

# Package: **tinytiger** (via r-universe)

July 4, 2024

**Title** Lightweight Interface to TIGER/Line Shapefiles

**Version** 0.0.9

**Description** Download geographic shapes from the United States Census Bureau TIGER/Line Shapefiles  
<<https://www.census.gov/geographies/mapping-files/time-series/geo/tiger-line-file.html>>.  
Functions support downloading and reading in geographic boundary data. All downloads can be set up with a cache to avoid multiple downloads. Data is available back to 2000 for most geographies.

**License** MIT + file LICENSE

**Depends** R (>= 2.10)

**Imports** cli, glue, curl, sf

**Suggests** knitr, rappdirs, rmarkdown, testthat (>= 3.0.0)

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.1

**LazyData** true

**URL** <https://github.com/alarm-redis/tinytiger>,  
<https://alarm-redis.org/tinytiger/>

**BugReports** <https://github.com/alarm-redis/tinytiger/issues>

**Config/testthat/edition** 3

**VignetteBuilder** knitr

**Repository** <https://christopherkenny.r-universe.dev>

**RemoteUrl** <https://github.com/alarm-redis/tinytiger>

**RemoteRef** HEAD

**RemoteSha** fc91b8e21f434a680f3f6545692ea826c1ca13bb

## Contents

county_fips_2020 . . . . .	3
tt_address_ranges . . . . .	3
tt_ai_an_nh_areas . . . . .	4
tt_anrc . . . . .	4
tt_area_landmarks . . . . .	5
tt_area_water . . . . .	5
tt_blocks . . . . .	6
tt_block_groups . . . . .	6
tt_cache_size . . . . .	7
tt_cbsa . . . . .	8
tt_coastline . . . . .	8
tt_congressional_districts . . . . .	9
tt_consolidated_cities . . . . .	9
tt_counties . . . . .	10
tt_county_subdivisions . . . . .	11
tt_csa . . . . .	11
tt_elementary_school_districts . . . . .	12
tt_estates . . . . .	12
tt_linear_water . . . . .	13
tt_metropolitan_divisions . . . . .	14
tt_military . . . . .	14
tt_new_england_cities . . . . .	15
tt_new_england_city_divisions . . . . .	15
tt_new_england_combined_areas . . . . .	16
tt_places . . . . .	16
tt_point_landmarks . . . . .	17
tt_polygon_edges . . . . .	17
tt_polygon_faces . . . . .	18
tt_primary_roads . . . . .	19
tt_primary_secondary_roads . . . . .	19
tt_puma . . . . .	20
tt_rails . . . . .	20
tt_roads . . . . .	21
tt_secondary_school_districts . . . . .	21
tt_states . . . . .	22
tt_state_leg_lower . . . . .	23
tt_state_leg_upper . . . . .	23
tt_subbarrios . . . . .	24
tt_tracts . . . . .	24
tt_tribal_block_groups . . . . .	25
tt_tribal_subdivisions . . . . .	25
tt_tribal_tracts . . . . .	26
tt_uac . . . . .	27
tt_unified_school_districts . . . . .	27
tt_voting_districts . . . . .	28
tt_zcta . . . . .	28

*county\_fips\_2020*

3

## Index

**30**

---

*county\_fips\_2020*      *Counties FIPS 2020*

---

### Description

Contains three columns:

- state: state FIPS
- county: county FIPS
- name: county name

### Usage

```
data("county_fips_2020")
```

### Value

tibble

---

*tt\_address\_ranges*      *Download TIGER shapes for Address Ranges*

---

### Description

Download TIGER shapes for Address Ranges

### Usage

```
tt_address_ranges(state, county, year = 2023)
```

### Arguments

state            FIPS, postal codes, or full names of states.  
county          FIPS codes or full names of counties. Optional.  
year            Integer year. Required. 2000 and 2010-2022 are currently supported.

