Using ESS-DIVE's API for Dataset Search and Aggregate Download

Madison Burrus

Computer Systems Engineer

ESS-DIVE | BERKELEY LAB | NERSC | NCEAS | DataONE | U.S. DEPARTMENT OF ENERGY | Office of Science

September 2022 Webinar

Objectives

Part 1: Search and download both data and metadata with the API **+ demo**

Part 2: Create dataset reports for projects **+ demo**

- Review permission structure to search for private datasets

Part 3: Download and visualize data files with the API **+ demo**

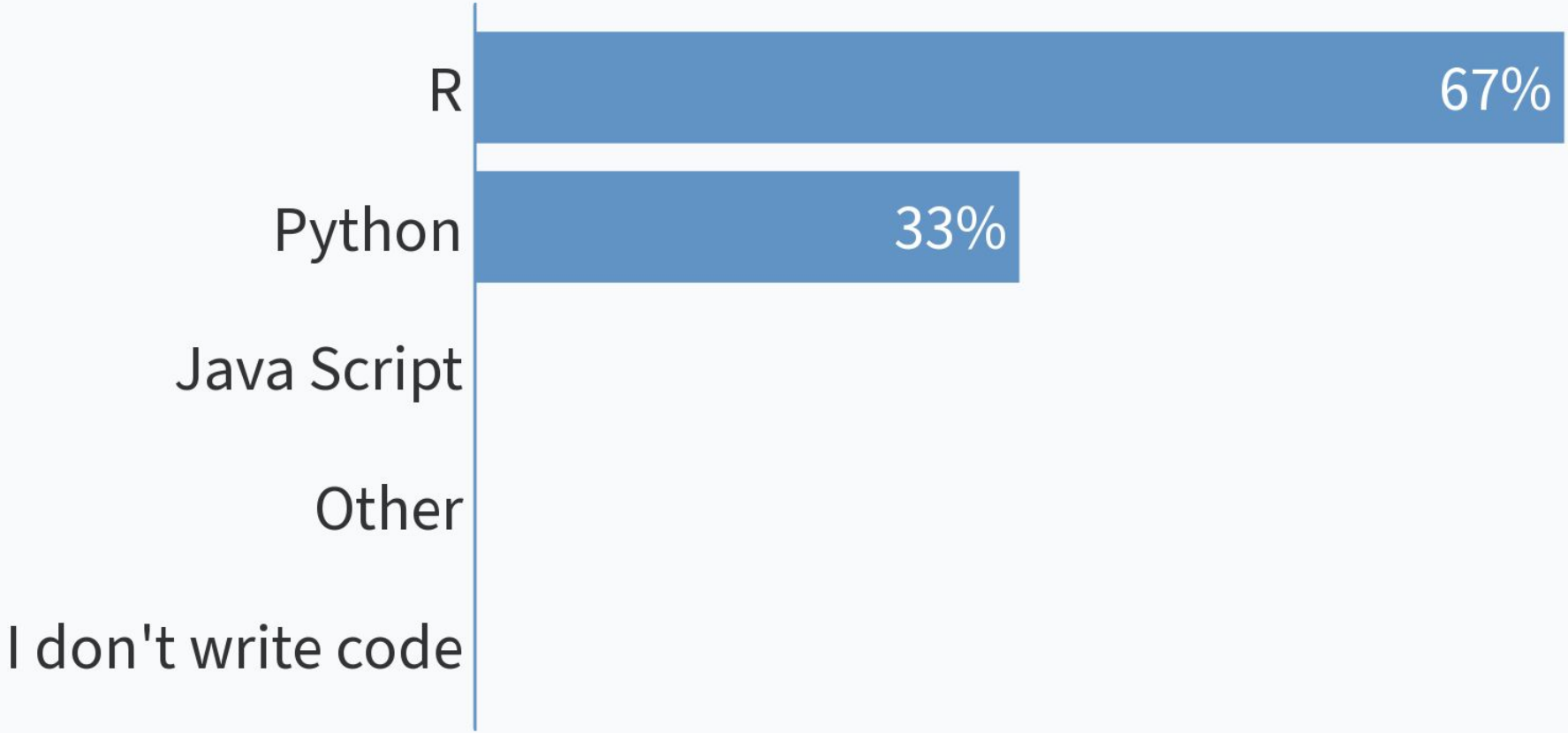
Q/A & Optional review of API data submissions with the API

This webinar is designed for Data Users, Data Managers and Software Developers

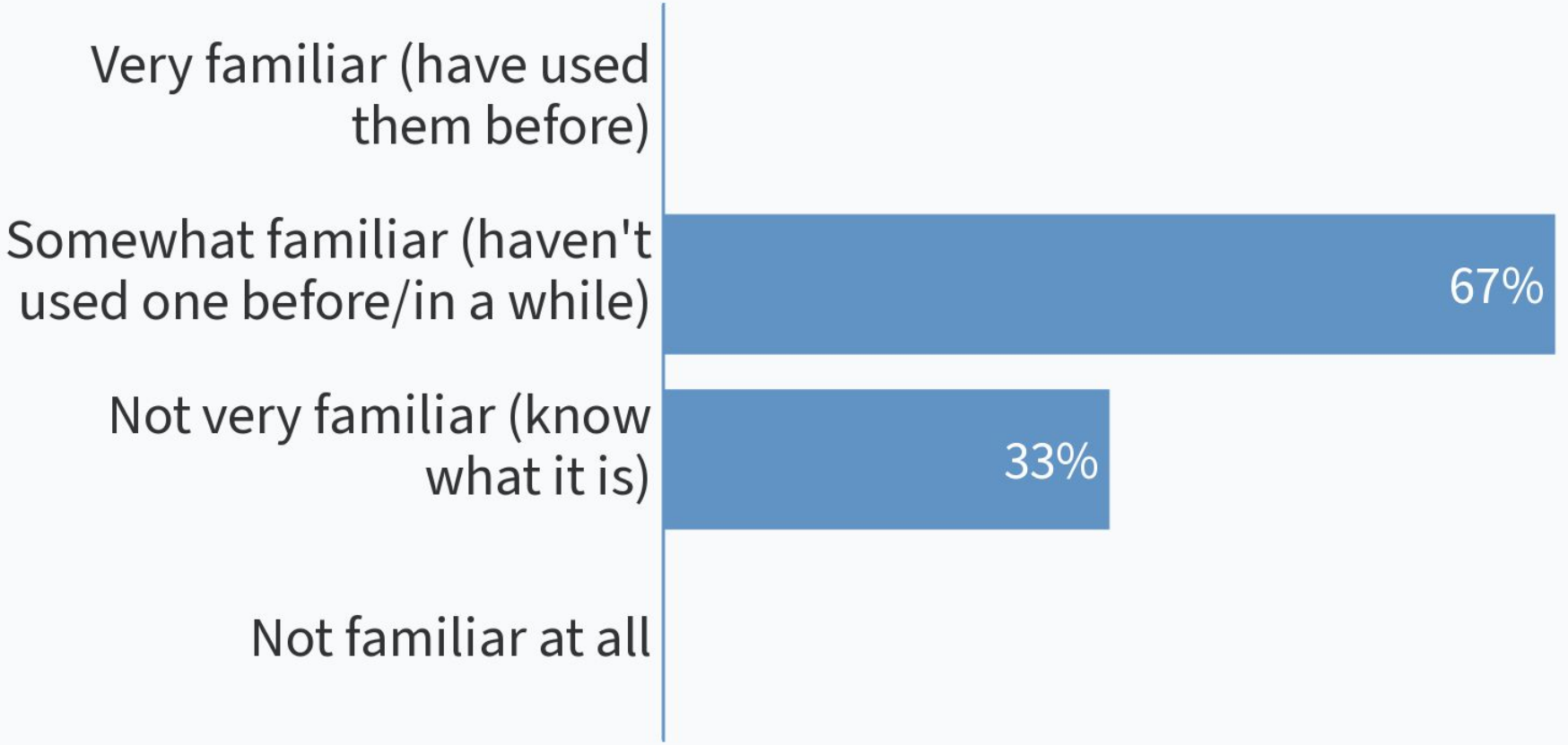
Important Terminology

Term	Definition
ORCID Account	Persistent identifier for researchers – necessary to login to ESS-DIVE
Registered Data Submitter	Someone who can submit or edit datasets, create portals, or view private datasets
REST API	A web API that conforms to the constraints of REST architectural style and allows for interaction with RESTful web services
JSON LD (and other Schemas)	Defines how dataset metadata is organized in ESS-DIVE and is used in the Dataset API to interact with datasets

