

STAC

SpatioTemporal Asset Catalogs
and
Open-Source Software Supporting Open Science

WGISS-48, Hanoi Vietnam

Dan Pilone
dan@element84.com || @danpilone

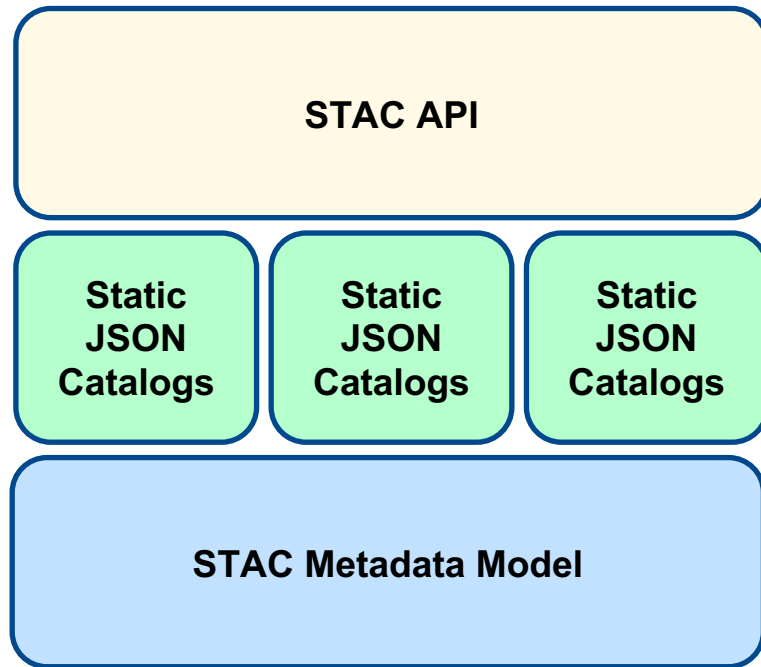
WHY??



SAINT KITTS AND

What is it?

- JSON defined metadata for geospatial catalogs and assets
 - Focus on search and discovery
 - Simple and extensible
- Dynamic APIs
 - Searchable on arbitrary fields
- Static catalogs
 - Crawlable



What is it not?

- Full-fledged metadata standard
- Single source of truth
- Covering *all* kind of datasets
- Seriously - **not a** replacement for ISO 19115, OGC CSW, ...



Radiant Earth
Foundation

EARTH IMAGERY FOR IMPACT



Element

84



MAXAR
TECHNOLOGIES



developmentSEED



THE CLIMATE
CORPORATION



HARRIS



Australian Government
Geoscience Australia



Google Earth Engine



HEXAGON
GEOSPATIAL



Boundless

The specification

- Catalogs

- There's always a root catalog
- Contains other catalogs and collections

- Collections

- Can contain other catalogs (called sub-catalogs) to group Items further
 - e.g., For Landsat-8 there might be sub-catalog by Landsat Path and Row
- Ultimately contains Items which are similar in some way(s)
 - e.g., same instrument, same region containing data from different instruments, etc.

- Items

- A single scene or set of datafiles for a specific location at a specific date and time (when and where)



STAC Ecosystem

<https://github.com/radiantearth/stac-spec/blob/master/implementations.md>

```
1 {
2   "type": "Feature",
3   "id": "LC80190232015097LGN00",
4   "geometry": {
5     "type": "Polygon",
6     "coordinates": [
7       [
8         [
9           -76.61570759444504,
10          52.43864121143704
11        ],
12        [
13          -76.6585808927941,
14          52.35408793014861
15        ],
16        [
17          -76.73673489501518,
18          52.19995522778887
19        ],
20        [
```

STAC URL



Paste your code on the left to validate

Validate

v0.7.0



Planet Disaster Data (planet-disaster-data)

Version 1.0.0

  <https://storage.googleapis.com/pdd-stac/disasters/catalog.json>

[Planet Disaster Data](#) makes imagery available directly to the public, volunteers, humanitarian organizations, and other coordinating bodies in support of the International Charter for Space and Major Disasters. Data is released for individual disaster events, providing a 30 day window pre- and post-disaster. Imagery is provided under Creative Commons licenses, free of charge, with either CC-BY-SA or CC-BY-NC.

