www.kb.gg/databases

Landsat Metadata

Kevin Booth

www.kb.gg/databases

Bulk Metadata Service

This page allows users to download bulk metadata files for U.S. Landsat Analysis Ready Data (ARD) and Landsat Collection 1 Level-1 data. These files are updated daily.

Note: the files on this web page are for the entire Landsat archive and may take several minutes to download depending on your internet connection.

Access to Level-1 or U.S. ARD bulk metadata downloads for a particular area of interest or date range is available using the EarthExplorer Export Metadata functionality. Details about this tool are on the the last two pages of this EarthExplorer Help Documentation.

Updates to Landsat Level-1 metadata fields were made in May 2018 and are described here.

U.S Landsat Analysis Ready Data (ARD)

- 1984 - present - .xml.gz
- 1984 - present - .csv.gz

Landsat Collection 1 Level-1

Landsat 8 Operational Land Imager (OLI)/ Thermal Infrared Sensor (TIRS) (combined Pre- and On-WRS)

- April 2013 - present - .xml.gz
- April 2013 - present - .csv.gz

Landsat 7 Enhanced Thematic Mapper Plus (ETM+) (combined SLC-on/off)

- 1999 - Present - .xml.gz
- 1999 - Present - .csv.gz

Landsat 4-5 Thematic Mapper

- 1980-2012 - .xml.gz
- 1980-2012 - .csv.gz

Landsat 1-5 Multispectral Scanner (MSS)

- 1972-1997, 2012-2013 - .xml.gz
- 1972-1997, 2012-2013 - .csv.gz

www.kb.gg/databases

```
SELECT count(*)
FROM scenes;
```

www.kb.gg/databases

```
SELECT count(*)
FROM scenes;
```

	count bigint
1	8330247

www.kb.gg/databases

```
SELECT count(*)
FROM scenes;
```

	count bigint
1	8330247

www.kb.gg/databases

```
SELECT scene_id,
sensor,
platform,
path,
row,
date_acquired,
cloud_cover
FROM scenes LIMIT 10;
```

www.kb.gg/databases

```
SELECT scene_id,
sensor,
platform,
path,
row,
date_acquired,
cloud_cover
FROM scenes LIMIT 10;
```

	scene_id character (21)	sensor character varying (8)	platform character (4)	path integer	row integer	date_acquired date	cloud_cover integer
1	LT513604519913…	TM	L05	136	45	1991-11-03	100
2	LT515203419913…	TM	L05	152	34	1991-11-03	8
3	LT515203519913…	TM	L05	152	35	1991-11-03	2
4	LT515203619913…	TM	L05	152	36	1991-11-03	2
5	LT515203719913…	TM	L05	152	37	1991-11-03	0
6	LT515203819913…	TM	L05	152	38	1991-11-03	0
7	LT515204019913…	TM	L05	152	40	1991-11-03	0
8	LT515204219913…	TM	L05	152	42	1991-11-03	0
9	LE72301112003…	ETM	L07	230	111	2003-01-14	100
10	LE72301122003…	ETM	L07	230	112	2003-01-14	76

www.kb.gg/databases

```
WITH city_data AS (
    SELECT gid, name || ', ' || sov0name AS city_name, geom FROM cities
)
SELECT avg(avg_clouds) AS avg_clouds,
city_data.city_name
FROM scene_clouds, city_data
WHERE ST_Intersects(scene_clouds.geom, city_data.geom)
GROUP BY city_data.city_name
ORDER BY avg_clouds DESC
LIMIT 10;
```

www.kb.gg/databases

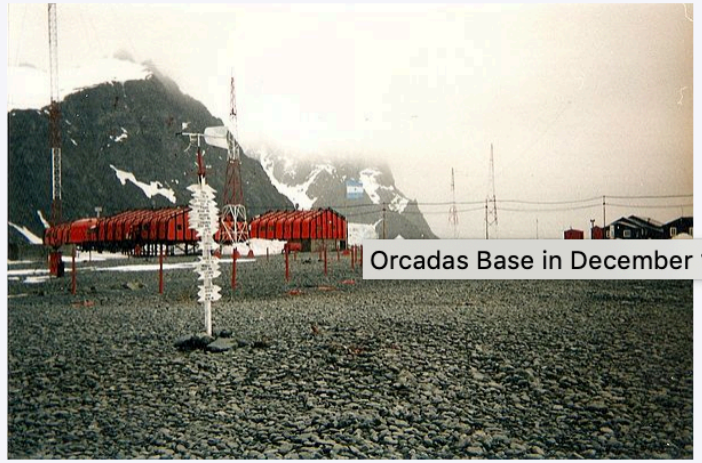
```sql
WITH city_data AS (
    SELECT gid, name || ', ' || sov0name AS city_name, geom FROM cities
)
SELECT avg(avg_clouds) AS avg_clouds,
city_data.city_name
FROM scene_clouds, city_data
WHERE ST_Intersects(scene_clouds.geom, city_data.geom)
GROUP BY city_data.city_name
ORDER BY avg_clouds DESC
LIMIT 10;
```

