bcdata

Andy Teucher[1]

Sam Albers[2]

Stephanie Hazlitt[2]

GIS CoP Face to Face

2019-11-27

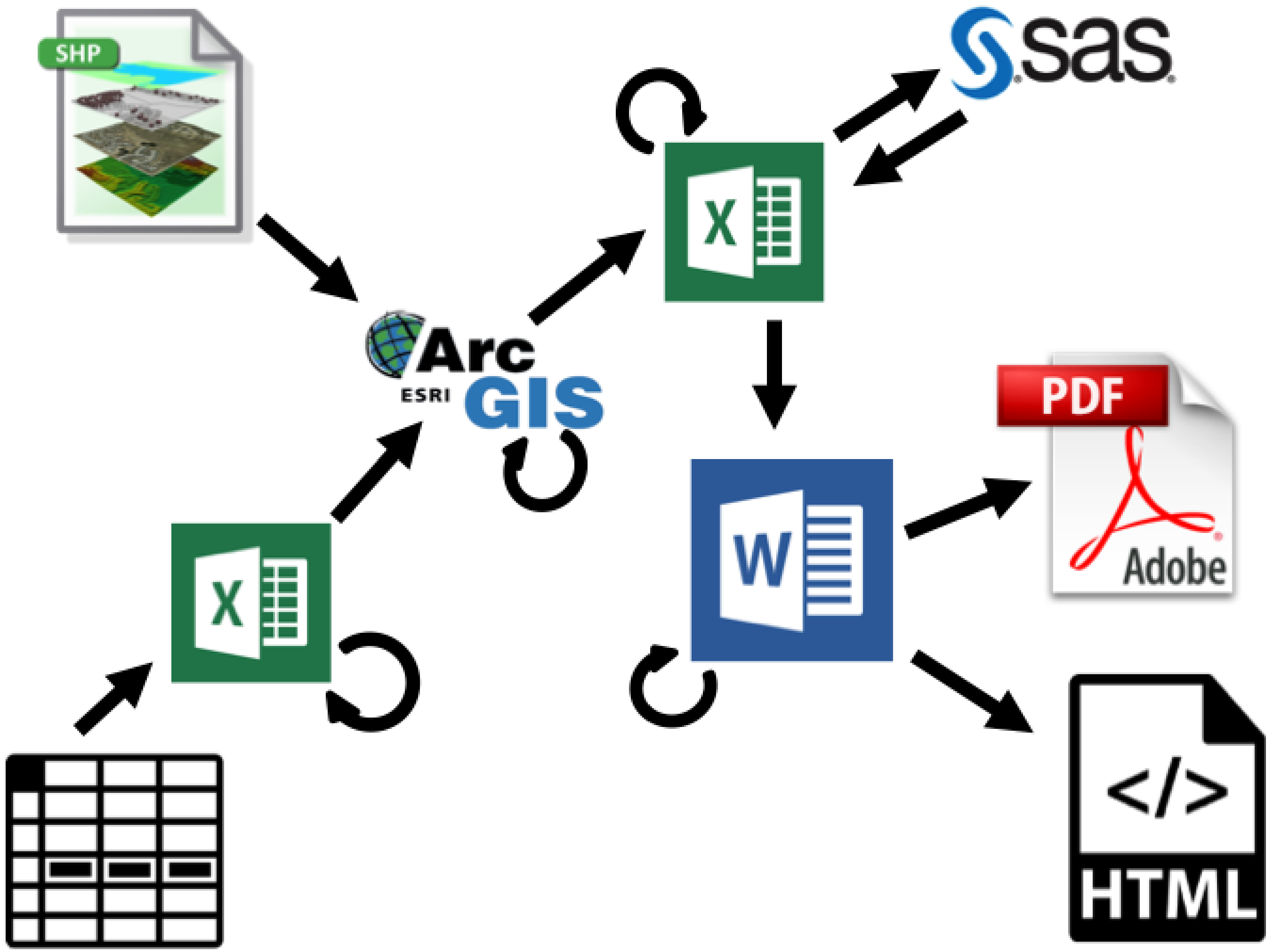
1: Ministry of Environment & Climate Change Strategy

