

# Package: crashapi (via r-universe)

November 5, 2024

**Type** Package

**Title** CrashAPI

**Version** 0.1.2

**Description** Get Fatality Analysis Reporting System (FARS) data with the FARS API from the U.S. National Highway Traffic Safety Administration (NHTSA).

**License** MIT + file LICENSE

**URL** <https://github.com/elipousson/crashapi>,  
<https://elipousson.github.io/crashapi/>

**BugReports** <https://github.com/elipousson/crashapi/issues>

**Depends** R (>= 2.10)

**Imports** cli, dplyr (>= 1.1.0), glue, httr2 (>= 0.2.3), janitor, rlang, stats, stringr, utils, vctrs

**Suggests** covr, httptest2, knitr, readr, rmarkdown, sf, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Config/testthat/parallel** true

**Encoding** UTF-8

**LazyData** true

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

**Config/pak/sysreqs** libicu-dev libssl-dev

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

**RemoteUrl** <https://github.com/elipousson/crashapi>

**RemoteRef** HEAD

**RemoteSha** d19acf8a301543d65f5107bbd354078d6db34f09

## Contents

crash_data_index . . . . .	2
fars_terms . . . . .	3
fars_vars . . . . .	4
fars_vars_labels . . . . .	5
format_crashes . . . . .	6
get_crss_zip . . . . .	6
get_fars . . . . .	7
get_fars_crash_persons . . . . .	10
get_fars_crash_vehicles . . . . .	11
get_fars_zip . . . . .	12
mmucc_codes . . . . .	13
read_crashapi . . . . .	14
<b>Index</b>	<b>15</b>

---

crash_data_index	<i>U.S. vehicular crash data index (city, county, regional, and state)</i>
------------------	--

---

## Description

This index include identified data from cities, counties, or regional entities in 43 of 50 U.S. states. Statewide data sources are included from 33 states. In 4 states (NE, OH, ME, and CT), the only identified statewide data sources allow limited access through a web-based public query form. In 1 state (MN), data only available through restricted access mapping/query tool. Not all statewide data sources include all crashes (some include only cyclist/pedestrian crashes or fatal crashes) and the structure and format of the crash data provided varies considerably.

## Usage

crash\_data\_index

## Format

A data frame with 75 rows and 22 variables:

name Name of data set from provider.

level Geographic scope/level (e.g. city, county, region, state, national)

city City name

county County name

region logical COLUMN\_DESCRIPTION

state\_name U.S. state name

state\_abb U.S. state abbreviation

info\_url Informational URL (e.g. informational page about file download options)

data\_url Data URL (e.g. direct link to ArcGIS FeatureServer layer)

format Data format (e.g. Socrata, CKAN, ArcGIS MapServer, etc.)  
 statewide\_yn Yes for data with statewide geographic scope; NA for data from city, county, or regional level providers  
 batch\_download\_yn Yes for data where batch download is possible  
 start\_year Earliest year for crashes in dataset  
 end\_year Latest year for crashes in dataset  
 publisher Agency/organization responsible for publishing the data online  
 description Description of the dataset from provider  
 bike\_ped\_only Yes for data that only includes bike/ped involved crashes (common for Vision Zero programs)  
 rolling\_window Description of rolling time window if data is only available within a rolling window  
 fatal\_severe\_only Yes for data that only includes fatal/severe crashes (common for Vision Zero programs)  
 date\_note Note on the dates for the crash data  
 updates Information on update schedule if available  
 note General notes

## Details

This index was compiled by Eli Pousson between October 2021 and February 2022 with additional contributions from Mae Hanzlik.

Added: March 27 2022 Updated: June 05 2022

Corrections, updates, or additional sources should be added to this public Google Sheet: <https://docs.google.com/spreadsheets/d/1rmn6GbHNkfWLLDEEmA87iuy2yHdh7hBybCTZiQJEY0k/edit?usp=sharing>

---

fars\_terms

*NHSTA Terms and Definitions*

---

## Description

FARS-related terms defined by the National Highway Traffic Safety Administration based on ANSI D16.1-1996: Manual on Classification of Motor Vehicle Traffic Accidents.