	avg_clouds numeric	city_name text
1	90.2180555555555556	Orcadas Station, Indeterminate
2	86.3240740740740741	Peter I Island, Indeterminate
3	85.0031746031746032	Signy Research Station, Indeter…
4	77.1940670289855073	Capitan Arturo Prat Station, Ind…
5	76.9476309226932668	Ebolowa, Cameroon
6	75.9308755760368664	Kayes, Congo (Brazzaville)
7	75.8926829268292683	Matadi, Congo (Kinshasa)
8	75.3929897694304474	Elephant Island, Indeterminate
9	73.9936261812781230	Palmer Station, Indeterminate
10	72.7564245810055866	Guiyang, China

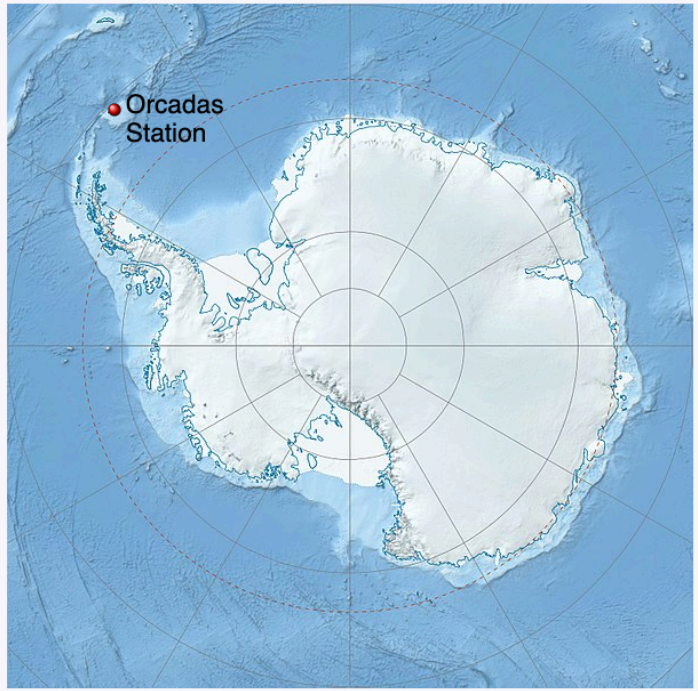
www.kb.gg/databases

Orcadas Station

Antarctic base



Orcadas Base in December 1996



Location of Orcadas Station in Antarctica

Coordinates: 60°44′17″S 44°44′17″W

www.kb.gg/databases

```sql
WITH city_data AS (
    SELECT gid, name || ', ' || sov0name AS city_name, geom FROM cities WHERE sov0name = 'United States'
)
SELECT avg(avg_clouds) AS avg_clouds,
city_data.city_name
FROM scene_clouds, city_data
WHERE ST_Intersects(scene_clouds.geom, city_data.geom)
GROUP BY city_data.city_name
ORDER BY avg_clouds DESC
LIMIT 10;
```

www.kb.gg/databases

```sql
WITH city_data AS (
    SELECT gid, name || ', ' || sov0name AS city_name, geom FROM cities WHERE sov0name = 'United States'
)
SELECT avg(avg_clouds) AS avg_clouds,
city_data.city_name
FROM scene_clouds, city_data
WHERE ST_Intersects(scene_clouds.geom, city_data.geom)
GROUP BY city_data.city_name
ORDER BY avg_clouds DESC
LIMIT 10;
```