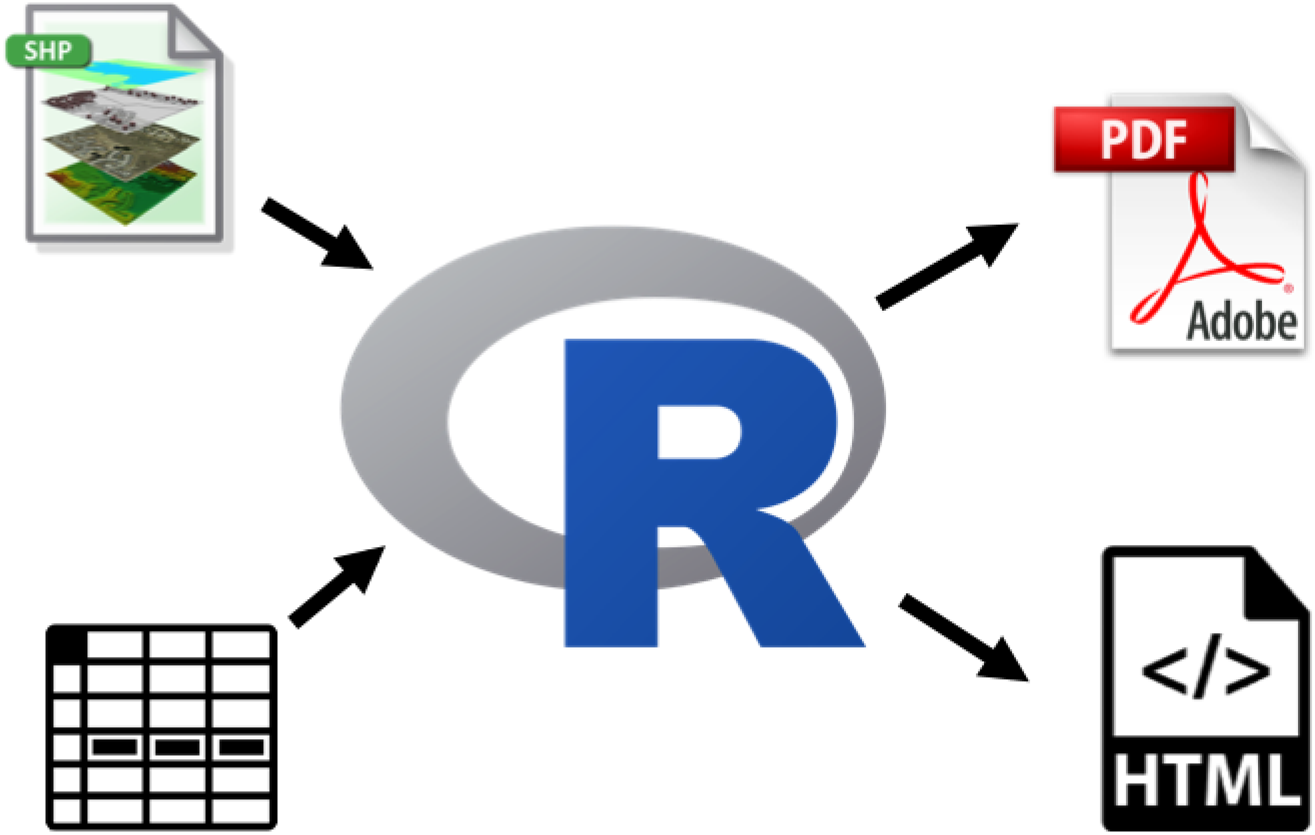
2: Ministry of Citizens' Services

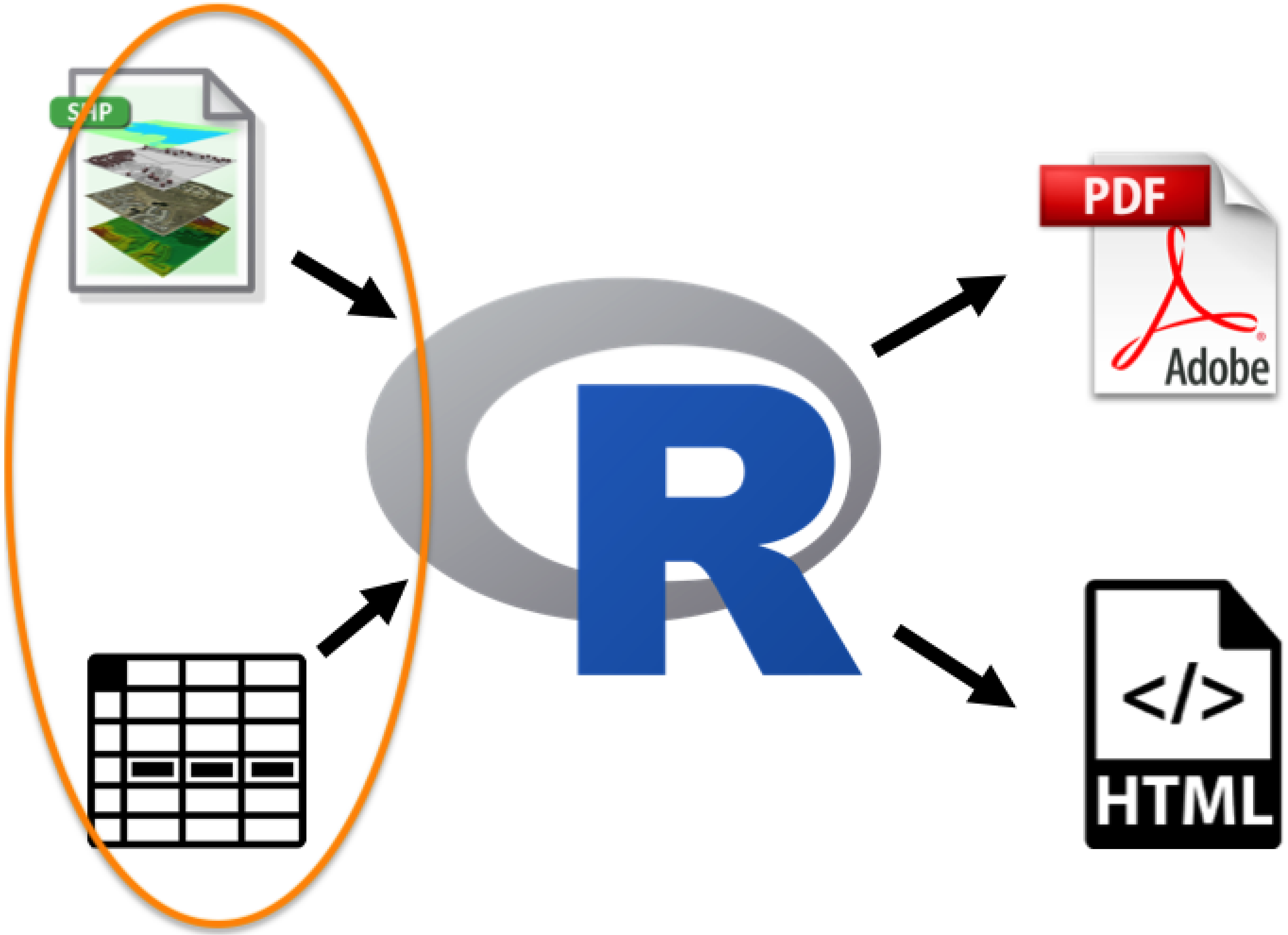
Open source programming language

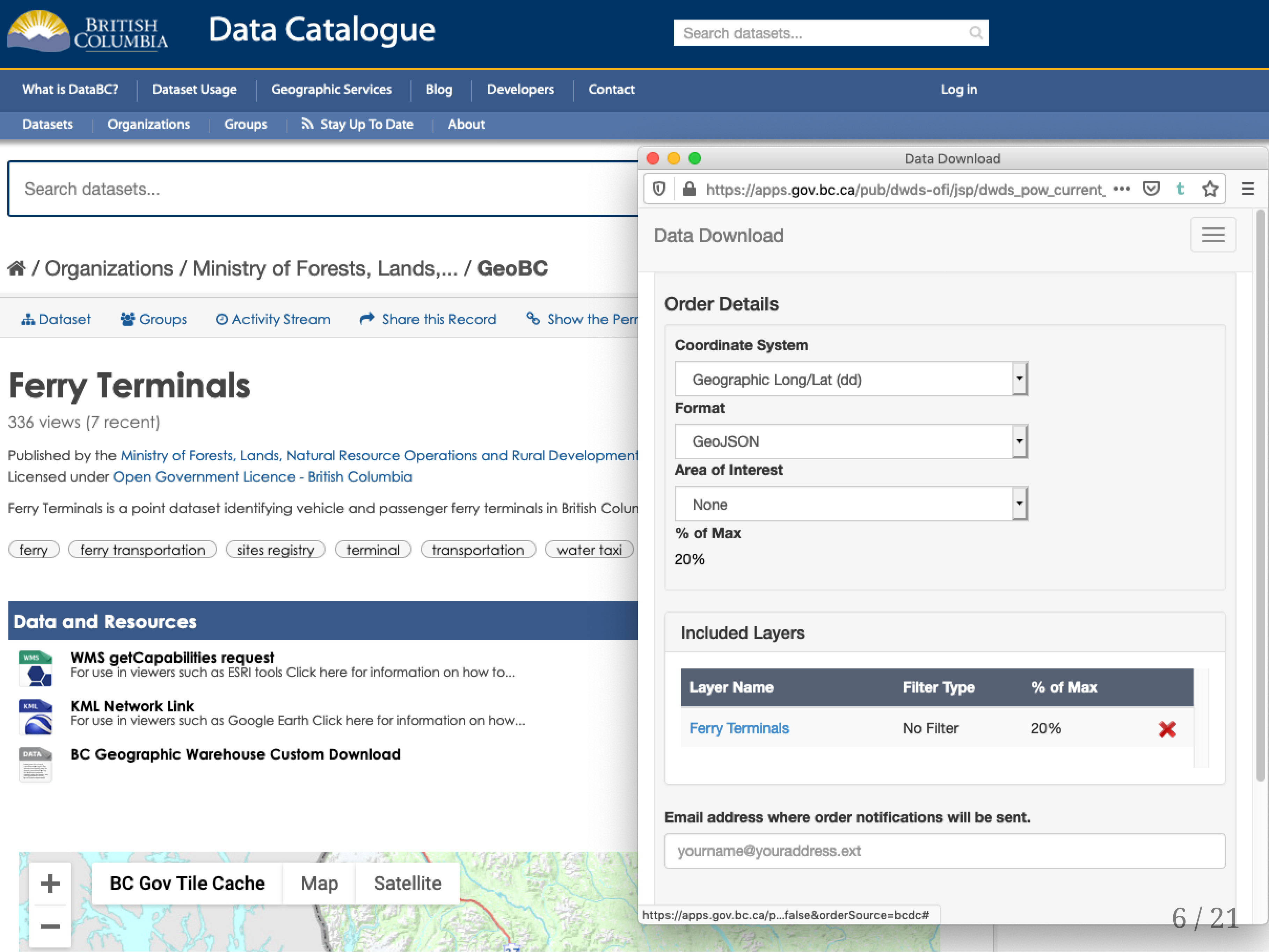
Data analysis focus

Cross-platform


SHP
Arc
ESRI
GIS
sas
X
X
W
PDF
Adobe
</>
HTML


SHP
PDF
Adobe
</>
HTML


SHP
PDF
Adobe
HTML



/ Organizations / Ministry of Forests, Lands,... / GeoBC

Ferry Terminals

336 views (7 recent)

Published by the Ministry of Forests, Lands, Natural Resource Operations and Rural Development
Licensed under Open Government Licence - British Columbia

Ferry Terminals is a point dataset identifying vehicle and passenger ferry terminals in British Colun

ferry | ferry transportation | sites registry | terminal | transportation | water taxi

Data and Resources

WMS getCapabilities request
For use in viewers such as ESRI tools Click here for information on how to...

KML Network Link
For use in viewers such as Google Earth Click here for information on how...

BC Geographic Warehouse Custom Download

Data Download

https://apps.gov.bc.ca/pub/dwds-ofi/jsp/dwds_pow_current_

Data Download

Order Details

Coordinate System: Geographic Long/Lat (dd)
Format: GeoJSON
Area of Interest: None
% of Max: 20%