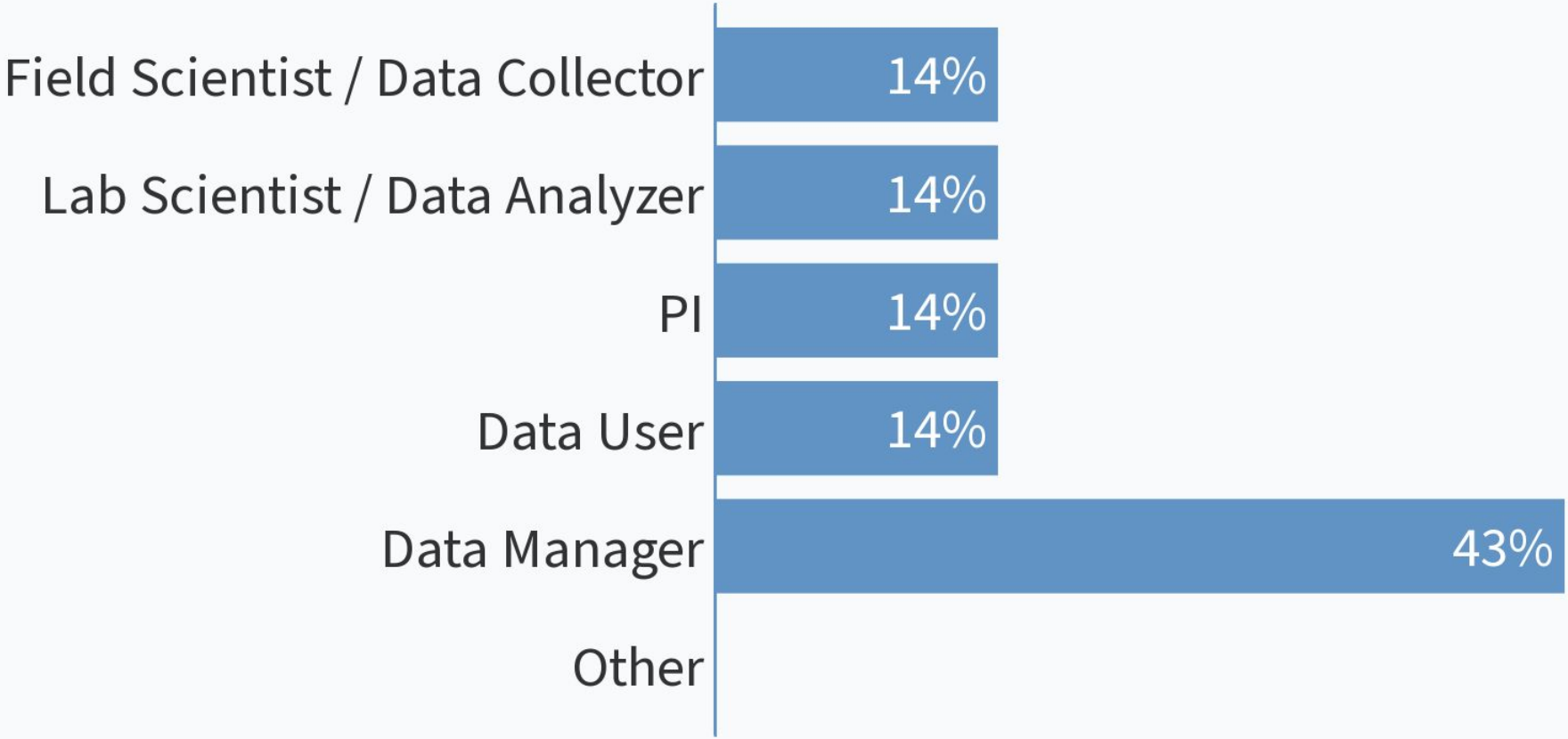
Which coding language(s) do you prefer using?



How familiar are you with APIs or web services?

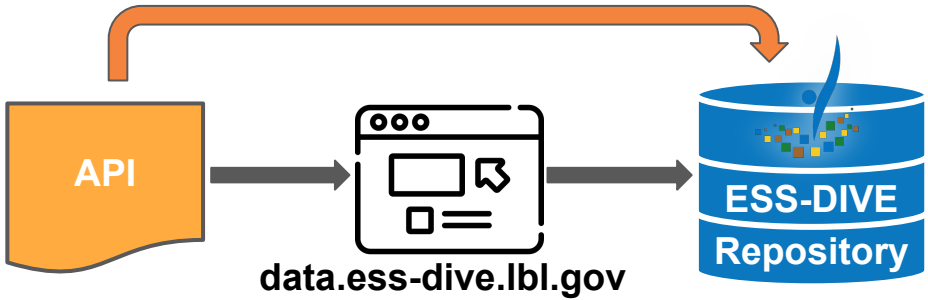


What is your primary role in your research group?



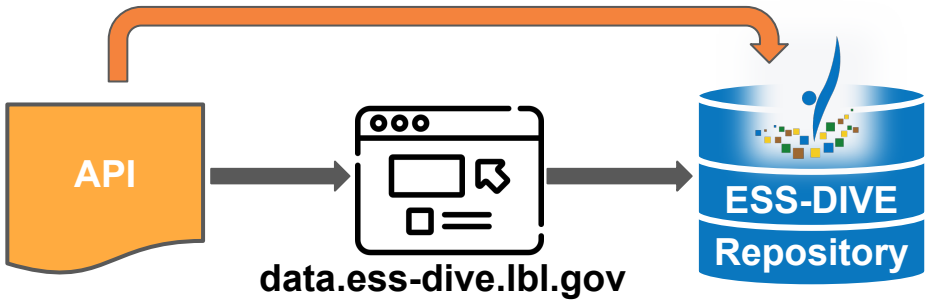
What is the Dataset API?

The Dataset Application Programming Interface (API) **helps data providers** create and collaborate on datasets and **data users** find and download datasets faster and easier.



What is the Dataset API?

It does this by allowing data users and submitters **a way to access ESS-DIVE's features by writing scripts** rather than manually through a browser.



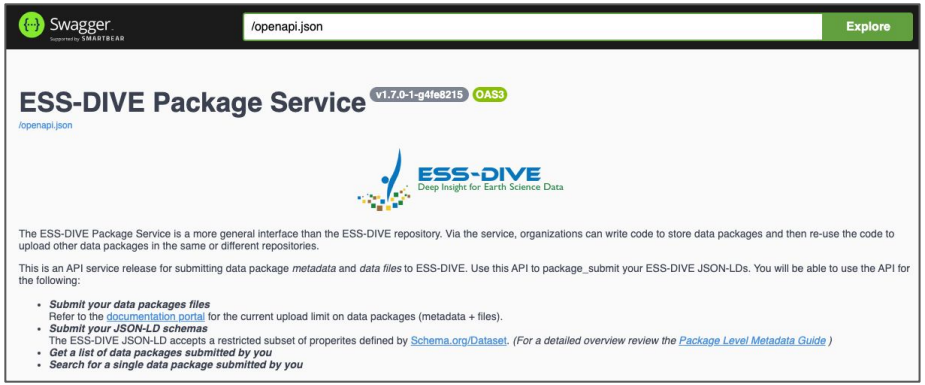
What is the Dataset API?

