

ParlAT: An R package to access the Open Data of the Austrian Parliament

<https://werkstattcodes.github.io/ParlAT/>

Roland Schmidt, PhD – <https://werkstattcodes.com> – rs2903@gmail.com – [@zoowalk.bsky.social](https://www.zoowalk.at)
16 June 2026



Context & Purpose

The ParlAT package seeks to provide an easy way for R users to access the Open Data offered by the Austrian Parliament. By leveraging the Parliament's API, it allows users to retrieve a wide range of parliamentary data, including legislative proposals, committee reports, details on MPs and their mandates, and parliamentary meetings. The package is designed to provide a consistent interface for accessing different datasets, so that researchers, journalists, students, and other R users can all work with parliamentary data without having to write bespoke API calls or wrestle with raw JSON.

Please note that the package is under active development (0.0.5). Also note that neither the package nor its author is affiliated with the Austrian Parliament. It is developed as an independent, open-source side project, maintained in spare time. The goal is simply to lower the barrier to working with Austrian parliamentary data and to encourage more transparent, data-driven engagement with the legislative process.



Function overview

Main functions

`get_committees()` (experimental)
Retrieve committee data
`get_events()`
Get event data
`get_items()`
Get items under negotiation ('Verhandlungsgegenstände')
`get_item_details()` (experimental)
Get detailed information about a parliamentary item ('Verhandlungsgegenstand')
`get_mandates()`
Get mandates

`get_mps()`
Get Members of Parliament
`get_mps_current()`
Get current Members of Parliament
`get_mps_details()`
Get detailed information about Members of Parliament
`get_participation()`
Get participation data
`get_persons()`
Search persons in Austrian political institutions
`get_plenary_meeting_details()`
Get details of a plenary meeting

get_plenary_meetings()

Get data on plenary meetings
`get_transcripts()`
Retrieve transcripts

Auxiliary functions

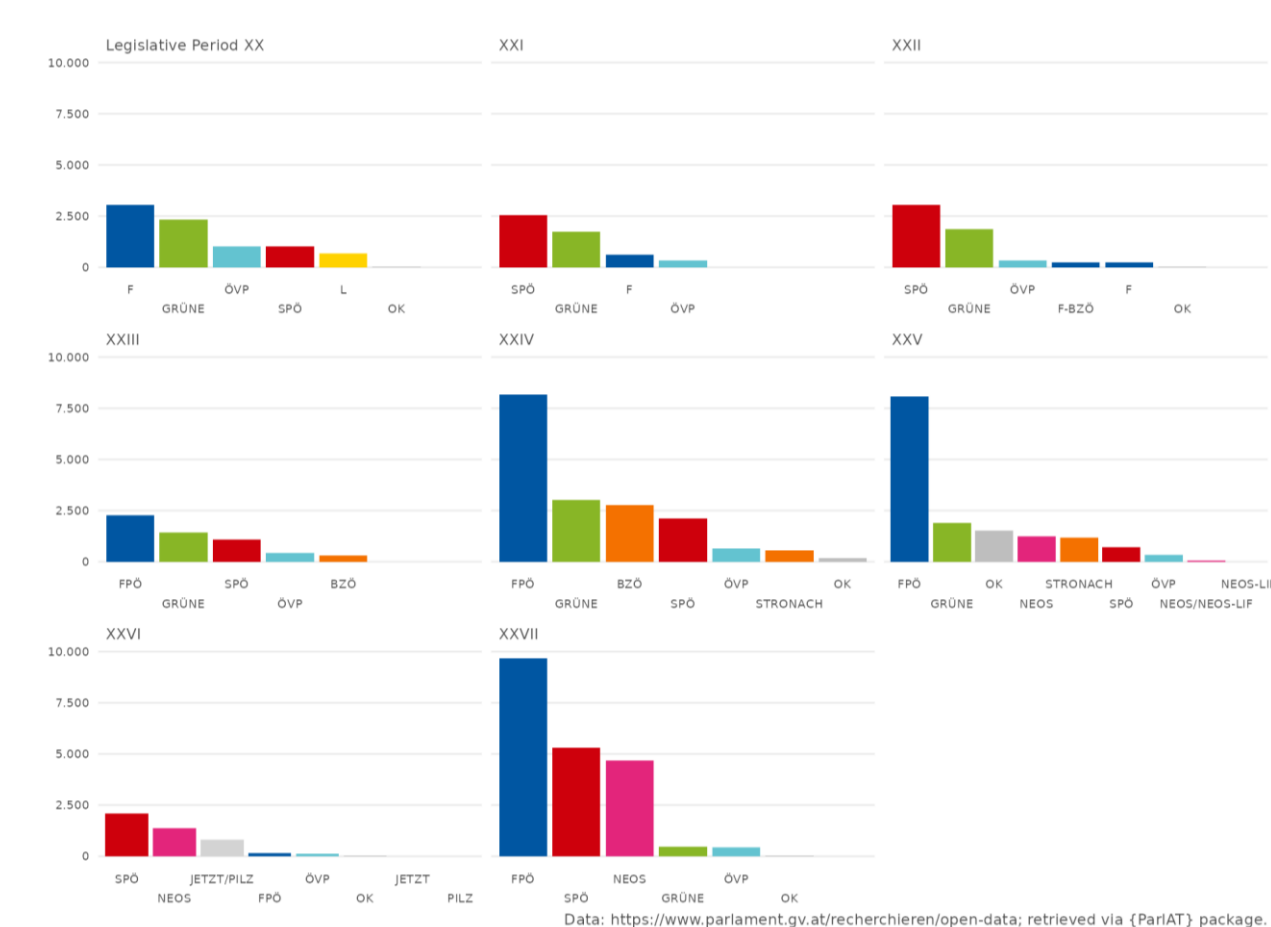
`get_names()`
Get name variants of a Member of Parliament
`get_pad_intern()`
Get unique identification number (pad_intern)
`get_legis_periods()`
Get start and end dates of legislative periods