## Usage

fars\_terms

## Format

A data frame with 66 rows and 2 variables:

term character Term

definition character Term definition

**Details**

Added: October 25 2021 Updated: October 25 2021

**Source**

[NHTSA FARS Terms](#)

---

fars_vars	<i>Get variables and variable attributes for the Fatality Analysis Reporting System (FARS) API</i>
-----------	--

---

**Description**

By default, this function returns the returns the list of variables for the data year specified. If vars is "make", "model", or "bodytype", the function returns #' list of variable attributes for the specified variable name or attributes for make model and body type specified in the FARS dataset.

**Usage**

```
fars_vars(year, var = NULL, make = NULL, model = NULL)
```

**Arguments**

year	Case year. Year must be between 2010 and 2019.
var	Default NULL. Supported values are "make", "model", and "bodytype". Using the var parameter returns variable attributes for the specified variable name or attributes for make model and body type specified in the dataset.
make	Integer. Make ID number. Required to return variables for "model" and "bodytype". Get a list of make ID numbers using the "make" var for the selected year, e.g. <code>fars_vars(year = 2010, var = "make")</code> .
model	Integer. Model ID number. Required to return variables for "bodytype". Get a list of model ID numbers using the "model" var for the selected year with a valid make ID number, e.g. <code>fars_vars(year = 2010, var = "model", make = 37)</code>

**Examples**

```
head(fars_vars(year = 2022, var = "make"), 5)

head(fars_vars(year = 2022, var = "model", make = 12), 5)

fars_vars(year = 2022, var = "bodytype", make = 12, model = 37)
```

---

fars_vars_labels	<i>FARS variable names and labels</i>
------------------	---------------------------------------

---

### Description

A table of FARS table variable names extracted from the Fatality Analysis Reporting System (FARS) Analytical User's Manual, 1975-2019, documentation of the SAS format data files.

### Usage

fars\_vars\_labels

### Format

A data frame with 498 rows and 14 variables:

name character Variable name

label character Variable label

order double Sort order

data\_file character SAS data file name

data\_file\_id double SAS data file ID

file\_id character File ID

key logical Indicator for key variables

location double Location in SAS data file

mmuc\_equivalent logical Equivalent term in MMUC (placeholder)

discontinued logical Indicator for discontinued variables

api\_only logical Indicator for variables only used by API

api character Name(s) of corresponding CrashAPI service

name\_var logical Indicator for "NAME" variable returned by API

nm Short version of the variable name

api\_list\_col logical Indicator for list columns returned by API

### Details

Added: January 31 2022 Updated: March 27 2022

---

format_crashes	<i>Format crash data</i>
----------------	--------------------------

---

### Description

Reorder columns to match the order documented in Fatality Analysis Reporting System (FARS) Analytical User's Manual, 1975-2019 and append derived columns for date, time, and datetime.

### Usage

```
format_crashes(x, details = TRUE)
```

### Arguments

x	Data frame with crash data.
details	If TRUE, append date, time, datetime columns to formatted crash data; defaults to TRUE

---

get_crss_zip	<i>Download CRSS data files as zipped CSV or SAS files</i>
--------------	--

---

### Description

This function is similar to [get\\_fars\\_zip\(\)](#) to download files directly from NHTSA FTP site. If read is TRUE, the function reads a list containing data frames for each table available in the selected year. If geometry is TRUE, the accident table is converted to an sf object.

### Usage

```
get_crss_zip(
  year = 2022,
  format = "csv",
  path = NULL,
  aux = FALSE,
  read = TRUE,
  geometry = FALSE,
  overwrite = FALSE
)
```

**Arguments**

year	Year of data from 2016 to 2022, Default: 2022
format	Format of zipped data tables ('csv' or 'sas'). Default: 'csv'. unzip and geo options are only supported if format is "csv".
path	Path to download zip file. Set to <code>getwd()</code> if NULL (default).
aux	If TRUE, download auxiliary CRSS datasets .
read	If TRUE, unzip the downloaded file and read CSV files into a list of tables with each list item corresponding to one CSV file.
geometry	If TRUE, convert the accident table to a sf object.
overwrite	If FALSE, abort if file exists at the provided path. If TRUE, overwrite file.

**Value**

Downloads zip file with CSV or SAS tables and returns the zip file path invisibly or returns a list of data frames (if geo is FALSE), or returns a list of data frames with the accident table converted to a sf object.