- The more datasets you want to work with, the more beneficial it is to use the Dataset API – so **it is a great asset to projects**
- **Pick it up quickly** – example code, Jupyter Notebooks, documentation, and live support are all available
- **What can it do exactly?**
  - Create and edit datasets
  - Upload data files more efficiently
  - Give registered submitters permission to edit your datasets
  - **Search and download datasets**

Background Needed to use the API

Some familiarity with the following will help you get started:

- **REST APIs** and reading **API documentation**



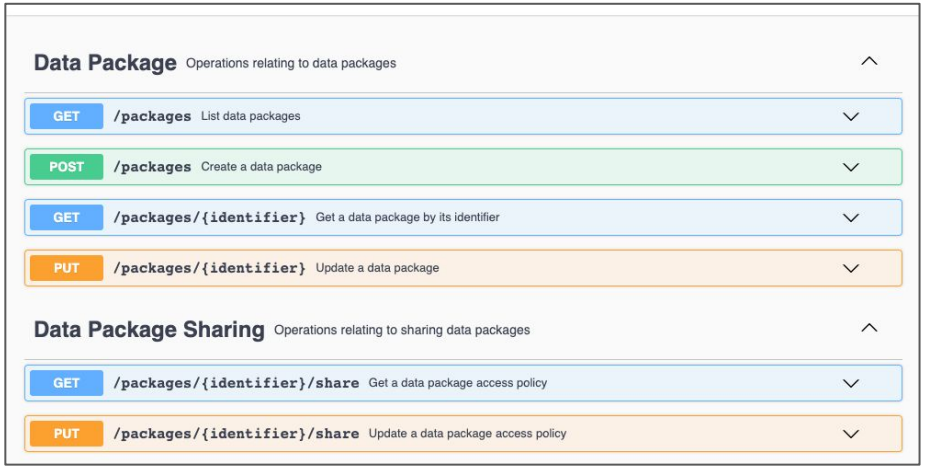
ESS-DIVE uses Swagger Open API to document the Dataset API service

Understanding API capabilities will help enable you to better use ESS-DIVE's features

Background Needed to use the API

Some familiarity with the following will help you get started:

- **REST APIs** and reading **API documentation**
- Running **HTTP operations**



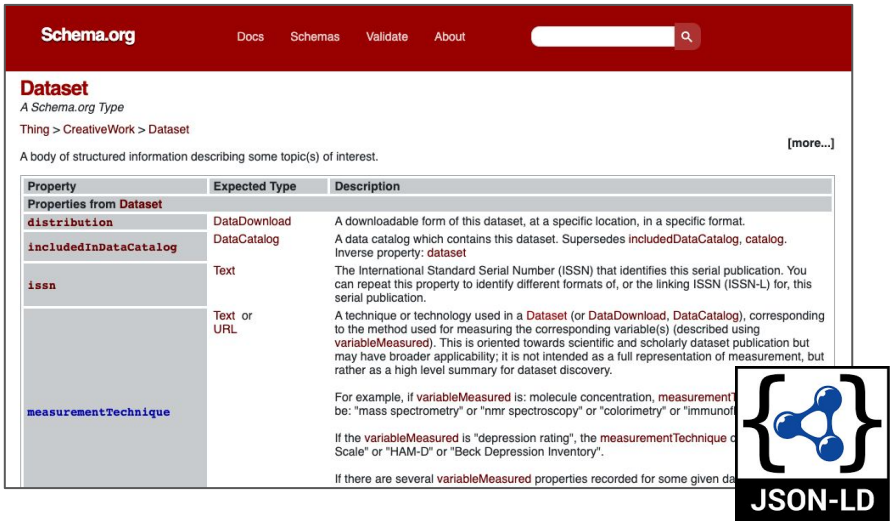
List of the Dataset API's available HTTP operations

Understanding API capabilities will help enable you to better use ESS-DIVE's features

Background Needed to use the API

Some familiarity with the following will help you get started:

- **REST APIs** and reading **API documentation**
- Running **HTTP operations**
- **JSON LD** or other **schemas**



The Dataset API uses the "Dataset" schema defined by Schema.org

Understanding API capabilities will help enable you to better use ESS-DIVE's features

Multiple coding languages to choose from

You can write code to the Dataset API service in **three coding languages**.

- Python
- R
- Java Script

Code examples of all Dataset API operations are available in each language on **ESS-DIVE's API Guide** *(https://docs.ess-dive.lbl.gov/programmatic-tools/ess-dive-dataset-api)*.

For this presentation, all examples are in **Python**

Part 1: Search for Datasets with the API

Searching with the API is now more advanced

The API now allows **anyone** to search through **all public datasets** on ESS-DIVE based on specific search terms you define.

Search for datasets by:

- **Dataset Creator**
- **Publication Date**
- **Project Name**
- **Text**

Search Query Examples

The API can find all datasets…

- Published by the **NGEE-Tropics Project**
- Published by the NGEE-Tropics Project in **2021**
- Related to **groundwater** or **CO2 data**
- Where Jane Doe is an **author**
- Containing the **exact phrase** “Seward Peninsula”

How is searching with the API different?

ESS-DIVE Website

- **8** filter options
- **Download one** dataset at a time
- Search by location using **map view**
- **Anonymously** search or download datasets

Dataset API

- **5** filter options
- **Download many** datasets automatically
- Easily **compare and summarize** datasets in results
- **ORCID/login required** to search or download datasets

API Search Process

Setup/Authenticate

01

Login to data.ess-dive.lbl.gov and copy your authentication token

Enter Parameters

02

Format and enter search parameters into the HTTP request

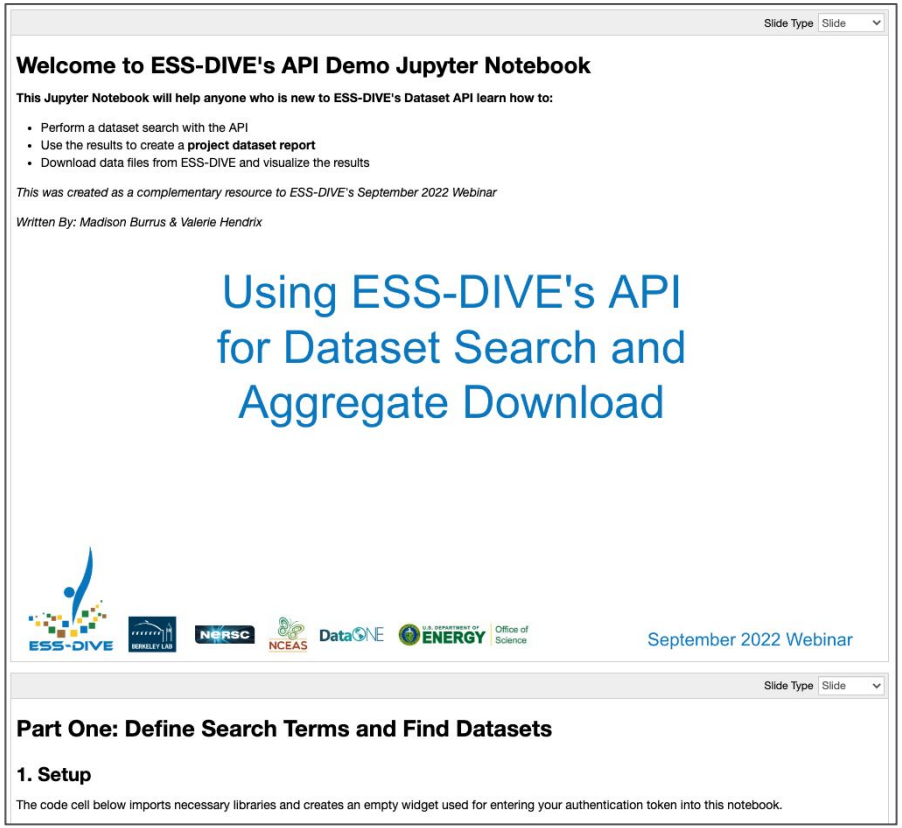
Send Request

03

Send your search query to ESS-DIVE and review results

Results will be returned in JSON

Search with API Demo



Slide Type Slide

Welcome to ESS-DIVE's API Demo Jupyter Notebook

This Jupyter Notebook will help anyone who is new to ESS-DIVE's Dataset API learn how to:

- Perform a dataset search with the API
- Use the results to create a **project dataset report**
- Download data files from ESS-DIVE and visualize the results

This was created as a complementary resource to ESS-DIVE's September 2022 Webinar

Written By: Madison Burrus & Valerie Hendrix

Using ESS-DIVE's API for Dataset Search and Aggregate Download

ESS-DIVE · BERKELEY LAB · NERSC · NCEAS · DataONE · U.S. DEPARTMENT OF ENERGY Office of Science

September 2022 Webinar

Slide Type Slide

Part One: Define Search Terms and Find Datasets

1. Setup

The code cell below imports necessary libraries and creates an empty widget used for entering your authentication token into this notebook.