### Value

sf data.frame

### Examples

```
# Wrapped in try due to false positive 304 errors  
try(tt_address_ranges("DE", county = "001")) # downloads slow on CRAN
```

---

tt_ai_an_nh_areas	<i>Download TIGER shapes for American Indian / Alaska Native / Native Hawaiian Areas</i>
-------------------	--

---

**Description**

Download TIGER shapes for American Indian / Alaska Native / Native Hawaiian Areas

**Usage**

```
tt_ai_an_nh_areas(year = 2023)
```

**Arguments**

year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_ai_an_nh_areas())
```

---

tt_anrc	<i>Download TIGER shapes for Alaska Native Regional Corporation (Alaska)</i>
---------	--

---

**Description**

Download TIGER shapes for Alaska Native Regional Corporation (Alaska)

**Usage**

```
tt_anrc(year = 2023)
```

**Arguments**

year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_anrc()) # downloads slow on CRAN
```

---

tt\_area\_landmarks      *Download TIGER shapes for Area Landmarks*

---

**Description**

Download TIGER shapes for Area Landmarks

**Usage**

```
tt_area_landmarks(state, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_area_landmarks("DE")) # downloads slow on CRAN
```

---

tt\_area\_water      *Download TIGER Shapes for Area Water*

---

**Description**

Download TIGER Shapes for Area Water

**Usage**

```
tt_area_water(state, county, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
county	FIPS codes or full names of counties. Optional.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_area_water("DE")) # downloads slow on CRAN
```

---

tt_blocks	<i>Download TIGER shapes for blocks</i>
-----------	---

---

**Description**

Download TIGER shapes for blocks

**Usage**

```
tt_blocks(state, county, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
county	FIPS codes or full names of counties. Optional.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_blocks(state = "DE", county = "001"))
```

---

tt_block_groups	<i>Download TIGER shapes for block groups</i>
-----------------	---

---

**Description**

Download TIGER shapes for block groups

**Usage**

```
tt_block_groups(state, county, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
county	FIPS codes or full names of counties. Optional.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_block_groups(state = "DE", county = "001")) # downloads slow on CRAN
```

---

tt\_cache\_size

*Work with the the tinytiger cache*

---

**Description**

Functions to inspect and clear the cache. If the cache is not enabled, uses a temporary directory.

**Usage**

```
tt_cache_size()
```

```
tt_cache_clear(force = FALSE)
```

```
tt_cache_path()
```

**Arguments**

**force** FALSE by default. Asks the user to confirm if interactive. Does not clear cache if force is FALSE and not interactive.

**Value**

For `tt_cache_size()`, the size in bytes, invisibly

For `tt_cache_clear()`, the path to the cache, invisibly.

For `tt_cache_path()`, the path to the cache

**Examples**

```
tt_cache_size()
```

```
tt_cache_clear()
```

```
tt_cache_path()
```

---

`tt_cbsa`*Download TIGER shapes for Core Based Statistical Areas*

---

**Description**

Download TIGER shapes for Core Based Statistical Areas

**Usage**

```
tt_cbsa(year = 2021)
```

**Arguments**

`year` Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# takes > 5 seconds  
# Wrapped in try due to false positive 304 errors  
try(tt_cbsa()) # downloads slow on CRAN
```

---

`tt_coastline`*Download TIGER shapes for Coastlines*

---

**Description**

Download TIGER shapes for Coastlines

**Usage**

```
tt_coastline(year = 2023)
```

**Arguments**

`year` Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame



**Examples**

```
# takes > 5 seconds
# Wrapped in try due to false positive 304 errors
try(tt_coastline()) # downloads slow on CRAN
```

---

```
tt_congressional_districts
```

*Download TIGER shapes for congressional districts*

---

**Description**

Download TIGER shapes for congressional districts

**Usage**

```
tt_congressional_districts(state, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# takes > 5 seconds
# Wrapped in try due to false positive 304 errors
try(tt_congressional_districts()) # downloads slow on CRAN
```

---

```
tt_consolidated_cities
```

*Download TIGER shapes for Public Use Microdata Areas*

---

**Description**

Download TIGER shapes for Public Use Microdata Areas

**Usage**