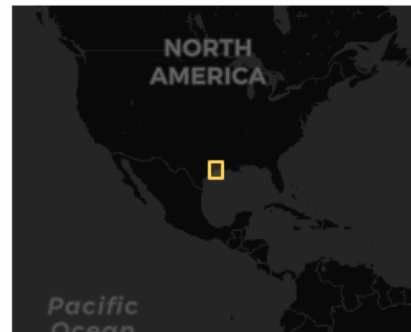
Catalogs

Title

[Hurricane Harvey](#)

<https://github.com/radianteearth/stac-browser>

<https://github.com/kbogg/qgis-stac-browser>



Provider

[Planet Disaster Team](#) <disaster-team@planet.com> (producer, licensor)

STAC 0.6.0

Version

Keywords disaster, open

License [CC-BY-SA-4.0](#)

Temporal 8/28/2017, 11:00:00

Extent AM - now

Going Deeper...



sat-utils

<https://github.com/sat-utils>

sat-api

<https://github.com/sat-utils/sat-api>

- STAC dynamic API reference implementation
- Node library for a (STAC) API
- Deploy your own API:
 - <https://github.com/sat-utils/sat-api-deployment>
- Crawl existing static STAC catalogs to index
- Keep up to date by subscribing to STAC SNS topics

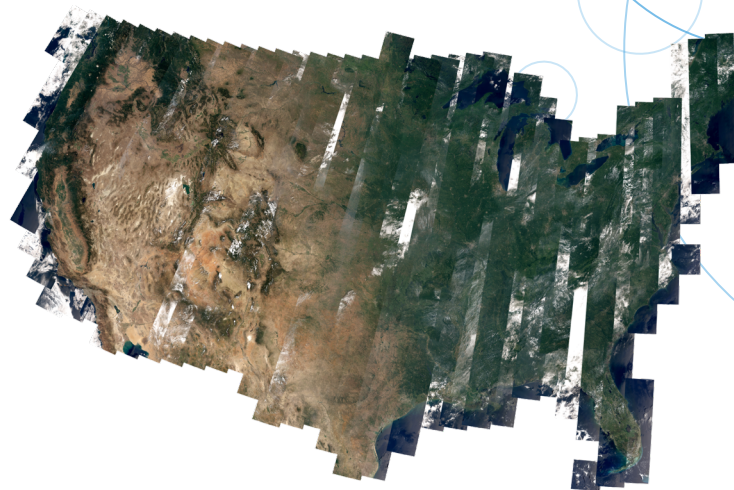


Earth Search

<https://earth-search.aws.element84.com/stac>

Public Search and Discovery API for Earth on AWS Datasets

- Central search catalog and standardized metadata
- Single API that allows searching of STAC metadata for all Earth on AWS datasets
- Powered by sat-api (<https://github.com/sat-utils/sat-api>)
- Backed by an Elasticsearch index
- Initial datasets:
 - Sentinel-2
 - CBERS-4
 - Landsat-8



Landsat-8

Landsat is perhaps one of the most oft-cited and used EO datasets because of its historical archive (Landsat-8 was launched in 2013, but the entire Landsat archive goes back to 1982, with some caveats).

CMR STAC API

API on top of NASA's **Common Metadata Repository (CMR)** that follows the exciting upcoming **SpatioTemporal Asset Catalog (STAC)** and **Web Feature Service 3.0 (WFS)** specifications.

We saw that these developing APIs are going to make geospatial data easier to access for everyone with the simplicity REST + JSON. Connecting these specifications with the CMR exposes thousands of collections with 300M+ granules of NASA data through an easy to use API.