ESS-DIVE's API Search Demo

Jupyter Notebook can be downloaded from:

https://github.com/ess-dive/essdive-package-service-examples

1. Setup

The code cell below imports necessary libraries and creates an empty widget used for entering your authentication token into this notebook.

1. Run cell, leave the token widget empty and continue.

```
# Import libraries
import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)
import pandas
import requests
import os
from ipywidgets import widgets, interact
from IPython.display import display

# Setup the inputs
token_text = widgets.Text("", description="Token:")
display(token_text)

```

Token:

2. Go to ESS-DIVE (https://data.ess-dive.lbl.gov/data), login with your ORCID, and copy your authentication token from your account settings page.
3. Enter your authentication token into the widget above
4. Run the following code cell

Always re-run this code cell when you update your token. Tokens expire every 24 hours.

```
token=token_text.value
essdive_api_url='https://api.ess-dive.lbl.gov'
```

The example below performs a search for all **public datasets published under a specific project**.

5. Optional: Go to ESS-DIVE's project list to find the controlled name for your project
6. Define your search parameters here

```
# Define parameter variables
project="Next-Generation Ecosystem Experiments (NGEE) Arctic"
creator="<creator/submitter name>"
text= "<enter search term>"
datePublished = "<[YYYY TO YYYY-MM-DD]>"
```

7. Enter any additional parameters you'd like to the `get_packages_response` string in the format of: `datePublished=\"{datePublished}\"`

Note: When using the `text` parameter, use quotations in your search to locate exact text matches

```
# Contruct URL query to send to the Dataset API
get_packages_response = f"{essdive_api_url}/packages?providerName=\"{project}\"&isPublic=tr
```

3. Run code & Debug

8. Run the following code cell
9. Continue when you receive a "Success!" response
10. Stop and debug the issue if you receive an error.

```
# Send request to Dataset API
response = requests.get(get_packages_response,
                        headers={"Authorization": f"Bearer {token}"})

# Review the response and debug if needed
if response.status_code == 200:
    # Success
    response_json = response.json()
    print("Success! Continue to look at the search results")

else:
    # There was an error
    print("There was an error. Stop here and debug the issue. Email ess-dive-support@lbl.go
    print(response.text)
```

```
Success! Continue to look at the search results
```

4. Look at Search results

11. Run the following cell and take a look at the JSON response. We will parse this output and do more with it in the next section of this notebook.

```
print("This is what a dataset search JSON result looks like with the API. \n")
display(response_json)
```

```
This is what a dataset search JSON result looks like with the API.


{'total': 8,
 'user': 'http://orcid.org/0000-0002-3512-5992',
 'query': {'isPublic': True,
  'providerName': '"Next-Generation Ecosystem Experiments (NGEE) Arctic"'},
 'pageSize': 25,
 'rowStart': 1,
 'result': [{'id': 'ess-dive-3ae4bf47c8a5c01-20220728T133818501',
   'viewUrl': 'https://data.ess-dive.lbl.gov/view/doi:10.5440/1575068',
   'dateUploaded': '2022-07-28T13:38:19.010Z',
   'dateModified': '2022-07-28T13:38:22.954Z',
   'isPublic': True,
   'citation': 'Rogers A; Alldred M; Serbin S (2020): Leaf Carbon and Nitrogen Content, Sewa
rd Peninsula, Alaska, 2014. Next-Generation Ecosystem Experiments (NGEE) Arctic. doi:10.544
0/1575068',
   'dataset': {'@context': 'http://schema.org/',
    '@type': 'Dataset',
    '@id': 'doi:10.5440/1575068'
```

Any Questions?

Photo by Towfiqu barbhuiya on Unsplash

API Help

Technical API Docs: api.ess-dive.lbl.gov/#/Data%20Package/listPackages

Jupyter Notebook Demo: github.com/ess-dive/essdive-package-service-examples

How-To Guide: docs.ess-dive.lbl.gov/programmatic-tools/ess-dive-dataset-api

Example Scripts: github.com/ess-dive/essdive-package-service-examples

Permission Structures

Searching for private datasets

Public Searches

Requirement: Login to ESS-DIVE with **ORCID**

- Set `isPublic` parameter to `true`

Private Searches

Requirement: Register to be a **data submitter** with ESS-DIVE

- Set `isPublic` parameter to `false`
- Returns private and public
- See private datasets that:
  - You created
  - Were shared with you

What's next?

What can we do now that we're able to search for datasets with the API?

Part 2:

Summarize and compare datasets in the search results

Part 3:

Download, explore, and visualize data

Part 2: Create Dataset Reports

What are Dataset Reports?

Reports are exported dataset search results

- API search results can be easily printed
- We can select the information we want to print and export it as a csv

Simple report example: a table of all datasets published by one project

	Public	Title	Publication Date	URL
0	True	Leaf Carbon and Nitrogen Content, Seward Peninsula, Alaska, 2014	2020	https://data.ess-dive.lbl.gov/view/doi:10.5440/1575068
1	True	Leaf Photosynthetic Parameters: Quantum Yield, Convexity, Respiration, Gross CO2 Assimilation Rate and Raw Gas Exchange Data, Utqiagvik (Barrow), Alaska, 2016	2019	https://data.ess-dive.lbl.gov/view/doi:10.5440/1482338
2	True	NGEE Arctic Plant Traits: Fine Roots, Kougarok Road Mile Marker 64, Seward Peninsula, Alaska, 2016	2021	https://data.ess-dive.lbl.gov/view/doi:10.5440/1735941
3	True	Active Layer Hydrology in an Arctic Tundra Ecosystem: Quantifying Water Sources and Cycling Using Water Stable Isotopes: Supporting Data	2019	https://data.ess-dive.lbl.gov/view/doi:10.5440/1164892
4	True	Radiocarbon in CO2 and Soil Organic Matter from Laboratory Incubations, Utqiagvik (Barrow), Alaska, 2012	2018	https://data.ess-dive.lbl.gov/view/doi:10.5440/1418852
5	True	Subsurface Flow Across Polygonal Tundra Measured by Bromide Tracer, Utqiagvik (Barrow), Alaska, 2015-2016	2020	https://data.ess-dive.lbl.gov/view/doi:10.5440/1342954
6	True	Alaskan carbon-climate feedbacks will be weaker than inferred from short-term manipulations: Alaskan Benchmark Data and Model runs	2020	https://data.ess-dive.lbl.gov/view/doi:10.15485/1670465
7	True	Active Layer Soil Carbon and Nutrient Mineralization, Barrow, Alaska, 2012	2018	https://data.ess-dive.lbl.gov/view/doi:10.5440/1185213

Why create dataset reports?

- Easily **summarize information** in search results
  - Answer questions like: “how many of these datasets were published in 2021?”
- Export your report as a CSV and send it to colleagues
  - Generate **annual reports** of your project’s publication progress
- Update reports by **reusing code**
  - Reduces repetitive manual effort

How to create a report

Step 1: Review search results

- The API prints search results in **JSON LD** format
- It will print 8 fields that describe each dataset