Examples (abbreviated)

How many oral or written inquiries ('Anfragen') were submitted by party and legislative period?

```
#get_items
df_items <- get_items(item = "J_PRR_M", legis_period = sum(20,27), echo=TRUE)
#> 1 Fetching items from API...
#> ✓ Fetched 77208 items
#> ["GP_CODE":["XXV","XXVI","XXVII","XXVIII","XXIX","XXX","XXXI","XXXII"],"VHG":["J_PRR_M"]
#> https://www.parlament.gv.at/recherchieren/gegenstaende/index.html?FP_001GP_CODE=XXX&F
#> [1] 77208

dplyr::glims(df_items %>% head())
#> Rows: 6
#> Columns: 16
#> $ legis_period
#> $ institution
#> $ date
#> $ item_type
#> $ item_number
#> $ item_number_type
#> $ stage
#> $ item_url
#> $ type_doc
#> $ type_doc_long
#> $ subject
#> $ topics
#> $ keywords
#> $ eurovoc
#> $ persons
#> $ parl_group
```



How many speeches did a specific MP make?

```
#get_pad_intern of Karlheinz Kopf
get_pad_intern("Karlheinz Kopf")
#> # A tibble: 1 x 2
#>   pad_intern names_variants
#>   <chr>         <chr>
#> 1 2822      Karlheinz Kopf

#get all statements held in the plenary of the National Council by Karlheinz Kopf
df_plenary_kopf <- get_mps_details(pad_intern=2822, detail_type="plenary", institution
#> ["PAD_INTERN":["2822"],"GORENUM":["N"]
#> https://www.parlament.gv.at/person/2822?BIO_250PAD_INTERN=2822&BIO_250GORENUM=N&sel
#> [1] 380

df_plenary_kopf %>%
  count(legis_period) %>%
  arrange(legis_period)
#> # A tibble: 9 x 2
#>   legis_period  n
#>   <chr>         <int>
#> 1 XIX           1
#> 2 XX           62
#> 3 XXI          44
#> 4 XXII         56
#> 5 XXIII        26
#> 6 XXIV         105
#> 7 XXV           4
#> 8 XXVI         18
#> 9 XXVII        64
```