We're excited about the future of these APIs and improving geospatial data access and usability. **Contact us** to find out more and make suggestions.

CHECK OUT CMR STAC API

 swagger

<https://cmr-stac-api.dev.element84.com/docs/swagger.yaml>

STAC API integrated with OGC Web Feature Service proxy Metadata Repository 0.5.1 OAS3

<https://cmr-stac-api.dev.element84.com/docs/swagger.yaml>

This is an implementation of the SpatioTemporal Asset Catalog API that also implements the OGC Web Feature Service 3.0 specification as a proxy for NASA's

[Jason Gilman - Element 84 - Website](#)

[Send email to Jason Gilman - Element 84](#)

[CC-BY 4.0 license](#)

Server

<https://cmr-stac-api.dev.element84.com> ▼

Capabilities Essential characteristics of this API including information about the data.

GET / landing page of this API

GET /**conformance** information about standards that this API conforms to

GET /**/collections** Describe the feature collections in the dataset.

GET /**/collections/{collectionId}** describe the {collectionId} feature collection

Features Access to data (features).

GET /**/collections/{collectionId}/items** retrieve features of feature collection {collectionId}

Creating STAC Metadata

A series of white geometric shapes, including circles and arcs, are positioned in the upper right corner of the slide. These shapes vary in size and overlap, creating a modern, abstract design against the solid green background.

sat-stac

<https://github.com/sat-utils/sat-stac>

- Python library for creating or working with STAC catalogs, collections, and items
- Used for creating existing catalogs, such as
 - <https://landsat.stac.cloud>
 - <https://sentinel.stac.cloud>
 - <http://cbers.stac.cloud/>
- See the Jupyter Notebook tutorials

sat-stac-sentinel

<https://github.com/sat-utils/sat-stac-sentinel>

- Sentinel-2(a/b), Sentinel-1 coming
- Index Sentinel scenes on AWS
- Transform Sentinel metadata to STAC (tileInfo.json -> STAC)
- Deployed Lambda function for real-time publishing new Sentinel scenes as STAC

Using STAC and Cloud Optimized Data



sat-search

<https://github.com/sat-utils/sat-search>

- Python library for searching STAC compliant endpoints
- Command Line Interface (CLI) as a general purpose tool
- Save search results to be loaded and used later
- Download specific assets from all items
- Specify specific band colors to download those bands
 - e.g., Download all “red” and “nir” bands from these Landsat and Sentinel Items

```
$ sat-search search --intersects maine.geojson --found
```

```
1674 items found
```

```
$ clear
```