Included Layers

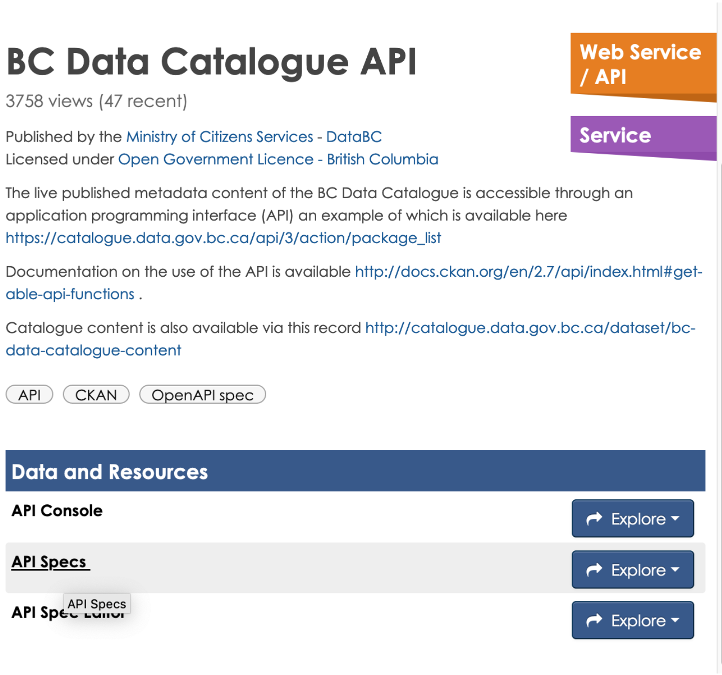
Layer Name	Filter Type	% of Max
Ferry Terminals	No Filter	20%

Email address where order notifications will be sent.

yourname@youraddress.ext

https://apps.gov.bc.ca/p...false&orderSource=bcdc#

Application Programming Interfaces (APIs)



BC Data Catalogue API

Web Service / API

Service

3758 views (47 recent)

Published by the Ministry of Citizens Services - DataBC
Licensed under Open Government Licence - British Columbia

The live published metadata content of the BC Data Catalogue is accessible through an application programming interface (API) an example of which is available here https://catalogue.data.gov.bc.ca/api/3/action/package_list

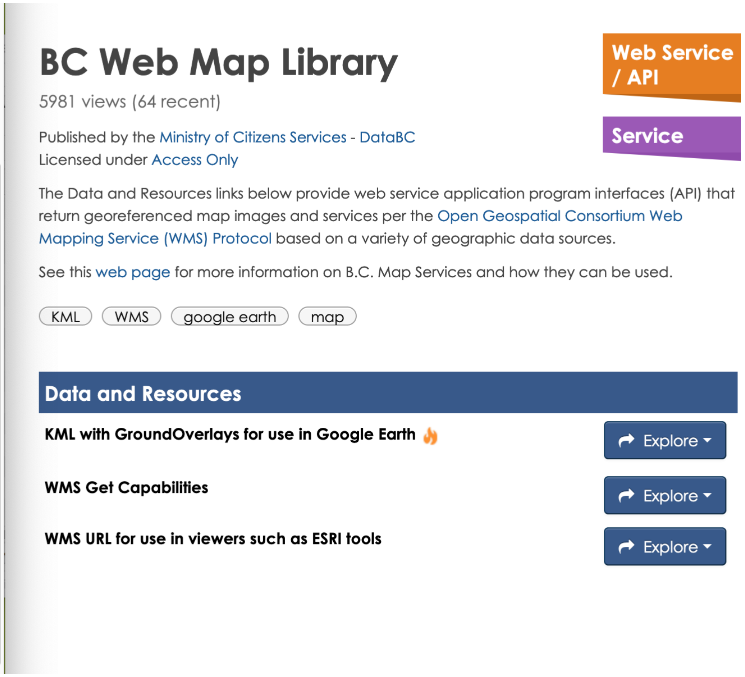
Documentation on the use of the API is available http://docs.ckan.org/en/2.7/api/index.html#get-able-api-functions .

Catalogue content is also available via this record http://catalogue.data.gov.bc.ca/dataset/bc-data-catalogue-content

API | CKAN | OpenAPI spec

Data and Resources

API Console	Explore
API Specs	Explore
API Spec Editor	Explore



BC Web Map Library

Web Service / API

Service

5981 views (64 recent)

Published by the Ministry of Citizens Services - DataBC
Licensed under Access Only

The Data and Resources links below provide web service application program interfaces (API) that return georeferenced map images and services per the Open Geospatial Consortium Web Mapping Service (WMS) Protocol based on a variety of geographic data sources.

See this web page for more information on B.C. Map Services and how they can be used.

KML | WMS | google earth | map

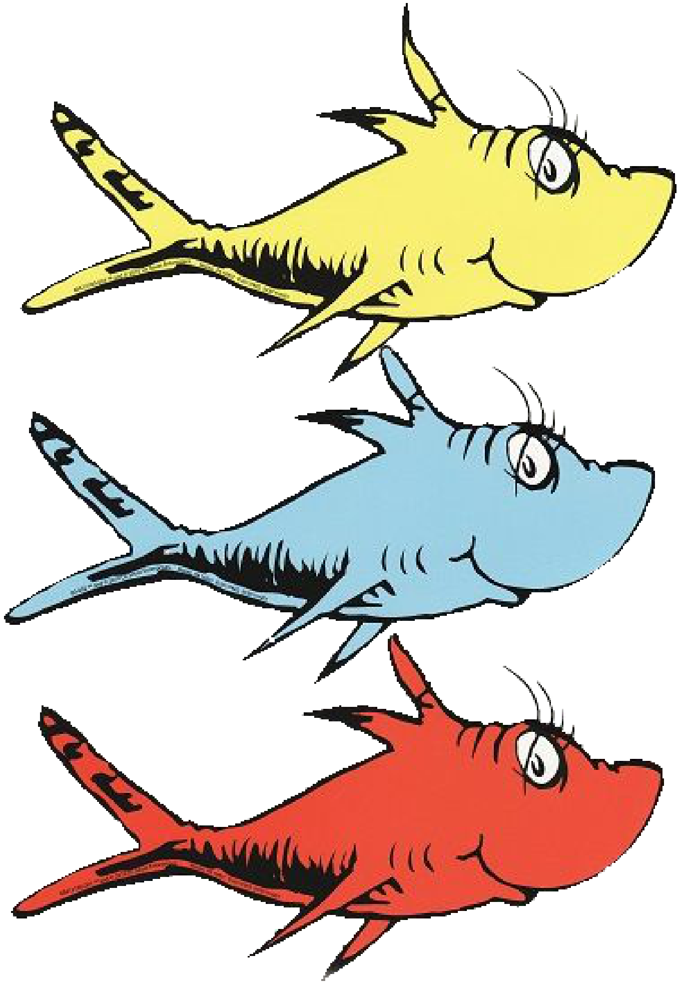
Data and Resources

KML with GroundOverlays for use in Google Earth	Explore
WMS Get Capabilities	Explore
WMS URL for use in viewers such as ESRI tools	Explore