Slide Type Fragment

```
1 print("This is what a dataset search JSON result looks like with the API. \n")
2 display(response_json)
```

```
This is what a dataset search JSON result looks like with the API.


{'total': 76,
 'user': 'http://orcid.org/0000-0002-3512-5992',
 'query': {'isPublic': True,
  'providerName': '"Next-Generation Ecosystem Experiments (NGEE) "'},
 'pageSize': 25,
 'rowStart': 1,
 'result': [{'id': 'urn:uuid:99f2b410-565b-4663-be7c-660b056cbb1f',
   'viewUrl': 'https://data-dev.ess-dive.lbl.gov/view/urn:uuid:99f2b410-565b-4663-be7c-660b056cbb1f',
   'dateUploaded': '2022-06-23T23:24:53.720Z',
   'dateModified': '2022-06-23T23:24:53.720Z',
   'isPublic': True,
   'citation': 'Christianson D; Varadharajan C; Christofferson B; Detto M; Faybishenko B; Hendrix V; Jardine K; Negro
n-Juarez R; Gimenez B; Pastorello G; Powell T; Sandesh M; Warren J; Wolfe B; Chambers J; Kueppers L; McDowell N; Agar
wal D (2017): FRAMES Metadata Reporting Templates for Ecohydrological Observations, version 1.1. Next-Generation Ecos
ystem Experiments (NGEE) Tropics. urn:uuid:99f2b410-565b-4663-be7c-660b056cbb1f',
   'dataset': {'@context': 'http://schema.org/',
    '@type': 'Dataset',
    '@id': None,
    'name': 'FRAMES Metadata Reporting Templates for Ecohydrological Observations, version 1.1',
    'description': 'FRAMES is a a set of Excel metadata files and package-level descriptive metadata that are designe
d to facilitate and improve capture of desired metadata for ecohydrological observations. The metadata are bundled wi
th data files into a data package and submitted to a data repository (e.g. the NGEE Tropics Data Repository) via a we
b form. FRAMES standardizes reporting of diverse ecohydrological and biogeochemical data for synthesis across a range
of spatiotemporal scales and incorporates many best data science practices. This version of FRAMES supports observati
ons for primarily automated measurements collected by permanently located sensors, including sap flow (tree water us
e), leaf surface temperature, soil water content, dendrometry (stem diameter growth increment), and solar radiation.
Version 1.1 extend the controlled vocabulary and incorporates functionality to facilitate programmatic use of data an
d FRAMES metadata.',
    'datePublished': '2017'}},
```

How to create a report

8 fields appear in search results:

- Date uploaded
- Date last modified
- Publication Date
- Landing page URL
- Dataset Title
- Citation
- Abstract
- ESS-DIVE dataset IDs (*useful for using in other API operations*)

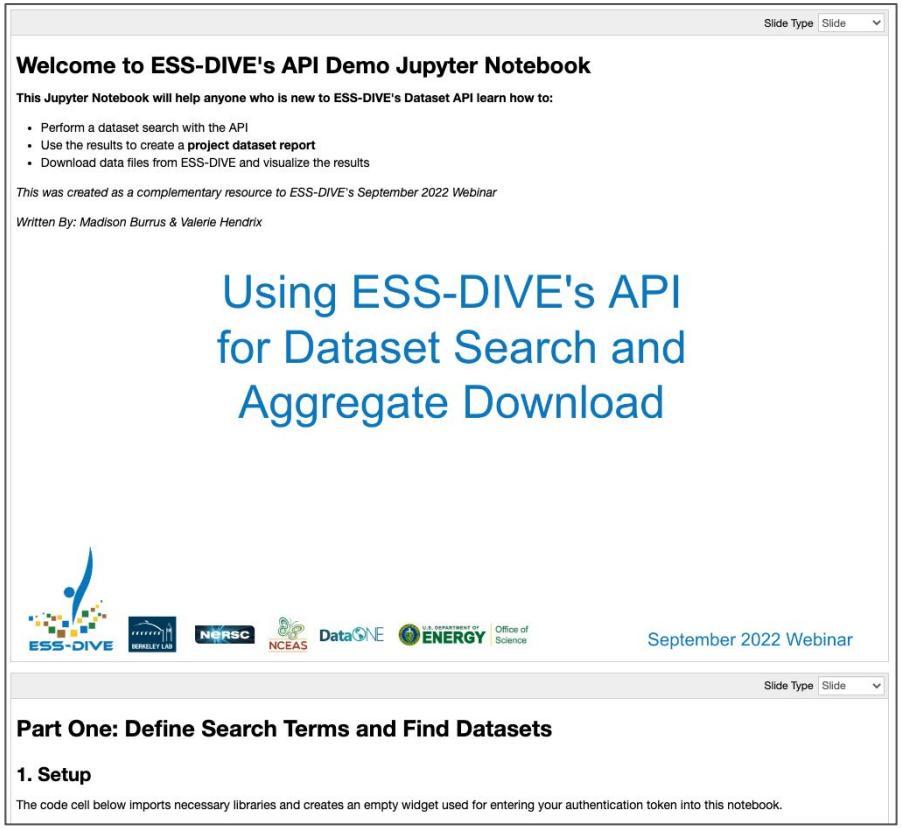
How to create a report

Step 2: Choose fields & create report

Choose the fields you want from the JSON search result, then enter the fields into a data structure for visualization and export

- In this presentation, we use **Pandas** to enter search results into a dataframe and **export it as a CSV**
- You can use any library or data structure to export your report

Create Dataset Report with API Demo



ESS-DIVE's API Search Demo

Jupyter Notebook can be downloaded from:

https://github.com/ess-dive/essdive-package-service-examples

Part Two: Dataset Report for Project

Using the results from Part One, we'll now create a dataset report that includes a list of all datasets published by one project.

1. Review number of results

Let's start by investigating our search results. We'll look at how many results there are because the API does not print more than 100 datasets in the results. If we have more than 100 datasets, we'll need to run another line of code.

1. Run the code cell below and review the number of search results.

```
:
1  print("Total datasets that match search results: ",response_json['total'])
2
3  if response_json['total'] > 100:
4      print("This call cannot print more than 100 results. Change the `pageSize` to 100, \
5      re-run the call then change rowStart to 100 and re-run the call as many times as needed
6      through all, ",response_json['total']," results.")
7
8  else:
9      "do nothing"
10
11 # Grab list of results to pass into function
12 results=response_json['result']
```

```
Total datasets that match search results:  8
```

2. Define pandas dataframe display

In the following code cell, we're defining a function that grabs the metadata fields we want to include in our report from the JSON search result and puts it into a data frame, as well as setting some display settings for our final data frame.

This example report will include the following fields:

- Landing page URL
- Dataset Title
- Publication Date

```
# Function to define what we want in the dataframe
def grab_metadata(r_json):
    df = pandas.DataFrame()

    for dataset in r_json:
        url = dataset['viewUrl']
        access = dataset['isPublic']
        title = dataset['dataset']['name']
        publicationDate = dataset['dataset']['datePublished']
        report={'Public':access, 'Title':title, 'Publication Date':publicationDate, 'URL':u
        df = df.append(report, ignore_index=True)

    return df

# Change dataframe display options to better visualize the results
pandas.set_option('display.max_columns', None)
pandas.set_option('display.max_rows', None)
pandas.set_option('display.max_colwidth', None)
pandas.set_option('display.colheader_justify', 'left')
```

3. Create, Summarize and Display Dataframe

Using the function defined above, we'll quickly create a dataframe using the cell below

```
# Create dataframe for the report
project_report=(grab_metadata(results))
```

Next, we'll summarize information of interest. In this case, we chose to look at the range of publication dates for our list of datasets

```
no_pub_dates=project_report['Publication Date'].unique()
display(no_pub_dates)
```

```
array(['2020', '2019', '2021', '2018'], dtype=object)
```

Finally, let's see what this report looks like

```
# Display dataframe
display(project_report.style.set_properties(**{'text-align': 'left'}))
```

	Public	Title	Publication Date	URL
0	True	Leaf Carbon and Nitrogen Content, Seward Peninsula, Alaska, 2014	2020	https://data.ess-dive.lbl.gov/view/doi:10.5440/1575068
1	True	Leaf Photosynthetic Parameters: Quantum Yield, Convexity, Respiration, Gross CO2 Assimilation Rate and Raw Gas Exchange Data, Utqiagvik (Barrow), Alaska, 2016	2019	https://data.ess-dive.lbl.gov/view/doi:10.5440/1482338
2	True	NGEE Arctic Plant Traits: Fine Roots, Kougarok Road Mile Marker 64, Seward Peninsula, Alaska, 2016	2021	https://data.ess-dive.lbl.gov/view/doi:10.5440/1735941
3	True	Active Layer Hydrology in an Arctic Tundra Ecosystem: Quantifying Water Sources and Cycling Using Water Stable Isotopes: Supporting Data	2019	https://data.ess-dive.lbl.gov/view/doi:10.5440/1164892
4	True	Radiocarbon in CO2 and Soil Organic Matter from Laboratory Incubations, Utqiagvik (Barrow), Alaska, 2012	2018	https://data.ess-dive.lbl.gov/view/doi:10.5440/1418852

4. Export dataframe as a csv

Now we'll save our output by running the line of code below

```
# Define file name, directory name, and file paths
file_name = 'project_report_example.csv'
new_dir = 'ESS-DIVE_Downloads'
parent_dir = os.getcwd()
path = os.path.join(parent_dir, new_dir)
file_path=os.path.join(path, file_name)

# attempt to make the new folder if it doesn't already exist
try:
    os.mkdir(path)
    print("Directory '% s' created" % new_dir)
except OSError as error:
    print(error)
```

```
Directory 'ESS-DIVE_Downloads' created
```

```
try:
    project_report.to_csv(file_path, sep=',', index=False)
    print("Your dataset report has been saved!")
except:
    print("Save failed.")
```

```
Your dataset report has been saved!
```

Any Questions?

