults to 2023 (the Voice referendum).
by	Aggregation level: "division" (default) or "state".
refresh	If TRUE, re-download from the AEC even if a cached copy exists.

Value

A tidy data frame of enrolment and turnout.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_referendum_turnout(2023)
get_referendum_turnout(2023, by = "state")
options(op)
```

get_senate	<i>Get Senate first preference votes by state</i>
------------	---

Description

Get Senate first preference votes by state

Usage

```
get_senate(year, refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame of Senate first preference votes by state.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_senate(2022)
options(op)
```

get_senators_elected	<i>Get senators elected</i>
----------------------	-----------------------------

Description

Returns the senators elected at each election, in the order they were elected within each state or territory.

Usage

```
get_senators_elected(year, refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame with one row per senator elected, including state, given_name, surname, party, party_name, and elected_order.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_senators_elected(2025)
options(op)
```

get_swing

*Compare TPP results between two elections***Description**

Calculates the change in two-party preferred vote share between any two federal elections. Joins on division ID (not name) to handle renamed divisions correctly. Divisions that don't appear in both elections are flagged rather than silently dropped.

Usage

```
get_swing(from, to, division = NULL, state = NULL, refresh = FALSE)
```

Arguments

from	Election year to compare from. Use list_elections() to see available years.
to	Election year to compare to.
division	Optionally filter to one or more division names.
state	Optionally filter to a state abbreviation (e.g. "VIC").
refresh	If TRUE, re-download from the AEC even if a cached copy exists.

Details

The winner columns report the party that actually won the seat (from the AEC's members elected file), so crossbench seats are identified correctly. The TPP leader columns report which of ALP or the Coalition led the two-party preferred count, which is not the same thing in seats won by independents and minor parties.

Value

A data frame with one row per division containing:

division	Division name (from the later election)
division_id	AEC division ID
state	State abbreviation

```

alp_pct_from, alp_pct_to      ALP TPP percentage in each election
lnp_pct_from, lnp_pct_to      LNP TPP percentage in each election
alp_swing                     Change in ALP TPP (positive = swing to ALP)
lnp_swing                     Change in LNP TPP (positive = swing to LNP)
winner_from, winner_to        Party that won the seat in each election
tpp_leader_from, tpp_leader_to Which of ALP/LNP led the TPP count
seat_changed                   TRUE if the seat changed hands
redistribution_flag            TRUE if the division only appears in one election

```

Examples

```

op <- options(readaec.cache_dir = tempdir())
# National swing 2019 to 2022
get_swing(2019, 2022)

# Teal seats in Victoria
get_swing(2019, 2022, state = "VIC")

# A single seat
get_swing(2019, 2022, division = "Kooyong")

# Long-run comparison
get_swing(2013, 2025)
options(op)

```

get_tcp	<i>Get two-candidate preferred votes by division</i>
---------	--

Description

Get two-candidate preferred votes by division

Usage

```
get_tcp(year, refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame of TCP votes by division.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_tcp(2022)
options(op)
```

`get_tcp_by_booth`*Get two-candidate preferred votes by polling place*

Description

Get two-candidate preferred votes by polling place

Usage

```
get_tcp_by_booth(year, refresh = FALSE)
```

Arguments

<code>year</code>	Election year. Use list_elections() to see available years.
<code>refresh</code>	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame of TCP votes by candidate by polling place.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_tcp_by_booth(2022)
options(op)
```

get_tpp	<i>Get two-party preferred votes by division</i>
---------	--

Description

Get two-party preferred votes by division

Usage

```
get_tpp(year, refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame of TPP votes by division.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_tpp(2022)
options(op)
```

get_tpp_by_booth	<i>Get two-party preferred votes by polling place</i>
------------------	---

Description

Get two-party preferred votes by polling place

Usage

```
get_tpp_by_booth(year, refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame of TPP votes by polling place.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_tpp_by_booth(2022)
options(op)
```

get_turnout	<i>Get turnout by division</i>
-------------	--------------------------------

Description

Get turnout by division

Usage

```
get_turnout(year, refresh = FALSE)
```

Arguments

year	Election year. Use list_elections() to see available years.
refresh	If TRUE, re-download from the AEC even if a cached copy exists. Useful on election night when counts are still updating.

Value

A tidy data frame of turnout figures by division.

Examples

```
op <- options(readaec.cache_dir = tempdir())
get_turnout(2022)
options(op)
```

list_by_elections	<i>List all federal by-elections</i>
-------------------	--------------------------------------

Description

Lists House of Representatives by-elections with results published on the AEC tally room, from 2005 onwards. The 2014 WA Senate special election is not included as it was not a House by-election.

Usage

```
list_by_elections()
```

Value

A data frame with one row per by-election, including columns division, state, date, year, event_id, and has_downloads. The has_downloads column is TRUE where AEC CSV downloads are available (2008 onwards).

Examples

```
list_by_elections()
```

list_elections	<i>List all available federal elections</i>
----------------	---

Description

List all available federal elections

Usage

```
list_elections()
```

Value

A data frame with one row per election, including columns year, date, event_id, type, and has_downloads. The has_downloads column is TRUE for years where AEC CSV downloads are available (2007 onwards). The 2001 and 2004 elections are listed for reference but their data cannot be fetched.

Examples

```
list_elections()

# Only years with downloadable data
list_elections()[list_elections()$has_downloads, ]
```

list_referendums	<i>List available referendums</i>
------------------	-----------------------------------

Description

List available referendums

Usage

```
list_referendums()
```

Value

A data frame with one row per referendum with results published on the AEC tally room, including columns year, date, event_id, and description.

Examples

```
list_referendums()
```

Index

`clear_cache`, [2](#)

`get_by_election_candidates`, [3](#)
`get_by_election_fp`, [4](#)
`get_by_election_tcp`, [4](#)
`get_by_election_tcp()`, [5](#)
`get_by_election_tpp`, [5](#)
`get_candidates`, [6](#)
`get_dop`, [6](#)
`get_enrolment`, [7](#)
`get_fp`, [8](#)
`get_fp_by_booth`, [8](#)
`get_members_elected`, [9](#)
`get_polling_places`, [10](#)
`get_referendum_by_booth`, [10](#)
`get_referendum_turnout`, [11](#)
`get_senate`, [12](#)
`get_senators_elected`, [12](#)
`get_swing`, [13](#)
`get_tcp`, [14](#)
`get_tcp_by_booth`, [15](#)
`get_tpp`, [16](#)
`get_tpp_by_booth`, [16](#)
`get_turnout`, [17](#)

`list_by_elections`, [18](#)
`list_by_elections()`, [4](#)
`list_elections`, [18](#)
`list_elections()`, [6–10, 12–17](#)
`list_referendums`, [19](#)
`list_referendums()`, [10, 11](#)