one `function()`

two `function()`

I need a `function()`



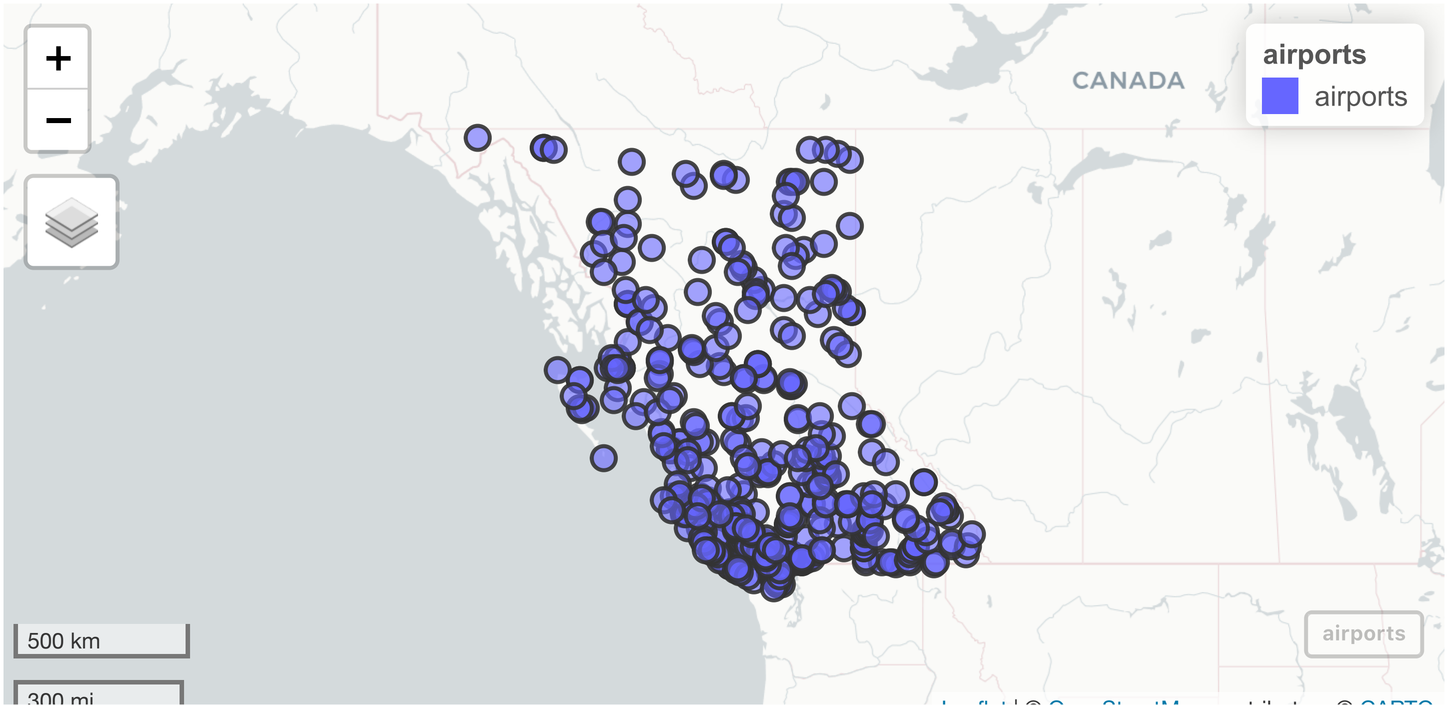
- **`bcdc_browse()`**
  - Open the catalogue in your default browser
- **`bcdc_search()`**
  - Search records in the catalogue
- **`bcdc_get_record()`**
  - Print a catalogue record
- **`bcdc_get_data()`**
  - Get catalogue data
- **`bcdc_query_geodata()`**
  - Get & query B.C. geospatial data from a web service

`bcdc_get_data()` - spatial data

```r
bcdc_get_data("bc-airports", resource = "4d0377d9-e8a1-429b-824f-0ce8f363512c")
```

```r
### OR use BCGW name #####
airports <- bcdc_get_data("WHSE_IMAGERY_AND_BASE_MAPS.GSR_AIRPORTS_SVW")

library(mapview)
mapview(airports)
```



bcdc_query_geodata()

```r
bcdc_query_geodata("municipalities-legally-defined-administrative-areas-of-bc")
```

```
Querying 'municipalities-legally-defined-administrative-areas-of-bc' record
● Using collect() on this object will return 161 features and 17 fields
● At most six rows of the record are printed here
────────────────────────────────────────────────────────────────────────────────
Simple feature collection with 6 features and 17 fields
geometry type:  MULTIPOLYGON
dimension:      XY
bbox:           xmin: 1011118 ymin: 528481.5 xmax: 1631603 ymax: 1056560
epsg (SRID):    3005
proj4string:    +proj=aea +lat_1=50 +lat_2=58.5 +lat_0=45 +lon_0=-126 +x_0=1000000 +y_0=0 +e
# A tibble: 6 x 18
  id    LGL_ADMIN_AREA_… ADMIN_AREA_NAME ADMIN_AREA_ABBR… ADMIN_AREA_BOUN… ADMIN_AREA_TYPE
  <chr>            <int> <chr>           <chr>            <chr>            <chr>
1 WHSE…              160 Village of Lyt… Lytton           Legal            Municipality
2 WHSE…              161 City of Merritt Merritt          Legal            Municipality
3 WHSE…              162 Sun Peaks Moun… Sun Peaks        Legal            Municipality
4 WHSE…              174 The Corporatio… Nelson           Legal            Municipality
5 WHSE…                3 The Corporatio… Burns Lake       Legal            Municipality
6 WHSE…                4 District of Fo… Fort St James    Legal            Municipality
# … with 12 more variables: ADMIN_AREA_GROUP_NAME <chr>, CHANGE_REQUESTED_ORG <chr>,
#   UPDATE_TYPE <chr>, WHEN_UPDATED <date>, OIC_NUMBER <chr>, OIC_YEAR <chr>,
#   AFFECTED_ADMIN_AREA_ABRVN <chr>, FEATURE_AREA_SQM <dbl>, FEATURE_LENGTH_M <dbl>,
#   OBJECTID <int>, SE_ANNO_CAD_DATA <chr>, geometry <MULTIPOLYGON [m]>
```