	avg_clouds numeric	city_name text
1	64.9606801244732279	Cold Bay, United States
2	59.9810934730163808	Sitka, United States
3	58.6358731067630434	Bethel, United States
4	58.0645156654210981	Kodiak, United States
5	57.7402745368827087	Juneau, United States
6	56.9922911003050376	Anchorage, United States
7	54.9956128651422790	Utqiaġvik, United States
8	53.2045454545454545	Burlington, United States
9	52.9685970715571338	Valdez, United States
10	52.8001817997241338	Point Hope, United States

www.kb.gg/databases

```sql
WITH city_data AS (
    SELECT gid, name || ', ' || sov0name AS city_name, geom FROM cities WHERE sov0name = 'United States'
)
SELECT avg(avg_clouds) AS avg_clouds,
city_data.city_name
FROM scene_clouds, city_data
WHERE ST_Intersects(scene_clouds.geom, city_data.geom)
GROUP BY city_data.city_name
ORDER BY avg_clouds ASC
LIMIT 10;
```

```sql
WITH city_data AS (
    SELECT gid, name || ', ' || sov0name AS city_name, geom FROM cities WHERE sov0name = 'United States'
)
SELECT avg(avg_clouds) AS avg_clouds,
city_data.city_name
FROM scene_clouds, city_data
WHERE ST_Intersects(scene_clouds.geom, city_data.geom)
GROUP BY city_data.city_name
ORDER BY avg_clouds ASC
LIMIT 10;
```

	avg_clouds numeric	city_name text
1	15.0460100647016535	Phoenix, United States
2	15.1528614457831325	San Bernardino, United States
3	15.1861662987490802	Tucson, United States
4	17.8125915080527086	El Paso, United States
5	19.1702763256161314	Los Angeles, United States
6	19.1736311239193084	Las Vegas, United States
7	19.9686360320933625	Flagstaff, United States
8	20.6985093927938520	Albuquerque, United States
9	21.0648968817915684	Fresno, United States
10	22.5455861070911722	Santa Fe, United States

www.kb.gg/databases

Waymo, Lyft partner to offer self-driving rides in Phoenix

Posted: 6:16 PM, May 07, 2019 **Updated:** 8:16 PM, May 07, 2019

AP **By:** Associated Press

Photo by: Waymo

www.kb.gg/databases

```
import geopandas
import psycopg2
import matplotlib.pyplot as plt
import arrow
from datetime import datetime
from cartopy import crs as ccrs

conn = psycopg2.connect("dbname=landsat user=postgres host=10.0.10.15 password=      ")
cur = conn.cursor()

crs = ccrs.Robinson()
crs_proj4 = crs.proj4_init

i = 0
for period in range(0, 24):
    i += 1
    day = int((365 / 24) * period) + 1
    print('Working On Day {}'.format(day))
    date_string = '2018-{:03d}'.format(day)
    statement = '''
        WITH scenes AS (
                SELECT scenes_clouds.cloud_cover, scenes_clouds.geom
                FROM scenes_clouds
                WHERE day = {}
        ),
        country AS (
                SELECT geom, gid
                FROM countries_small
        ),
        cloud_cover AS (
                SELECT avg(scenes.cloud_cover) AS cloud_cover, country.gid
                FROM scenes, country
                WHERE ST_Intersects(scenes.geom, country.geom)
                GROUP BY country.gid
        )
        SELECT cloud_cover.cloud_cover, countries_small.geom
        FROM cloud_cover
        JOIN countries_small ON countries_small.gid = cloud_cover.gid
        '''.format(day)

    graticules = geopandas.GeoDataFrame.from_postgis('SELECT * FROM graticules', conn)
    graticules_ee = graticules.to_crs(crs_proj4)

    countries = geopandas.GeoDataFrame.from_postgis(statement, conn)
    countries_ee = countries.to_crs(crs_proj4)

    fig, ax = plt.subplots(subplot_kw={'projection':crs})
    graticules_ee.plot(ax=ax, color='lightgray', linewidth=0.5, zorder=0)
    countries_ee.plot(ax=ax, column='cloud_cover', legend=True, vmax=100, vmin=0,
                      k=5, edgecolor='#333333', linewidth=0.2, zorder=1)

    plt.title(arrow.get(datetime.strptime(date_string, '%Y-%j')).format('MMMM Do'))
    plt.savefig('/data/countries/cloud_cover_{:03d}.png'.format(i), dpi=300)
    plt.clf()
```

www.kb.gg/databases