```
tt_consolidated_cities(state, year = 2023)
```

**Arguments**

state FIPS, postal codes, or full names of states.  
year Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_consolidated_cities("CT")) # downloads slow on CRAN
```

---

tt\_counties *Download TIGER shapes for counties*

---

**Description**

Download TIGER shapes for counties

**Usage**

```
tt_counties(state, year = 2023)
```

**Arguments**

state FIPS, postal codes, or full names of states.  
year Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# takes > 5 seconds  
# Wrapped in try due to false positive 304 errors  
try(tt_counties(state = "DE")) # downloads slow on CRAN
```

---

`tt_county_subdivisions`*Download TIGER shapes for County Subdivisions*

---

**Description**

Download TIGER shapes for County Subdivisions

**Usage**

```
tt_county_subdivisions(state, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_county_subdivisions("DE")) # downloads slow on CRAN
```

---

`tt_csa`*Download TIGER shapes for Combined Statistical Area*

---

**Description**

Download TIGER shapes for Combined Statistical Area

**Usage**

```
tt_csa(year = 2021)
```

**Arguments**

year	Integer year. Required. 2000 and 2010-2022 are currently supported.
------	---

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_csa())
```

---

```
tt_elementary_school_districts  
  Download TIGER shapes for Elementary School Districts
```

---

**Description**

Download TIGER shapes for Elementary School Districts

**Usage**

```
tt_elementary_school_districts(state, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_elementary_school_districts("AZ")) # downloads slow on CRAN
```

---

```
tt_estates  
  Download TIGER shapes for Estates (US Virgin Islands)
```

---

**Description**

Download TIGER shapes for Estates (US Virgin Islands)

**Usage**

```
tt_estates(year = 2023)
```

**Arguments**

year	Integer year. Required. 2000 and 2010-2022 are currently supported.
------	---

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_estates()) # downloads slow on CRAN
```

---

tt_linear_water	<i>Download TIGER Shapes for Linear Water</i>
-----------------	---

---

**Description**

Download TIGER Shapes for Linear Water

**Usage**

```
tt_linear_water(state, county, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
county	FIPS codes or full names of counties. Optional.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_linear_water("DE")) # downloads slow on CRAN
```

---

tt\_metropolitan\_divisions

*Download TIGER shapes for Metropolitan Divisions*

---

**Description**

Download TIGER shapes for Metropolitan Divisions

**Usage**

```
tt_metropolitan_divisions(year = 2021)
```

**Arguments**

year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_metropolitan_divisions()) # downloads slow on CRAN
```

---

tt\_military

*Download TIGER shapes for Military Installations*

---

**Description**

Download TIGER shapes for Military Installations

**Usage**

```
tt_military(year = 2023)
```

**Arguments**

year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_military()) # downloads slow on CRAN
```

---

tt\_new\_england\_cities *Download TIGER shapes for New England City and Town Area*

---

**Description**

Download TIGER shapes for New England City and Town Area

**Usage**

```
tt_new_england_cities(year = 2023)
```

**Arguments**

year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_new_england_cities()) # downloads slow on CRAN
```

---

tt\_new\_england\_city\_divisions  
*Download TIGER shapes for New England City and Town Area Divisions*

---

**Description**

Download TIGER shapes for New England City and Town Area Divisions

**Usage**

```
tt_new_england_city_divisions(year = 2021)
```

**Arguments**

year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_new_england_city_divisions()) # downloads slow on CRAN
```

---

```
tt_new_england_combined_areas
```

*Download TIGER shapes for New England Combined City and Town Areas*

---

**Description**

Download TIGER shapes for New England Combined City and Town Areas

**Usage**

```
tt_new_england_combined_areas(year = 2021)
```

**Arguments**

year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_new_england_combined_areas()) # downloads slow on CRAN
```