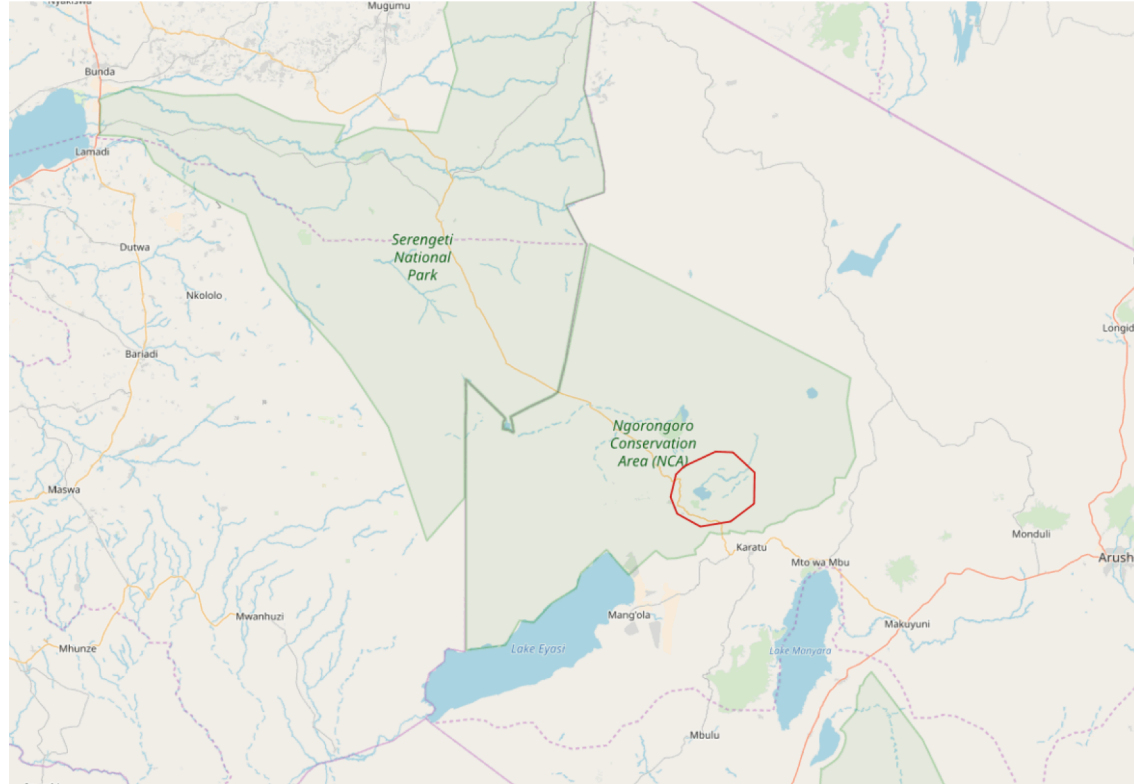
I

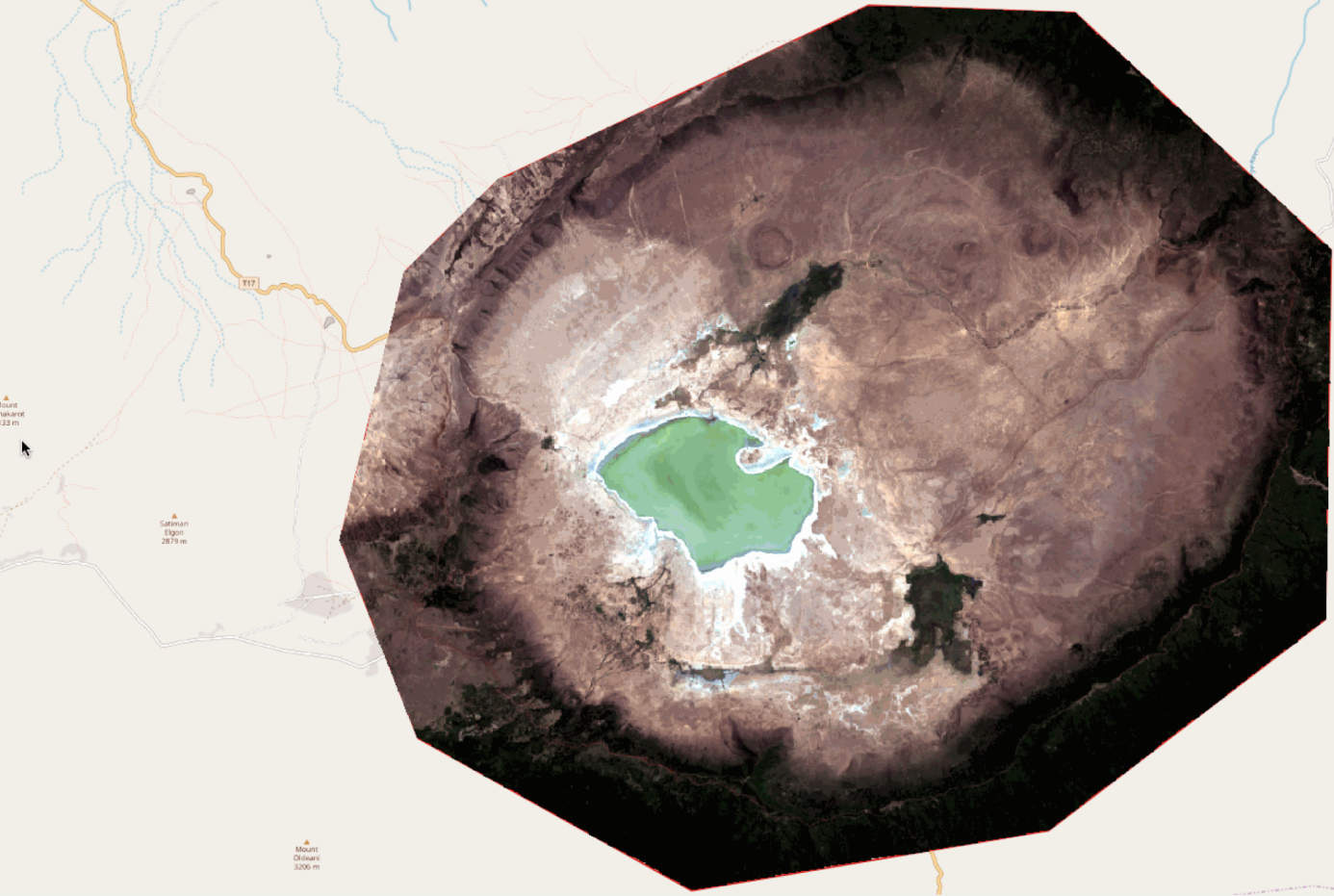
sat-fetch

<https://github.com/sat-utils/sat-fetch>

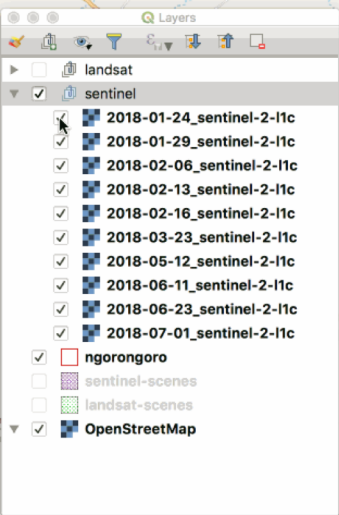
- Works just like sat-search, plus...
- Option to fetch imagery for just AOIs, rather than the entire asset
- Requires GDAL, or use publicly available Docker image

Ngorongoro crater: landsat-8 and sentinel-2





```
$ time sat-fetch load landsat-scenes.geojson \  
  --download red green blue  
real 0m59.674s
```



```
$ time sat-fetch load sentinel-scenes.geojson \  
  --download red green blue  
real 8m6.525s
```

How can you be part of this?

If you're an end user:

- Use the tools – provide feedback!
- Use STAC metadata – provide feedback!

If you're an expert in a data type:

- Provide input on the extensions
- If we don't have an extension for it, help write it

If you're a data provider:

- Surface STAC metadata, reach out to someone hosting a catalog and index your data
- Make STAC relevant parameters available in your native metadata
- Consider cloud optimized representations of your data

Join STAC Sprint #5 + OGC API — Features hackathon November 5–7



Chris Holmes [Follow](#)

Sep 9 · 4 min read

tl;dr: *[Signup](#) for the Joint STAC + OGC API Sprint, taking place November 5–7 at the IQT CosmiQ Works office in Arlington, VA. Virtual participants and new community members welcome!*

The longer version

The past couple of years has seen some major steps forward on geospatial interoperability. The trend in OGC towards open collaboration, JSON + REST focus, and OpenAPI specs that started with WFS 3 is sweeping

<https://bit.ly/312QjEG>

Thank you

<https://github.com/radianteearth/stac-spec>

WGISS-48, Hanoi Vietnam

Dan Pilone
dan@element84.com || @danpilone