---

get_fars	<i>Get Fatality Analysis Reporting System (FARS) data with the FARS API</i>
----------	---

---

**Description**

This function provides a convenient interface for accessing FARS data or data summaries using a range of criteria. The `api` parameter allows you to call one of the following functions to access DOT NHTSA's Crash API:

- `get_fars_crash_list` returns a list of fatal crashes that have occurred in multiple states in one or more years.
- `get_fars_crash_details` returns a details of a fatal crash that has occurred in a state for a single year.
- `get_fars_crashes` a list of fatal crashes by location that have occurred throughout U.S.
- `get_fars_summary` provides a count of injury severity that have occurred throughout U.S. including count of fatalities and crashes.
- `get_fars_year` provides one of 20 FARS data tables for a single year. Supports downloading to a CSV or JSON file.

Both `get_fars_crash_list` and `get_fars_crashes` limit the returned data to 5000 records so consider limiting the range of years requested if data exceeds that threshold.

This package also enables access to the FARS data available through the NHTSA data downloads server in a zip format. Set `api` to "zip" or use the `get_fars_zip` function to download this data.

**Usage**

```
get_fars(  
  year = 2022,  
  state,  
  county = NULL,  
  api = c("crashes", "cases", "state list", "summary count", "year dataset", "zip"),  
  type = NULL,  
  details = FALSE,  
  geometry = FALSE,  
  crs = NULL,  
  cases = NULL,  
  vehicles = NULL,  
  format = "json",  
  pr = FALSE,  
  path = NULL,  
  download = FALSE  
)  
  
get_fars_crashes(  
  year = 2022,  
  start_year,  
  end_year = NULL,  
  state,  
  county,  
  details = FALSE,  
  geometry = FALSE,  
  crs = NULL  
)  
  
get_fars_cases(  
  year = 2022,  
  state,  
  cases,  
  details = FALSE,  
  geometry = FALSE,  
  crs = NULL  
)  
  
get_fars_crash_list(  
  year = 2022,  
  start_year = NULL,  
  end_year = NULL,  
  state,  
  vehicles = c(1, 50)  
)  
  
get_fars_summary(year = 2022, start_year, end_year = NULL, state)
```



```

get_fars_year(
  year = 2022,
  type = "accident",
  state,
  format = "json",
  path = NULL,
  geometry = FALSE,
  crs = NULL,
  download = FALSE
)

```

### Arguments

year	numeric vector. Year or range with start and end year. If api is "details", "year dataset", or "zip" (or using the get_fars_crash_details, get_fars_year, or get_fars_zip functions), a single year is required. All other api options support a range with the minimum value is used as a start year and the maximum value used as an end year. Most api options support the years from 2010 through the most recent year of release. "year dataset" only supports 2010 to 2017 and "zip" supports 1975 to 2022. start_year and end_year are ignored if year is not NULL.
state	Required. State name, abbreviation, or FIPS number. get_fars_crash_list supports multiple states.
county	County name or FIPS number. Required for get_fars_crashes.
api	character. API function to use. Supported values include "crashes", "cases", "state list", "summary count", "year dataset", and "zip". Default: "crashes".
type	Name of the dataset or data file to download when using the "year dataset" api or get_fars_year. Supported values include "ACCIDENT", "CEVENT", "DAMAGE", "DISTRACT", "DRIMPAIR", "FACTOR", "MANEUVER", "NMCRAASH", "NMIMPAIR", "NMPRIOR", "PARKWORK", "PBTYPE", "PERSON", "SAFETYEQ", "VEHICLE", "VEVENT VINDECODE", "VINDERIVED", "VIOLATION", "VISION", and "VSOE". Lowercase or mixed case values are permitted.
details	Type of detailed crash data to return (either "events" or "vehicles"). If TRUE for get_fars or get_fars_crashes, detailed case data (excluding event and vehicle data) is attached to the returned crash data. If NULL for get_fars_cases, events and vehicle data are excluded from the returned case data. returned by get_fars_cases. Optional for get_fars_crash_details. Default: NULL for get_fars_cases; FALSE for get_fars and get_fars_crashes.
geometry	If TRUE, return sf object. Optional for get_fars_crashes.
crs	Coordinate reference system to return for get_fars_crashes if geometry is TRUE.
cases	One or more FARS case numbers. Required if api is "cases" or using get_fars_cases. Multiple case numbers can be provided.
vehicles	numeric vector with the minimum and maximum number of vehicles, e.g. c(1, 2) for minimum of 1 vehicle and maximum of 2. Required for get_fars_crash_list.

format	Default "json". "csv" is supported when using the "year dataset" api. "sas" is supporting for the "zip" api.
pr	logical. If TRUE, download zip file with FARS data for Puerto Rico. No Puerto Rico data available for years 1975-1977. Default: FALSE for get_fars_zip only.
path	File path used if download is TRUE.
download	logical. If TRUE and the api is "year dataset" or "zip", download the data to a file. Default FALSE.
start_year	Start year for crash reports.
end_year	End year for crash reports.