---

```
tt_places
```

*Download TIGER shapes for Places*

---

**Description**

Download TIGER shapes for Places

**Usage**

```
tt_places(state, year = 2023)
```

**Arguments**

state                    FIPS, postal codes, or full names of states.  
year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame



**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_places("DE")) # downloads slow on CRAN
```

---

tt\_point\_landmarks      *Download TIGER shapes for Point Landmarks*

---

**Description**

Download TIGER shapes for Point Landmarks

**Usage**

```
tt_point_landmarks(state, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_point_landmarks("DE")) # downloads slow on CRAN
```

---

tt\_polygon\_edges      *Download TIGER shapes for Polygon Edges*

---

**Description**

Download TIGER shapes for Polygon Edges

**Usage**

```
tt_polygon_edges(state, county, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
county	FIPS codes or full names of counties. Optional.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_polygon_edges("DE", county = "001")) # downloads slow on CRAN
```

---

tt_polygon_faces	<i>Download TIGER shapes for Polygon Faces</i>
------------------	--

---

**Description**

Download TIGER shapes for Polygon Faces

**Usage**

```
tt_polygon_faces(state, county, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
county	FIPS codes or full names of counties. Optional.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_polygon_faces("DE", county = "001")) # downloads slow on CRAN
```

---

tt\_primary\_roads      *Download TIGER shapes for Primary Roads*

---

**Description**

Download TIGER shapes for Primary Roads

**Usage**

```
tt_primary_roads(year = 2023)
```

**Arguments**

year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# takes > 5 seconds
# Wrapped in try due to false positive 304 errors
try(tt_primary_roads()) # downloads slow on CRAN
```

---

tt\_primary\_secondary\_roads      *Download TIGER shapes for Primary and Secondary Roads*

---

**Description**

Download TIGER shapes for Primary and Secondary Roads

**Usage**

```
tt_primary_secondary_roads(state, year = 2023)
```

**Arguments**

state                    FIPS, postal codes, or full names of states.  
year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_primary_secondary_roads("DE")) # downloads slow on CRAN
```

---

tt_puma	<i>Download TIGER shapes for Public Use Microdata Areas</i>
---------	---

---

**Description**

Download TIGER shapes for Public Use Microdata Areas

**Usage**

```
tt_puma(state, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_puma("DE")) # downloads slow on CRAN
```

---

tt_rails	<i>Download TIGER shapes for Rails</i>
----------	--

---

**Description**

Download TIGER shapes for Rails

**Usage**

```
tt_rails(year = 2023)
```

**Arguments**

year	Integer year. Required. 2000 and 2010-2022 are currently supported.
------	---

**Value**

sf data.frame

**Examples**

```
# takes > 5 seconds
# Wrapped in try due to false positive 304 errors
try(tt_rails()) # downloads slow on CRAN
```

---

tt_roads	<i>Download TIGER shapes for Roads</i>
----------	--

---

**Description**

Download TIGER shapes for Roads

**Usage**

```
tt_roads(state, county, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
county	FIPS codes or full names of counties. Optional.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_roads("DE")) # downloads slow on CRAN
```

---

tt_secondary_school_districts	<i>Download TIGER shapes for Secondary School Districts</i>
-------------------------------	---

---

**Description**

Download TIGER shapes for Secondary School Districts

**Usage**

```
tt_secondary_school_districts(state, year = 2023)
```

**Arguments**

state FIPS, postal codes, or full names of states.  
year Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_secondary_school_districts("AZ")) # downloads slow on CRAN
```

---

tt\_states *Download TIGER shapes for states*

---

**Description**

Download TIGER shapes for states

**Usage**

```
tt_states(year = 2023)
```

**Arguments**

year Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_states())
```

---

tt\_state\_leg\_lower      *Download TIGER shapes for lower state legislative districts*

---

**Description**

Download TIGER shapes for lower state legislative districts

**Usage**

```
tt_state_leg_lower(state, year = 2023)
```

**Arguments**

state                    FIPS, postal codes, or full names of states.  
year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_state_leg_lower("DE")) # downloads slow on CRAN
```