Select columns (attributes) with `select()`

```
bcdc_query_geodata("municipalities-legally-defined-administrative-areas-of-bc") %>%
  select(ADMIN_AREA_ABBREVIATION, ADMIN_AREA_GROUP_NAME)
```

```
Querying 'municipalities-legally-defined-administrative-areas-of-bc' record
● Using collect() on this object will return 161 features and 5 fields
● At most six rows of the record are printed here
────────────────────────────────────────────────────────────────────────────────────
Simple feature collection with 6 features and 5 fields
geometry type:  MULTIPOLYGON
dimension:      XY
bbox:           xmin: 1011118 ymin: 528481.5 xmax: 1631603 ymax: 1056560
epsg (SRID):    3005
proj4string:    +proj=aea +lat_1=50 +lat_2=58.5 +lat_0=45 +lon_0=-126 +x_0=1000000 +y_0=0 +e
# A tibble: 6 x 6
  id    LGL_ADMIN_AREA_… ADMIN_AREA_ABBR… ADMIN_AREA_GROU… OBJECTID
  <chr>            <int> <chr>            <chr>               <int>
1 WHSE…              160 Lytton           Thompson-Nicola…    13685
2 WHSE…              161 Merritt          Thompson-Nicola…    13686
3 WHSE…              162 Sun Peaks        Thompson-Nicola…    13687
4 WHSE…              174 Nelson           Regional Distri…    13688
5 WHSE…                3 Burns Lake       Regional Distri…    13602
6 WHSE…                4 Fort St James    Regional Distri…    13603
# … with 1 more variable: geometry <MULTIPOLYGON [m]>
```

Filter rows (features) with `filter()`

```
bcdc_query_geodata("municipalities-legally-defined-administrative-areas-of-bc") %>%
  select(ADMIN_AREA_ABBREVIATION, ADMIN_AREA_GROUP_NAME) %>%
  filter(ADMIN_AREA_GROUP_NAME == "Capital Regional District")
```

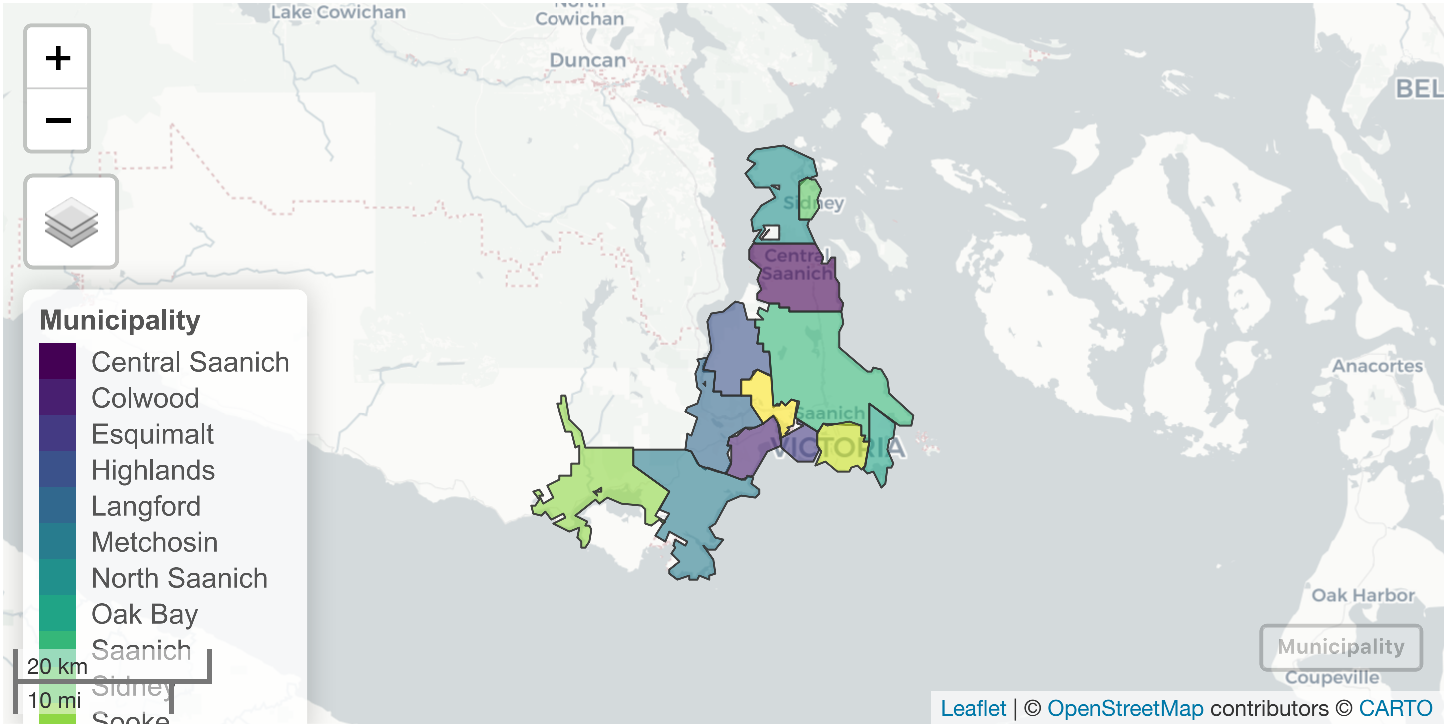
```
Querying 'municipalities-legally-defined-administrative-areas-of-bc' record
● Using collect() on this object will return 13 features and 5 fields
● At most six rows of the record are printed here
```

```
Simple feature collection with 6 features and 5 fields
geometry type:  MULTIPOLYGON
dimension:      XY
bbox:           xmin: 1174651 ymin: 368738.7 xmax: 1195365 ymax: 403223.9
epsg (SRID):    3005
proj4string:    +proj=aea +lat_1=50 +lat_2=58.5 +lat_0=45 +lon_0=-126 +x_0=1000000 +y_0=0 +e
# A tibble: 6 x 6
  id     LGL_ADMIN_AREA_… ADMIN_AREA_ABBR… ADMIN_AREA_GROU… OBJECTID
  <chr>             <int> <chr>            <chr>               <int>
1 WHSE…               258 Central Saanich  Capital Regiona…    13727
2 WHSE…               259 Colwood          Capital Regiona…    13728
3 WHSE…               260 Esquimalt        Capital Regiona…    13729
4 WHSE…               261 Highlands        Capital Regiona…    13730
5 WHSE…               262 Langford         Capital Regiona…    13731
6 WHSE…               263 Metchosin        Capital Regiona…    13732
# … with 1 more variable: geometry <MULTIPOLYGON [m]>
```

