R pour le Géospatial

Programmer

Etienne Racine

Valeurs manquantes

```
tibble(WKT = c("POINT(1 2)", "POINT(1 2)")) %>% st_as_sf(wkt = "WKT")
```

```
tibble(WKT = c("POINT(1 2)", "POINT EMPTY")) %>% st_as_sf(wkt = "WKT")
```

```
tibble(WKT = c("POINT(1 2)", NA)) %>% st_as_sf(wkt = "WKT")
#> OGR: Unsupported geometry type
#> Error in CPL_sfc_from_wkt(x) : OGR error
```

Tidy

```
school <-
  school %>%
  mutate(buffer_3km = geometry %>%
           st_transform(3978) %>%
           st_buffer(3000) %>% #<<
           st_transform(4326)
         )
```

Base

```
school$buffer_3km <- st_transform(
  st_buffer(
    st_transform(
      school$geometry, 4326),
    3000),
  3978)
```

sf ou sfc

```
x <- data.frame(geometry = c("POINT(1 2)", "POINT EMPTY")) %>% st_as_sf(wkt = "geometry")
class(x)
```

```
## [1] "sf"         "data.frame"
```

```
class(st_geometry(x))
```

```
## [1] "sfc_POINT" "sfc"
```

```
x %>%
  summarise(geometry = st_combine(x) %>% st_cast("LINESTRING")) %>%
  st_geometry() %>%
  class()
```

```
## [1] "sfc_LINESTRING" "sfc"
```

Implémenter `sfc`, puis utiliser `st_geometry()` pour `sf`

```
my_print <- function(x) UseMethod("my_print")

my_print.sfc <- function(x) {
  gsub("^(.{3})\\w+", "\\1", st_as_text(x))
}

my_print.sf <- function(x) {
  my_print(st_geometry(x))
}
```

```
my_print(x)
```

```
## [1] "POI (1 2)" "POI EMPTY"
```

```
x %>%
  mutate(my = my_print(geometry))
```

```
## Simple feature collection with 2 features and 1 field (with 1 geometry empty)
## geometry type:  POINT
## dimension:      XY
## bbox:           xmin: 1 ymin: 2 xmax: 1 ymax: 2
## epsg (SRID):    NA
## proj4string:    NA
```

À vous

- Comment modifier le comportement d'une fonction selon le type de géométrie?