Photo by Towfiqu barbhuiya on Unsplash

Part 3: Download and Visualize Data Files

Next steps required to download data files

Search queries (from Part 1) give a sense of data availability and print basic dataset information

- Basic info can be used to create a dataset report

To find and **download data files** we need to lookup datasets **individually**

Next steps required to download data files

To download data files, we'll need to:

1. Get **dataset IDs** from search results
2. Send **another HTTP request** to ESS-DIVE that prints entire dataset
3. View data files **one dataset at a time**
    - Scripts can be written to loop through all datasets and download data automatically

Warning: Data file size is not shown before download

Get Dataset IDs from search results

Dataset IDs can be found in the output of dataset **search queries**

```
1 print("This is what a dataset search JSON result looks like with the API. \n")
2 display(response_json)
```

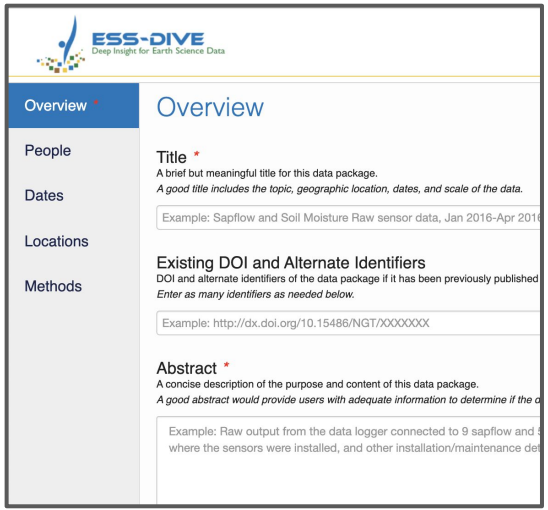
```
This is what a dataset search JSON result looks like with the API.


{'total': 8,
 'user': 'http://orcid.org/0000-0002-3512-5992',
 'query': {'isPublic': True,
  'providerName': '"Next-Generation Ecosystem Experiments (NGEE) Arctic"'},
 'pageSize': 25,
 'rowStart': 1,
 'result': [{'id': 'ess-dive-3ae4bf47c8a5c01-20220728T133818501',
   'viewUrl': 'https://data.ess-dive.lbl.gov/view/doi:10.5440/1575068',
   'dateUploaded': '2022-07-28T13:38:19.010Z',
   'dateModified': '2022-07-28T13:38:22.954Z',
   'isPublic': True,
   'citation': 'Rogers A; Alldred M; Serbin S (2020): Leaf Carbon and Nitrogen Content, Sewa
rd Peninsula, Alaska, 2014. Next-Generation Ecosystem Experiments (NGEE) Arctic. doi:10.544
0/1575068',
   'dataset': {'@context': 'http://schema.org/',
    '@type': 'Dataset',
    '@id': 'doi:10.5440/1575068'
```

This is the dataset ID

Results from an individual dataset search

Individual dataset searches will return **all metadata fields and data files** associated with that dataset

- Each field in the JSON LD result corresponds to a field on the dataset landing page



Get data files from individual dataset

Data files are found in the output of an **individual dataset request**


```
 'measurementTechnique': ["Carbon and nitrogen content of leaves sampled from locations on the Council, Kougarok and
Teller roads, Seward Peninsula, Alaska. Species not fully identified in all cases, but include samples from the gener
a: Alnus, Arctagrostis, Arctophila, Arctostaphylos, Betula, Carex, Eriophorum, Petasites, Picea, Populus, Rubus, Sali
x, Saxifraga and Vaccinium. Locations along the Council, Kougarok and Teller roads, Seward Peninsula, Alaska. Samples
were collected from various locations as part of reconnaissance field work prior to establishment of formal NGEE Arct
ic study sites. Dried, ground leaf material was analyzed using a Perkin Elmer 2400 Series II CHNS/O Elemental Analyze
r, according to manufacturer's instructions. http://www.perkinelmer.com"},
 'distribution': [{'contentUrl': 'https://data.ess-dive.lbl.gov/catalog/d1/mn/v2/object/ess-dive-b11df853638a8f6-2022
0520T180414357556',
   'encodingFormat': 'application/pdf',
   'identifier': 'ess-dive-b11df853638a8f6-20220520T180414357556',
   'name': 'nga205_User_Guide_20220519.pdf'},
  {'contentUrl': 'https://data.ess-dive.lbl.gov/catalog/d1/mn/v2/object/ess-dive-3ae4bf47c8a5c01-20220728T133818501',
   'encodingFormat': 'https://eml.ecoinformatics.org/eml-2.2.0',
   'identifier': 'ess-dive-3ae4bf47c8a5c01-20220728T133818501',
   'name': 'Leaf_Carbon_and_Nitrogen_Content_Seward.xml'},
  {'contentUrl': 'https://data.ess-dive.lbl.gov/catalog/d1/mn/v2/object/ess-dive-0b1fcc5aeacbe24-20220426T19505823826
2',
   'encodingFormat': 'text/csv',
   'identifier': 'ess-dive-0b1fcc5aeacbe24-20220426T195058238262',
   'name': 'NGEEArctic_Seward_CN_2014.csv'}]}
```


Data files can be found under `distribution`

Visualize Data

Once you have the data, you can use your preferred data structures and visualization tools

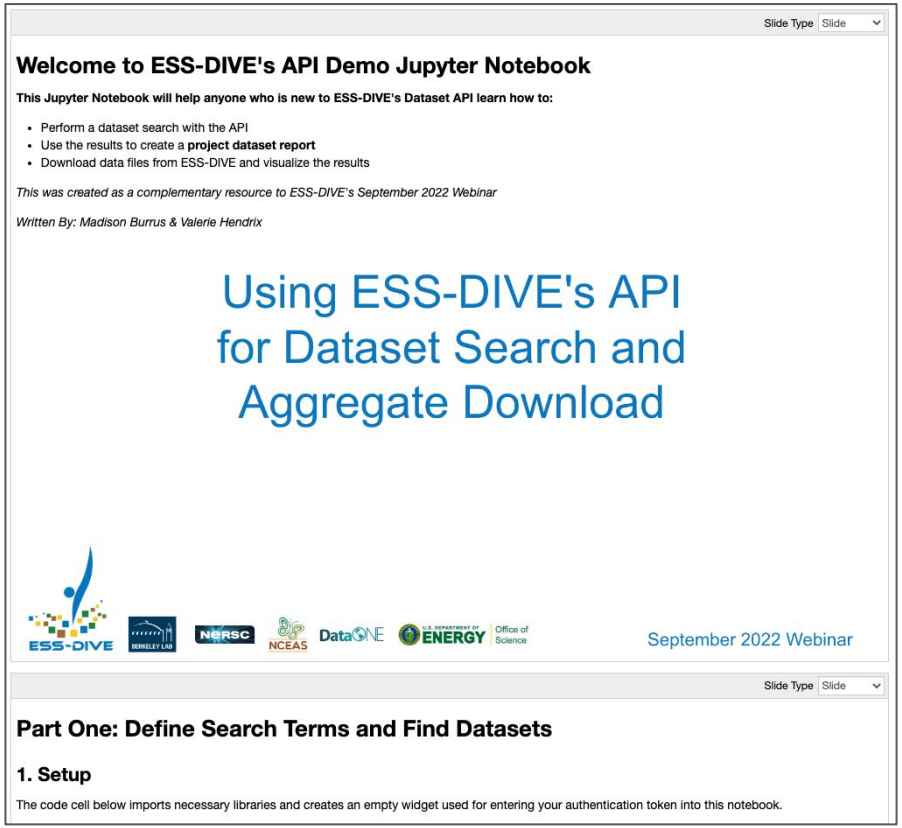
In this example, we:

- Used only csv files and dataframes
- Summarized data by species and field collection site

Site	C:N			C (%)			N (%)		
	min	max	mean	min	max	mean	min	max	mean
Cabins (DOT)	21.355	21.355	21.355000	44.47	44.47	44.470000	2.43	2.43	2.430000
Council Thermokarst Site	18.727	78.628	36.459286	44.15	51.12	47.382143	0.75	2.97	1.720000
Drive up Solifluction lobes	22.357	43.337	30.088857	38.92	48.56	45.422857	1.31	2.31	1.822857
Grand Central River	19.276	19.276	19.276000	44.62	44.62	44.620000	2.70	2.70	2.700000
Guy Rowe Creek	19.645	92.630	38.106083	39.96	50.13	47.009167	0.63	2.76	1.682500
High Hummocky Site	21.208	51.288	34.412143	40.56	48.52	44.384286	0.92	2.48	1.681429
Just outside Teller	20.785	20.785	20.785000	45.22	45.22	45.220000	2.54	2.54	2.540000
Korean Site	47.378	47.378	47.378000	50.99	50.99	50.990000	1.26	1.26	1.260000
Kougarok River Bridge	21.335	84.942	42.227857	34.52	50.80	43.910714	0.62	1.92	1.368571

	Date	Road	Site	Species	sample ID	C (%)	N (%)	C:N	C (mg gDW-1)	N (mg gDW-1)	Notes	Addional site notes
0	2014-08-22	Kougarok Road	Kougarok River Bridge	Betula nana	48	48.29	1.14	49.606	482.9	11.4	NaN	NaN
1	2014-08-22	Kougarok Road	Kougarok River Bridge	Petasites	49	41.62	1.30	37.390	444.7	24.3	NaN	NaN
2	2014-08-22	Kougarok Road	Cabins (DOT)	Balsam poplar	50	44.47	2.43	21.355	444.7	24.3	NaN	NaN
3	2014-08-22	Kougarok Road	Norby's Pond	Salix	51	48.98	2.13	26.876	489.8	21.3	NaN	NaN
4	2014-08-22	Kougarok Road	Shawn's Slope	Petasites	52	42.12	1.45	33.936	421.2	14.5	NaN	where group split

Data Download with API Demo



ESS-DIVE's API Search Demo

Jupyter Notebook can be downloaded from:

https://github.com/ess-dive/essdive-package-service-examples

1. Grab a dataset ID from search result

For this example, we'll use the Dataset ID from the first dataset in the search result

```
dataset_id = response.json()['result'][0]['id']
```

2. Enter dataset ID into new HTTP request URL

Much like in Part one, we'll add the dataset ID parameter into the URL following a specific format.

To learn how to format an individual dataset search with the Dataset API, visit the techincal documentation at https://api-sandbox.ess-dive.lbl.gov/#/Data%20Package/getPackage.

```
get_package_url=f"{essdive_api_url}/packages/{dataset_id}"
```

3. Send request and debug

```
# Get a CSV file from the data package
dataset_response = requests.get(get_package_url,
                          headers={"Authorization": f"Bearer {token}"})

if dataset_response.status_code == 200:
    # Success
    print("Success! Continue to look at the complete dataset metadata and files")
    dataset_json = dataset_response.json()['dataset']

else:
    # There was an error
    print(dataset_response.text)

```

```
Success!
```

5. Grab the data file URL

```
# Get a CSV file from the dataset
csv_file = [d for d in dataset_json['distribution'] if d['encodingFormat'] == 'text/csv'][0]
print(f"File Name: {csv_file['name']}")
```

```
File Name: NGEEArctic_Seward_CN_2014.csv
```

6. Import data file into Jupyter Notebook

```
# Import CSV file as dataframe
try:
    from urllib.request import Request, urlopen  # Python 3
except ImportError:
    from urllib2 import Request, urlopen  # Python 2

req = Request(csv_file['contentUrl'])
req.add_header('User-Agent', 'Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:77.0) Gecko/20100101 Firefox/77.0')
content = urlopen(req)
```

7. Download data file to local folder

We'll use the same directory we created in Part Two

```
import urllib.request

data_file_path=os.path.join(path, csv_file['name'])

try:
    urllib.request.urlretrieve(csv_file['contentUrl'], data_file_path)
    print(f"File Saved: {csv_file['name']}")
except:
    print("Save failed.")

```

```
File Saved: NGEEArctic_Seward_CN_2014.csv
```

8. Convert csv to dataframe

```
df = pandas.read_csv(content)
```

```
# Convert Date string to Date type
df['Date'] = pandas.to_datetime(df['Date'], format='%Y%m%d')
df
```

	Date	Road	Site	Species	sample ID	C (%)	N (%)	C:N	C (mg gDW-1)	N (mg gDW-1)	Notes	Addional site notes
0	2014-08-22	Kougarok Road	Kougarok River Bridge	Betula nana	48	48.29	1.14	49.606	482.9	11.4	NaN	NaN
1	2014-08-22	Kougarok Road	Kougarok River Bridge	Petasites	49	41.62	1.30	37.390	444.7	24.3	NaN	NaN
2	2014-08-22	Kougarok Road	Cabins (DOT)	Balsam poplar	50	44.47	2.43	21.355	444.7	24.3	NaN	NaN
3	2014-08-22	Kougarok Road	Norby's Pond	Salix	51	48.98	2.13	26.876	489.8	21.3	NaN	NaN
4	2014-08-22	Kougarok Road	Shawn's Slope	Petasites	52	42.12	1.45	33.936	421.2	14.5	NaN	where group split
5	2014-08-22	Kougarok Road	Kougarok River Bridge	Betula nana	53	48.81	1.86	30.679	488.1	18.6	NaN	NaN
6	2014-08-22	Kougarok Road	Kougarok River Bridge	Eriophorum	54	43.36	1.54	32.870	433.6	15.4	NaN	NaN

9. Summarize the data

```
df.max(numeric_only=True)
```

```
df.groupby('Date').agg({'C:N': ['min', 'max', 'mean'],
                        'C (%)': ['min', 'max', 'mean'],
                        'N (%)': ['min', 'max', 'mean'] })

```

```
df.groupby('Site').agg({'C:N': ['min', 'max', 'mean'],
                        'C (%)': ['min', 'max', 'mean'],
                        'N (%)': ['min', 'max', 'mean'] })
```

```
df.groupby('Species').agg({'C:N': ['min', 'max', 'mean'],
                        'C (%)': ['min', 'max', 'mean'],
                        'N (%)': ['min', 'max', 'mean'] })
```

Any Questions?

Photo by Towfiqu barbhuiya on Unsplash

Troubleshooting: Public vs Private Searches

I logged in to ESS-DIVE and used my authentication token to run my code, **but I don't see the dataset I'm looking for**. Why is that?