Get the data with `collect()`

```
crd_mun <- bcdc_query_geodata("municipalities-legally-defined-administrative-areas-
  select(ADMIN_AREA_ABBREVIATION, ADMIN_AREA_GROUP_NAME) %>%
  filter(ADMIN_AREA_GROUP_NAME == "Capital Regional District") %>%
  collect()
```

```
mapview(crd_mun, zcol = "ADMIN_AREA_ABBREVIATION", layer = "Municipality")
```



Let's find the 'greenest' city in the CRD

In addition to normal logical predicates (`==`. `!=`, `>`, `<`, etc.), `filter()` can take *geometric predicates*:

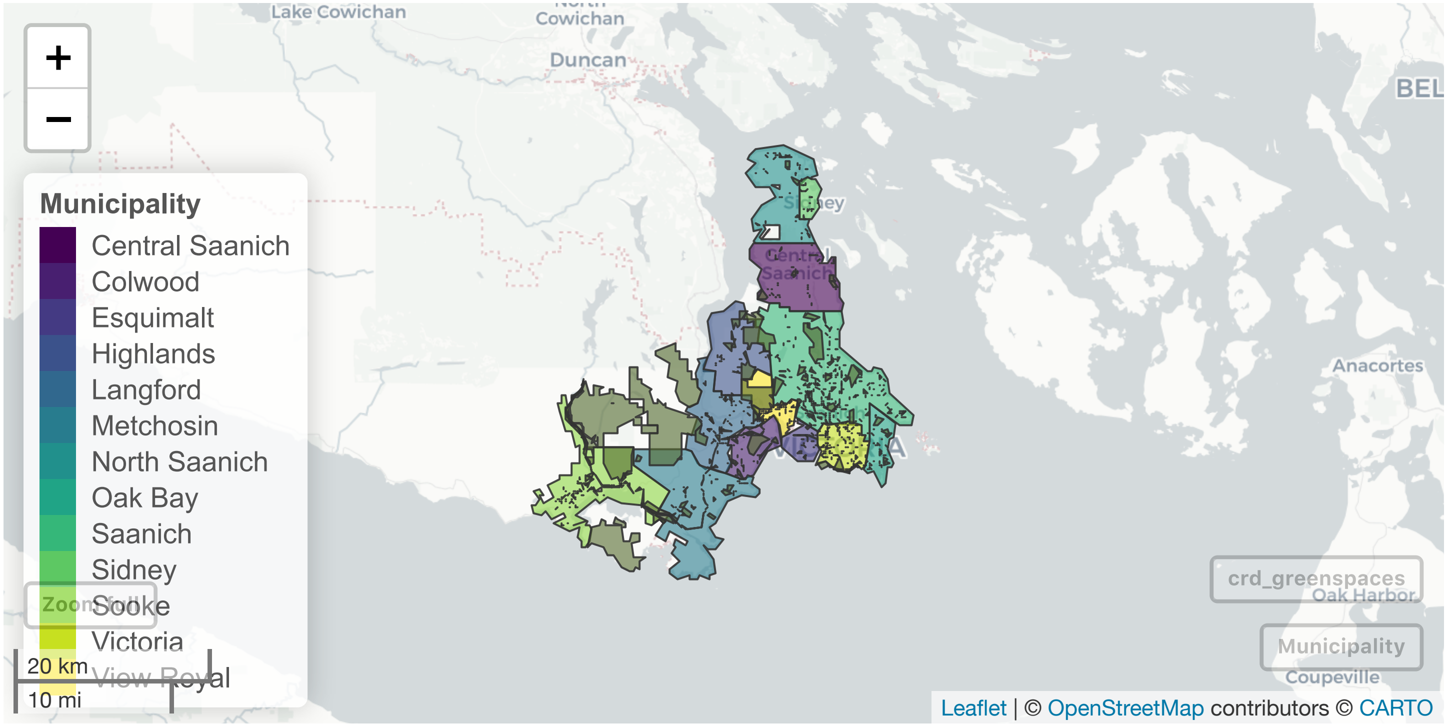
`EQUALS`, `DISJOINT`, `INTERSECTS`, `TOUCHES`, `CROSSES`, `WITHIN`, `CONTAINS`, `OVERLAPS`, `RELATE`, `DWITHIN`, `BEYOND`

```
crd_greenspaces <- bcdc_query_geodata("local-and-regional-greenspaces") %>%
  select(PARK_NAME, PARK_TYPE, PARK_PRIMARY_USE) %>%
  filter(INTERSECTS(crd_mun)) %>%
  collect()
```

https://catalogue.data.gov.bc.ca/dataset/local-and-regional-greenspaces

```
muni_map <- mapview(crd_mun, zcol = "ADMIN_AREA_ABBREVIATION",
                    layer = "Municipality")

muni_map +
  mapview(crd_greenspaces, col.regions = "darkolivegreen")
```



