

# Package: scgraph (via r-universe)

August 18, 2024

**Title** Common Forms for Graph Structures

**Version** 0.0.1.9003

**Description** Provides support for the 'silicate' common form data structure for igraph.

**Depends** R (>= 3.3.2), silicate (>= 0.1.5)

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Suggests** testthat, knitr, rmarkdown, covr, gibble

**RoxygenNote** 6.1.1

**Imports** dplyr, igraph, rlang, tibble, tidygraph

**Remotes** hypertidy/silicate

**VignetteBuilder** knitr

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

**RemoteUrl** <https://github.com/hypertidy/scgraph>

**RemoteRef** HEAD

**RemoteSha** 9f2579e53a808e08e391726d2b2e5b4e8f5bd234

## Contents

as.igraph . . . . .	2
as_tbl_graph . . . . .	2
SC . . . . .	3
<b>Index</b>	<b>4</b>

---

as.igraph *Convert model to a graph.*

---

### Description

Only segments from an input model are extracted and used to build the graph.

### Usage

```
## S3 method for class 'SC'
as.igraph(x, ..., layout = TRUE)

## S3 method for class 'sf'
as.igraph(x, ..., layout = TRUE)

sc_as_igraph(x, ..., layout = TRUE)

## S3 method for class 'SC0'
as.igraph(x, ..., layout = TRUE)
```

### Arguments

x	input model
...	arguments passed to methods
layout	keep the input vertex coordinates x-y as the graph layout, defaults to 'TRUE' otherwise no layout is provided ## sf ## spatstat ## raw track data ## make up segments starting with the unjoin thing, that might show the way forward

### Examples

```
data("minimal_mesh", package = "silicate")
as.igraph(minimal_mesh)
```

---

as\_tbl\_graph *Convert to tidygraph.*

---

### Description

Convert to tidygraph.

### Usage

```
## S3 method for class 'SC'
as_tbl_graph(x, ...)

## S3 method for class 'sf'
as_tbl_graph(x, ...)
```

**Arguments**

x	model
...	other args

**Examples**

```
library(tidygraph)
library(scgraph)
data("minimal_mesh", package = "silicate")
as_tbl_graph(minimal_mesh)
library(silicate)
umodel <- SC(as_tbl_graph(minimal_mesh))
#gibble.PATH <- function(x, ...) {
#inner_join(x[["path"]], x[["path_link_vertex"]] %>% group_by(path) %>% summarize(nrow = n()) ) %>%
# dplyr::mutate(ncol = 2, type = "MULTILINESTRING")
#}
#library(gibble)
#geomap <- gibble(prim %>% PATH())
#silicate:::build_sf()
## some kind of round trip
```

---

SC

---

SC

---

**Description**

tbl\_graph methods for the universal model

**Usage**

```
## S3 method for class 'tbl_graph'
SC(x, ...)
```

**Arguments**

x	input model
...	arguments passed to methods

# Index

`as.igraph`, [2](#)

`as_tbl_graph`, [2](#)

SC, [3](#)

`sc_as_igraph` (`as.igraph`), [2](#)