### Examples

```

head(get_fars_crashes(state = "MD", county = "Baltimore city"), 5)

get_fars_cases(state = "MD", cases = "240274")

get_fars_crash_list(state = "MD", vehicles = 5)

get_fars_summary(state = "MD")

head(get_fars_year(state = "MD", type = "PERSON"), 5)

```

---

```

get_fars_crash_persons
      Get Crashes By Occupant

```

---

### Description

This function returns a list of fatal crashes by occupant that have occurred throughout United States. This function is not currently working.

### Usage

```

get_fars_crash_persons(
  year = NULL,
  start_year,
  end_year = NULL,
  state,
  age = NULL,
  sex = NULL,
  seat,
  injury,
  occupants = TRUE,
  nonoccupants = TRUE
)

```

**Arguments**

year	numeric vector. Year or range with start and end year. 2010 to 2022 supported.
start_year	Start year for crash reports.
end_year	End year for crash reports.
state	Required. State name, abbreviation, or FIPS number.
age	numeric
sex	Options "m", "f", "male", "female", "unknown", "not reported."
seat	Seat position
injury	Options "unknown", "not reported", "died prior", "injured", "fatal", "suspected serious", "suspected minor", "possible", "no apparent"
occupants	Include vehicle occupants in query; defaults to TRUE
nonoccupants	Include non-occupants in query; defaults to TRUE

---

 get\_fars\_crash\_vehicles

*Get Crashes By Vehicle*


---

**Description**

This function returns a list of fatal crashes by vehicle type that have occurred throughout United States. The make, model, and body type must match the options returned by fars\_vars. This function accepts named options that are converted to ID numbers for use in the API query.

**Usage**

```
get_fars_crash_vehicles(
  year = NULL,
  start_year,
  end_year = NULL,
  state,
  make = NULL,
  model = NULL,
  model_year = 2010,
  body_type = NULL
)
```

**Arguments**

year	numeric vector. Year or range with start and end year. 2010 to 2022 supported.
start_year	Start year for crash reports.
end_year	End year for crash reports.
state	Required. State name, abbreviation, or FIPS number.

make	Make name or ID, Required. The start_year is used to return a list of support make options. Default: NULL
model	Model name or ID, Optional. Default: NULL
model_year	Model year, Optional. Default: NULL
body_type	Body type, Optional. model must be provided to use body_type parameter. Default: NULL

---

get\_fars\_zip

*Download FARS data files as zipped CSV or SAS files*


---

### Description

This function provides an alternative to `get_fars_year()` that downloads files directly from NHTSA FTP site. If `read` is TRUE, the function reads a list containing data frames for each table available in the selected year. If `geometry` is TRUE, the accident table is converted to an sf object.

### Usage

```
get_fars_zip(
  year = 2022,
  format = "csv",
  path = NULL,
  pr = FALSE,
  aux = FALSE,
  read = TRUE,
  geometry = FALSE,
  overwrite = FALSE
)
```

### Arguments

year	Year of data from 1975 to 2022, Default: 2022
format	Format of zipped data tables ('csv' or 'sas'). Default: 'csv'. unzip and geo options are only supported if format is "csv".
path	Path to download zip file. Set to <code>getwd()</code> if NULL (default).
pr	If TRUE, download FARS data for Puerto Rico. No Puerto Rico data available for years 1975-1977. Default: FALSE
aux	If TRUE and year is after 1982, download auxiliary FARS datasets that "contain data derived from the regular FARS/GES variables using NCSA analytical data classifications." In 2010, the NHTSA explained: "These classifications are widely used in NCSA publications and research. Many definitions such as "speeding-related" or "distracted driving" comprise a combination of variables whose names or attributes have changed over time. The derived variables in the auxiliary files incorporate these nuances, thus simplifying the use of standard classifications in any traffic safety research." Learn more from the FARS and GES Auxiliary Datasets Q & A: <a href="https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/811364">https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/811364</a>

read	If TRUE, unzip the downloaded file and read CSV files into a list of tables with each list item corresponding to one CSV file.
geometry	If TRUE, convert the accident table to a sf object.
overwrite	If FALSE, abort if file exists at the provided path. If TRUE, overwrite file.