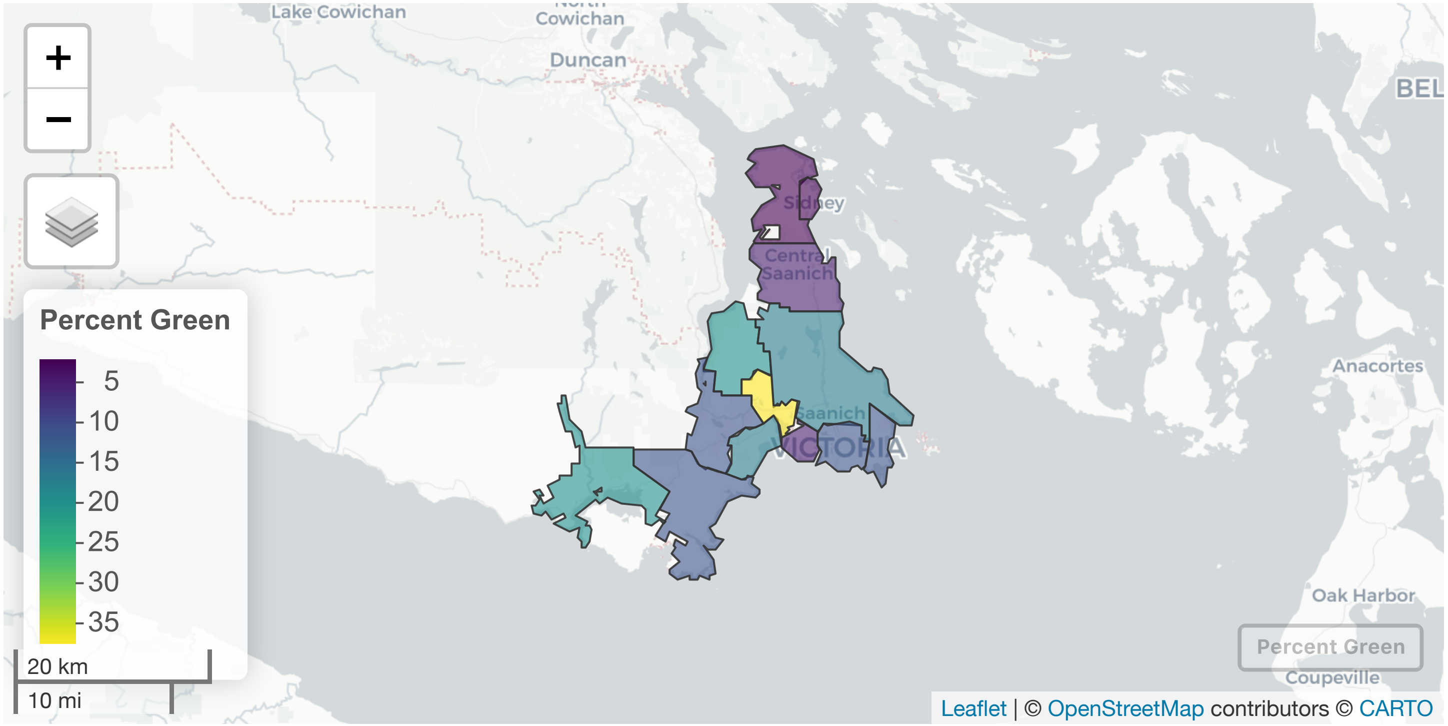
Find the amount of green space in each municipality

```r
library(sf)
grn_intersected <- st_intersection(crd_greenspaces, crd_mun)

grn_intersected_summary <- group_by(grn_intersected,
                                    ADMIN_AREA_ABBREVIATION) %>%
  summarise(grn_area = sum(st_area(geometry)))

crd_mun <- left_join(crd_mun, st_drop_geometry(grn_intersected_summary),
                     by = "ADMIN_AREA_ABBREVIATION") %>%
  mutate(muni_area = st_area(geometry),
         percent_green = (grn_area / muni_area) * 100)
```

```r
mapview(crd_mun, zcol = "percent_green", layer = "Percent Green")
```



Kudos

- BC Data Catalogue Team
- Michelle Douville
- Simon Norris
- Our bosses for giving us time/space for innovation and collaboration

- Install from CRAN:
  - https://cran.r-project.org/package=bcdata
  - `install.packages("bcdata")`
- Help & documentation:
  - https://bcgov.github.io/bcdata
- Issues/bugs:
  - https://github.com/bcgov/bcdata/issues

Bonus slide!

Get VRI for ~ 10kmx10km area NW of Prince George

```
tictoc::tic()
vri_pg <- bcdc_query_geodata("WHSE_FOREST_VEGETATION.VEG_COMP_LYR_R1_POLY") %>%
  filter(BBOX(c(1196754.2218,1005314.8161,1206862.4977,1015887.0438),
              crs = "EPSG:3005")) %>%
  collect()
tictoc::toc()
4.646 sec elapsed

plot(vri_pg["PROJ_AGE_1"])
```