

# Package ‘crsmeta’

March 29, 2020

**Title** Extract Coordinate System Metadata

**Version** 0.3.0

**Description** Obtain coordinate system metadata from various data formats. There are functions to extract a 'CRS' (coordinate reference system, <[https://en.wikipedia.org/wiki/Spatial\\_reference\\_system](https://en.wikipedia.org/wiki/Spatial_reference_system)>) in 'EPSG' (European Petroleum Survey Group, <<http://www.epsg.org/>>), 'PROJ4' <<https://proj.org/>>, or 'WKT2' (Well-Known Text 2, <<http://docs.opengeospatial.org/is/12-063r5/12-063r5.html>>) forms. This is purely for getting simple metadata from in-memory formats, please use other tools for out of memory data sources.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.0

**Depends** R (>= 3.5.0)

**Suggests** testthat (>= 2.1.0), spelling

**Imports** methods

**URL** <https://github.com/hypertidy/crsmeta>

**BugReports** <https://github.com/hypertidy/crsmeta/issues>

**Language** en-US

**NeedsCompilation** no

**Author** Michael Sumner [aut, cre] (<<https://orcid.org/0000-0002-2471-7511>>)

**Maintainer** Michael Sumner <[mdsumner@gmail.com](mailto:mdsumner@gmail.com)>

**Repository** CRAN

**Date/Publication** 2020-03-29 10:10:02 UTC

## R topics documented:

crs_epsg	2
crs_input	3
crs_proj	4
crs_wkt2	5
sfx	6

<b>Index</b>	<b>7</b>
--------------	----------

---

crs_epsg	<i>Extract 'EPSG' value</i>
----------	-----------------------------

---

### Description

Obtain the 'EPSG' string from an object, if it has one. Supported inputs include sf.

### Usage

```
crs_epsg(x, ...)
```

### Arguments

x	object with 'EPSG' value
...	ignored

### Value

integer (or NA)

### References

[EPSG website](#)

### See Also

[crs\\_wkt2\(\)](#) [crs\\_proj\(\)](#) [crs\\_input\(\)](#)

### Examples

```
crs_epsg(sfx)
x <- sfx
attr(x$geom, "crs")$epsg <- NA ## oh no we lost it
crs_epsg(x)

crs_epsg(sfx_new) ## NA, doesn't exist now
```

---

crs_input	<i>Extract 'input' value</i>
-----------	------------------------------

---

### Description

Obtain the 'input' string from an object, if it has one. Supported inputs include sf ( $\geq 0.8-1$  - probably).

### Usage

```
crs_input(x, ...)
```

### Arguments

x	object with 'input' value
...	ignored

### Value

character (or NA)

### Warning

Note that the 'input' value could be almost anything, there is a huge variety of inputs that can work such as 4326, projstrings, WKT2 strings, EPSG declarations 'EPSG:4326', or common strings like 'WGS84' or 'NAD27'.

Strings like '+init=epsg:4326' have been deprecated but still can work, so beware.

### References

[sf](#)

### See Also

[crs\\_wkt2\(\)](#) [crs\\_proj\(\)](#) [crs\\_epsg\(\)](#)

### Examples

```
crs_input(sfx) ## doesn't have one
```

```
crs_input(sfx_new) ## a proj4string
```

---

crs_proj	<i>Extract 'PROJ4' string</i>
----------	-------------------------------

---

## Description

Obtain the 'PROJ4' string from an object, if it has one. Supported inputs include raster, sf, sp, and silicate.

## Usage

```
crs_proj(x, ...)
```

## Arguments

x	object with 'PROJ4' string
...	ignored

## Value

character string (or NA)

## References

[PROJ system website](#)

## See Also

[crs\\_epsg\(\)](#) [crs\\_wkt2\(\)](#) [crs\\_input\(\)](#)

## Examples

```
crs_proj(sfx)

crs_proj(sfx$geom)

crs_proj(sfx_new) ## NA
```

---

crs_wkt2	<i>Extract 'WKT2' string</i>
----------	------------------------------

---

### Description

Obtain the 'WKT2' string from an object, if it has one. Supported inputs include `sp` and `sf`.

### Usage

```
crs_wkt2(x, ...)
```

```
crs_wkt(x, ...)
```

### Arguments

<code>x</code>	object with 'WKT2' string
<code>...</code>	ignored

### Details

The functions `crs_wkt()` and `crs_wkt2()` are aliased, they do the same thing.

### Value

character string (or NA)

### Warning

For WKT2 only, PROJ6 and beyond

### References

[WKT2 specification](#)

### See Also

`crs_epsg()` `crs_proj()` `crs_wkt()` `crs_input()`

### Examples

```
crs_wkt2(sfx) # NA
crs_wkt2(sfx$geom) # NA
```

```
crs_wkt2(sfx_new)
crs_wkt2(sfx_new$geom)
```

---

`sfx`*Simple features example data*

---

**Description**

A copy of the 'minimal\_mesh' data set from the `silicate` package, with coordinate reference system information added.

**Details**

`sfx` is the old-style PROJ.4 and EPSG code CRS (prior to sf 0.8-1).

`sfx_new` is the new-style WKT2, with user input.

**Warning**

do not use this data in real situations, or as exemplary of the 'sf' format. It was created purely to add examples to this package.

**Examples**

```
## three equivalent representations, of increasing richness
crs_epsg(sfx)

crs_proj(sfx)

crs_wkt2(sfx) ## did not exist in earlier sf

## new style
crs_epsg(sfx_new) ## NA!
crs_proj(sfx_new) ## NA!

crs_input(sfx_new)
crs_wkt(sfx_new)
```

# Index

`crs_epsg`, 2  
`crs_epsg()`, 3–5  
`crs_input`, 3  
`crs_input()`, 2, 4, 5  
`crs_proj`, 4  
`crs_proj()`, 2, 3, 5  
`crs_wkt (crs_wkt2)`, 5  
`crs_wkt()`, 5  
`crs_wkt2`, 5  
`crs_wkt2()`, 2–5

`sfx`, 6  
`sfx_new (sfx)`, 6