Who votes with whom in third readings?

```
# Fetch and inspect one item with voting result details.
example <- get_item_details(
  item_url = "https://www.parlament.gv.at/gegenstand/XXVII/1833/selected?stage=105",
  votes = TRUE
)

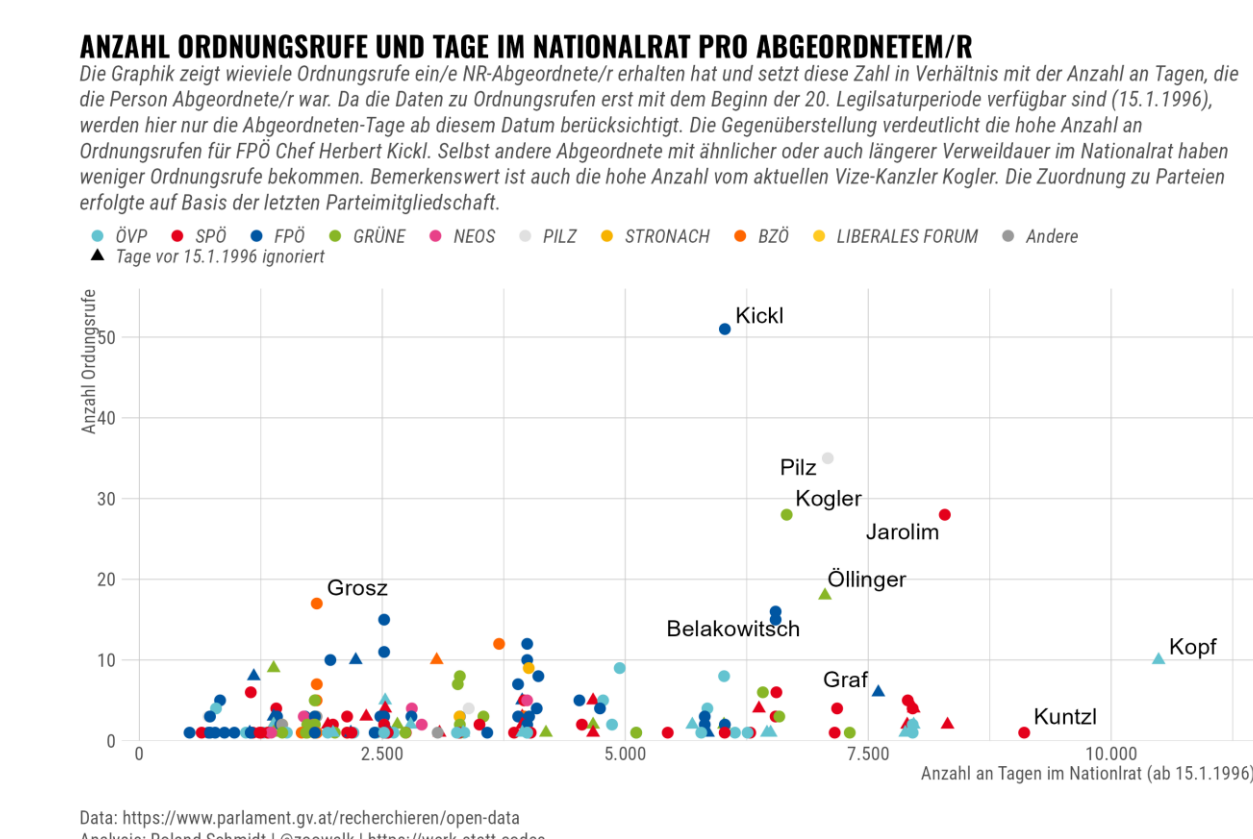
#count of votes list column
example$votes
#> [1]
#> #> text code color fraction in favor
#> 1 SPO 5 8790000 71 TRUE
#> 2 ÖVP 5 4000000 52 TRUE
#> 3 F 8 8005278 41 FALSE
#> 4 L 1 8000000 9 FALSE
#> 5 GRÜNE 0 8000000 9 FALSE
```

Austrian National Council: Party agreement matrix



Who gets Calls to Order ('Ordnungsrufe')?

```
df_calls <- ParlAT::get_items(institution = "Nationalrat", item = "GO", doc_type = "GO00") %>%
  as_tibble() %>%
  mutate(date = lubridate::dmy(date)) %>%
  arrange(date) %>%
  mutate(call_id = row_number(), before = 1)
#> [1] 766
```



Who spoke during a specific plenary meeting?

```
# Via URL - speaker list with timing information
result <- get_plenary_meeting_details(
  url = "https://www.parlament.gv.at/gegenstand/XXVIII/BR/SITZ/50",
  details_on = "speakers"
)
dplyr::glims(result)
#> Rows: 162
#> Columns: 21
#> $ meeting_url
#> $ meeting_title
#> $ meeting_citation
#> $ legis_period
#> $ meeting_type
#> $ debate_limit
#> $ debate_state
#> $ debate_type
#> $ debate_text
#> $ debate_text
#> $ debate_starttime
#> $ debate_endtime
#> $ debate_limit
#> $ debate_state
#> $ speech_nr
#> $ speech_state
#> $ speaker_name
#> $ pad_intern
#> $ sam_type
#> $ start_time
#> $ duration
#> $ speech_limit
```

Get transcripts of meetings

```
# Get transcript data for a specific period of time.
result <- get_transcripts(meeting_type = "BR/SITZ",
  date_start = "01-01-2024",
  date_end = "30-06-2024",
  echo = TRUE
)
#> ["NBVS":["BR/SITZ"],"DATUM":["2024-01-01T09:00:00.000Z","2024-06-30T00:00:00.000Z"]]
#> https://www.parlament.gv.at/recherchieren/protokolle/index.html?STENO_211NBVS=BR/SITZ
#> [1] 6

dplyr::glims(result)
#> Rows: 6
#> Columns: 8
#> $ date
#> $ meeting_url
#> $ legis_period
#> $ meeting_type
#> $ meeting_number
#> $ meeting
#> $ meeting_transcript_html
#> $ meeting_transcript_pdf
```

Which mandates did an active or previous MP have?

```
result <- get_mandates(c("Elisabeth Götzke", "Sebastian Kurz"))
dplyr::glims(result)
#> Rows: 10
#> Columns: 16
#> $ pad_intern
#> $ name
#> $ position_text
#> $ position_code
#> $ position_name
#> $ position_date_start
#> $ position_date_end
#> $ position_active
#> $ parl_group
#> $ wahlkreis
#> $ party
#> $ party_name
#> $ electoral_district_region_code
#> $ electoral_district_region
#> $ legis_period
#> $ url_biography
```

Download & further details

Package website: <https://werkstattcodes.github.io/ParlAT>

GitHub repo: <https://github.com/werkstattcodes/ParlAT>

Blog: <https://werkstattcodes.com>

Citation: Schmidt R (2026). ParlAT: A wrapper for the API of the Austrian Parliament. R package version 0.0.5.