---

tt\_state\_leg\_upper      *Download TIGER shapes for upper state legislative districts*

---

**Description**

Download TIGER shapes for upper state legislative districts

**Usage**

```
tt_state_leg_upper(state, year = 2023)
```

**Arguments**

state                    FIPS, postal codes, or full names of states.  
year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_state_leg_lower("DE")) # downloads slow on CRAN
```

---

tt_subbarrios	<i>Download TIGER shapes for Subbarrios (Puerto Rico)</i>
---------------	---

---

**Description**

Download TIGER shapes for Subbarrios (Puerto Rico)

**Usage**

```
tt_subbarrios(year = 2023)
```

**Arguments**

year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_subbarrios()) # downloads slow on CRAN
```

---

tt_tracts	<i>Download TIGER shapes for tracts</i>
-----------	---

---

**Description**

Download TIGER shapes for tracts

**Usage**

```
tt_tracts(state, county, year = 2023)
```

**Arguments**

state                    FIPS, postal codes, or full names of states.  
 county                   FIPS codes or full names of counties. Optional.  
 year                    Integer year. Required. 2000 and 2010-2022 are currently supported.



**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_tracts(state = "DE", county = "001")) # downloads slow on CRAN
```

---

```
tt_tribal_block_groups
```

*Download TIGER shapes for Tribal Block Groups*

---

**Description**

Download TIGER shapes for Tribal Block Groups

**Usage**

```
tt_tribal_block_groups(year = 2023)
```

**Arguments**

year                    Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_tribal_block_groups())
```

---

```
tt_tribal_subdivisions
```

*Download TIGER shapes for American Indian Tribal Subdivision National*

---

**Description**

Download TIGER shapes for American Indian Tribal Subdivision National

**Usage**

```
tt_tribal_subdivisions(year = 2023)
```

**Arguments**

year Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_tribal_subdivisions())
```

---

tt\_tribal\_tracts *Download TIGER shapes for Tribal Tracts*

---

**Description**

Download TIGER shapes for Tribal Tracts

**Usage**

```
tt_tribal_tracts(year = 2023)
```

**Arguments**

year Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors  
try(tt_tribal_tracts())
```

---

tt\_uac *Download TIGER shapes for Urban Area*

---

**Description**

Download TIGER shapes for Urban Area

**Usage**

```
tt_uac(year = 2023)
```

**Arguments**

year Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# takes > 5 seconds
# Wrapped in try due to false positive 304 errors
try(tt_uac()) # downloads slow on CRAN
```

---

tt\_unified\_school\_districts  
*Download TIGER shapes for Unified School Districts*

---

**Description**

Download TIGER shapes for Unified School Districts

**Usage**

```
tt_unified_school_districts(state, year = 2023)
```

**Arguments**

state FIPS, postal codes, or full names of states.  
year Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_unified_school_districts("DE")) # downloads slow on CRAN
```

---

tt\_voting\_districts     *Download TIGER shapes for Voting Districts*

---

**Description**

Download TIGER shapes for Voting Districts

**Usage**

```
tt_voting_districts(state, county, year = 2023)
```

**Arguments**

state	FIPS, postal codes, or full names of states.
county	FIPS codes or full names of counties. Optional.
year	Integer year. Required. 2000 and 2010-2022 are currently supported.

**Value**

sf data.frame

**Examples**

```
# Wrapped in try due to false positive 304 errors
try(tt_voting_districts("DE", county = "001")) # downloads slow on CRAN
```

---

tt\_zcta     *Download TIGER shapes for Zip Code Tabulation Areas*

---

**Description**

Download TIGER shapes for Zip Code Tabulation Areas

**Usage**

```
tt_zcta(year = 2023)
```

**Arguments**

year	Integer year. Required. 2000 and 2010-2022 are currently supported.
------	---

**Value**

sf data.frame

**Examples**

```
# takes > 5 seconds
# Wrapped in try due to false positive 304 errors
try(tt_zcta()) # downloads slow on CRAN
```