### Value

Downloads zip file with CSV or SAS tables and returns the zip file path invisibly or returns a list of data frames (if geometry is FALSE), or returns a list of data frames with the accident table converted to a sf object.

---

 mmucc\_codes

---

*Model Minimum Uniform Crash Criteria (MMUCC) codes (simple)*


---

### Description

A collection of the 73 unique codes identified as simple codes: <https://release.niem.gov/niem/codes/mmucc/4.1/mmucc.xsd>

See the MMUCC Guideline Fifth Edition (2017) for more information: <https://crashstats.nhtsa.dot.gov/Api/Public/Publication/812433>

About MMUC from NHTSA: <https://www.nhtsa.gov/mmucc-1>

To encourage greater uniformity, the National Highway Traffic Safety Administration (NHTSA) and the Governors Highway Safety Association (GHSA) cooperatively developed a voluntary data collection guideline in 1998. The MMUCC guideline identifies a minimum set of motor vehicle crash data elements and their attributes that States should consider collecting and including in their State crash data system.

The MMUCC 5th Edition is the result of an 18-month collaboration between NHTSA, the Federal Highway Administration (FHWA), the Federal Motor Carrier Safety Administration (FMCSA), the National Transportation Safety Board (NTSB), the GHSA, and subject matter experts from State DOTs, local law enforcement, emergency medical services, safety organizations, industry partners, and academia. The traffic records community and general public also contributed through external forums (Federal Register) and at the 2016 Traffic Records Forum.

- Added: March 26 2022
- Updated: March 26 2022

### Usage

mmucc\_codes

**Format**

A data frame with 700 rows and 6 variables:

```
code Attribute code
name Attribute code name
type Attribute code type
definition Code definition
restriction_id Restriction id number
restriction Restriction value
```

---

read_crashapi	<i>Read data from the CrashAPI using a url template</i>
---------------	---

---

**Description**

An updated utility function using the `httr2` package to read data from the CrashAPI using the API URL templates listed on the NHSTA website: <https://crashviewer.nhtsa.dot.gov/CrashAPI>

**Usage**

```
read_crashapi(
  url = "https://crashviewer.nhtsa.dot.gov",
  data = "crashes",
  type = NULL,
  format = "json",
  results = TRUE,
  ...,
  call = caller_env()
)
```

**Arguments**

<code>url</code>	Base url for CrashAPI.
<code>data</code>	Data (crashes, analytics, or fars), Default: 'crashes'
<code>type</code>	Type of API to use, Default: NULL
<code>format</code>	Format to return, Default: 'json'
<code>results</code>	If FALSE, return formatted url, Default: TRUE
<code>...</code>	Additional parameters used in template (varies by type).
<code>call</code>	The execution environment of a currently running function, e.g. <code>caller_env()</code> . The function will be mentioned in error messages as the source of the error. See the <code>call</code> argument of <code>abort()</code> for more information.

**Value**

Data frame with requested data or a formatted url (if `results = FALSE`)

# Index

## \* datasets

- crash\_data\_index, 2
- fars\_terms, 3
- fars\_vars\_labels, 5
- mmucc\_codes, 13

abort(), 14

crash\_data\_index, 2

fars\_terms, 3

fars\_vars, 4

fars\_vars\_labels, 5

format\_crashes, 6

get\_crss\_zip, 6

get\_fars, 7

get\_fars\_cases (get\_fars), 7

get\_fars\_crash\_details (get\_fars), 7

get\_fars\_crash\_list (get\_fars), 7

get\_fars\_crash\_persons, 10

get\_fars\_crash\_vehicles, 11

get\_fars\_crashes (get\_fars), 7

get\_fars\_summary (get\_fars), 7

get\_fars\_year (get\_fars), 7

get\_fars\_year(), 12

get\_fars\_zip, 12

get\_fars\_zip(), 6

get\_vars (fars\_vars), 4

getwd(), 7, 12

mmucc\_codes, 13

read\_crashapi, 14