# Index

- \* **cities**
    - tt\_consolidated\_cities, 9
    - tt\_county\_subdivisions, 11
    - tt\_places, 16
    - tt\_puma, 20
  - \* **city**
    - tt\_new\_england\_cities, 15
    - tt\_new\_england\_city\_divisions, 15
    - tt\_new\_england\_combined\_areas, 16
  - \* **data**
    - county\_fips\_2020, 3
  - \* **districts**
    - tt\_congressional\_districts, 9
    - tt\_state\_leg\_lower, 23
    - tt\_state\_leg\_upper, 23
    - tt\_voting\_districts, 28
  - \* **landmarks**
    - tt\_area\_landmarks, 5
    - tt\_point\_landmarks, 17
  - \* **other**
    - tt\_address\_ranges, 3
    - tt\_cache\_size, 7
    - tt\_cbsa, 8
    - tt\_csa, 11
    - tt\_estates, 12
    - tt\_metropolitan\_divisions, 14
    - tt\_military, 14
    - tt\_polygon\_edges, 17
    - tt\_polygon\_faces, 18
    - tt\_subbarrios, 24
    - tt\_uac, 27
    - tt\_zcta, 28
  - \* **roads**
    - tt\_primary\_roads, 19
    - tt\_primary\_secondary\_roads, 19
    - tt\_rails, 20
    - tt\_roads, 21
  - \* **schools**
    - tt\_elementary\_school\_districts, 12
    - tt\_secondary\_school\_districts, 21
    - tt\_unified\_school\_districts, 27
  - \* **spine**
    - tt\_block\_groups, 6
    - tt\_blocks, 6
    - tt\_counties, 10
    - tt\_states, 22
    - tt\_tracts, 24
  - \* **tribal**
    - tt\_ai\_an\_nh\_areas, 4
    - tt\_anrc, 4
    - tt\_tribal\_block\_groups, 25
    - tt\_tribal\_subdivisions, 25
    - tt\_tribal\_tracts, 26
  - \* **water**
    - tt\_area\_water, 5
    - tt\_coastline, 8
    - tt\_linear\_water, 13
- county\_fips\_2020, 3
- tt\_address\_ranges, 3
- tt\_ai\_an\_nh\_areas, 4
- tt\_anrc, 4
- tt\_area\_landmarks, 5
- tt\_area\_water, 5
- tt\_block\_groups, 6
- tt\_blocks, 6
- tt\_cache\_clear (tt\_cache\_size), 7
- tt\_cache\_path (tt\_cache\_size), 7
- tt\_cache\_size, 7
- tt\_cbsa, 8
- tt\_coastline, 8
- tt\_congressional\_districts, 9
- tt\_consolidated\_cities, 9
- tt\_counties, 10
- tt\_county\_subdivisions, 11
- tt\_csa, 11
- tt\_elementary\_school\_districts, 12
- tt\_estates, 12

tt\_linear\_water, 13  
tt\_metropolitan\_divisions, 14  
tt\_military, 14  
tt\_new\_england\_cities, 15  
tt\_new\_england\_city\_divisions, 15  
tt\_new\_england\_combined\_areas, 16  
tt\_places, 16  
tt\_point\_landmarks, 17  
tt\_polygon\_edges, 17  
tt\_polygon\_faces, 18  
tt\_primary\_roads, 19  
tt\_primary\_secondary\_roads, 19  
tt\_puma, 20  
tt\_rails, 20  
tt\_roads, 21  
tt\_secondary\_school\_districts, 21  
tt\_state\_leg\_lower, 23  
tt\_state\_leg\_upper, 23  
tt\_states, 22  
tt\_subbarrios, 24  
tt\_tracts, 24  
tt\_tribal\_block\_groups, 25  
tt\_tribal\_subdivisions, 25  
tt\_tribal\_tracts, 26  
tt\_uac, 27  
tt\_unified\_school\_districts, 27  
tt\_voting\_districts, 28  
tt\_zcta, 28