- The dataset you're looking for might be private
  - Check that your search query URL states `` `isPublic=false` ``
- Access to private datasets is **limited** to registered data submitters
- A registered data submitter can see private datasets that:
  - **They created**
  - Someone else has **shared with them**

Summary

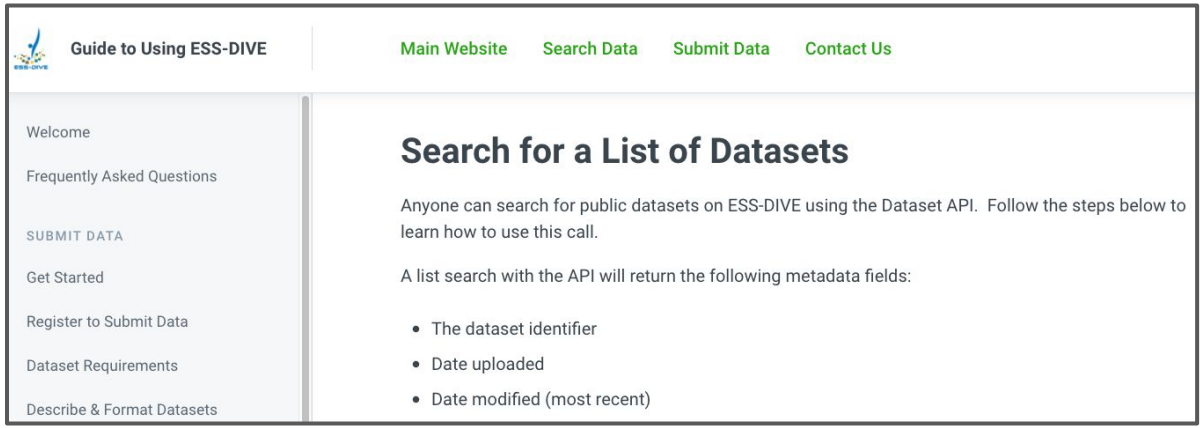
Summary

The Dataset API enables bulk dataset search and download

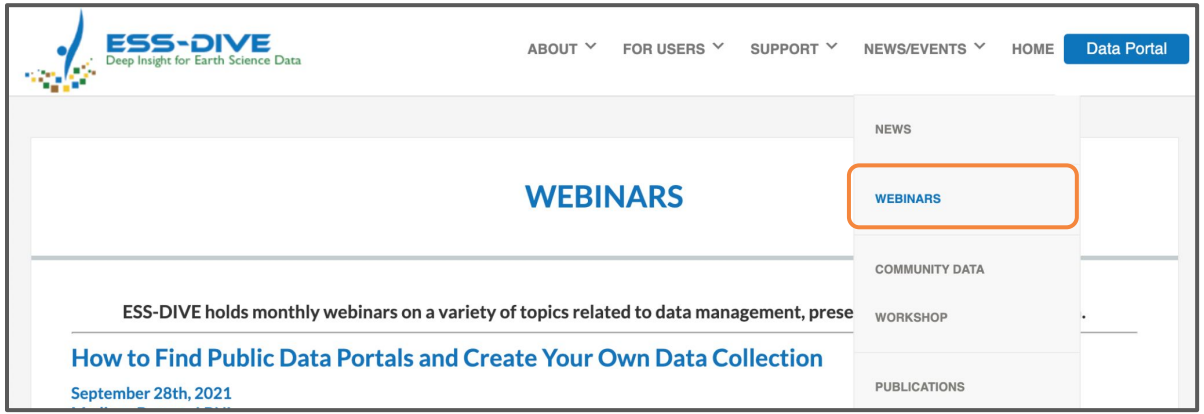
- The API has many other capabilities, including submitting and editing datasets
- Searches with the API are beneficial to projects and anyone working with a lot of datasets at once
- Create dataset reports and data visualizations programmatically with the API
- Start using the API right away by downloading our Jupyter Notebook demo

Questions? *Email ess-dive-support@lbl.gov*

You can find all this material on ESS-DIVE's…



Search for Data with API How-to Guide
docs.ess-dive.lbl.gov/



Webinar page
where this slide deck is available for download
ess-dive.lbl.gov/webinars/

API Resources

How-to Guides: docs.ess-dive.lbl.gov/programmatic-tools/ess-dive-dataset-api

Technical API Docs: api.ess-dive.lbl.gov/#/Data%20Package/listPackages

Code Examples: github.com/ess-dive/essdive-package-service-examples

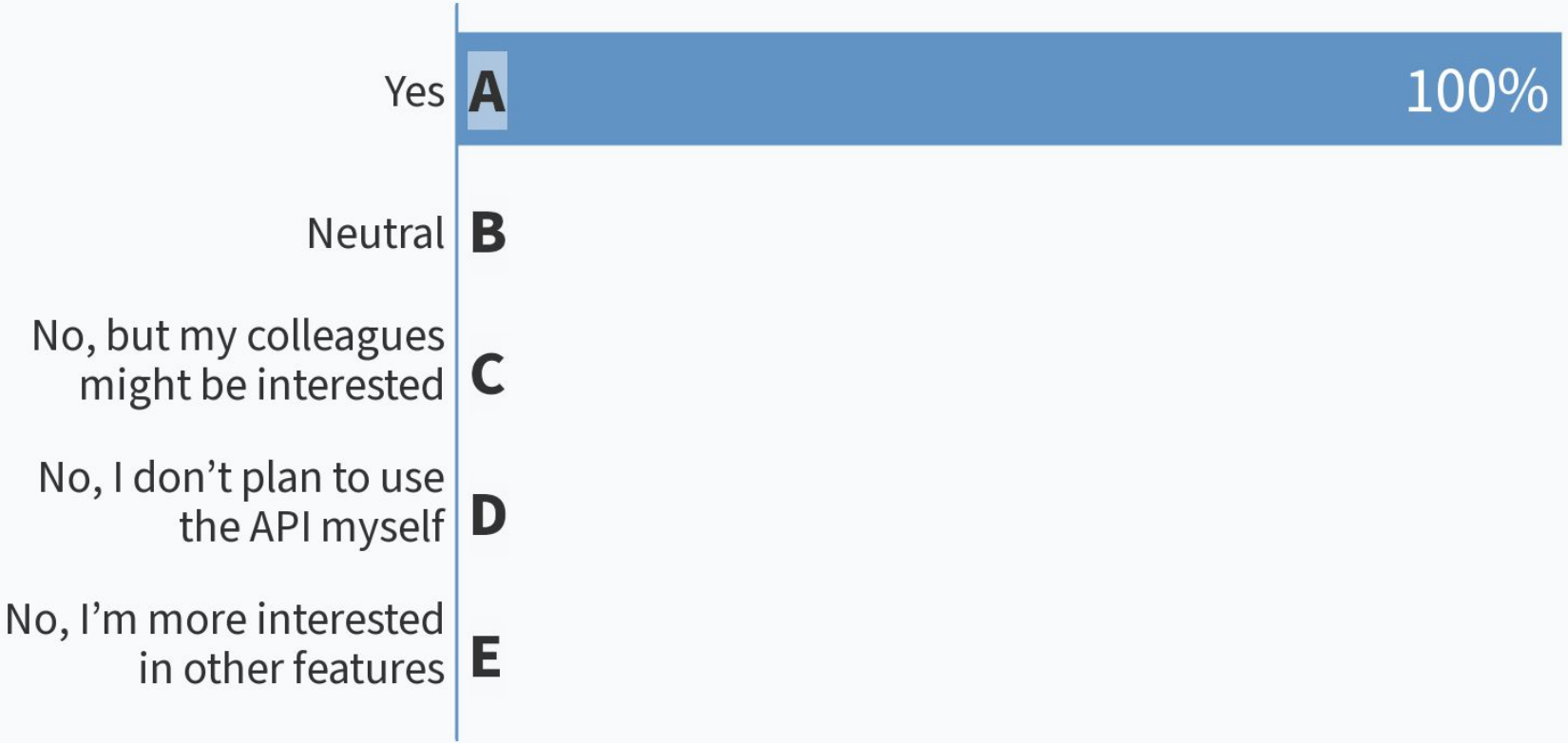
Jupyter Notebook Demo: github.com/ess-dive/essdive-package-service-examples

Feedback Tell us what you need!

Would you use the Dataset API to search for datasets? Or would you use other Dataset API features?

“ Annual reports. Comparing metadata between OSTI and ESS-Dive records ”

Would you like to see more webinars on the API?



Thank You!

@ESS-DIVE

Join ESS-DIVE's Community Mailing List!

http://bit.ly/essdiveMailingList

Contact us at ess-dive-support@lbl.gov

API 101: How to submit datasets

What can the API do?

- Create dataset metadata
- Upload data files
- Update datasets
- Search and download metadata or data
- Share datasets

Who benefits from using the API?

- Data users who want to examine a large number of datasets
- Projects creating a large number of datasets
- Projects with consistent metadata standards
- Teams who want to duplicate dataset metadata

API Dataset Submission Process

Review tutorial documentation for Dataset API

Get authentication token from ESS-DIVE

Create package metadata in JSON-LD

Success! View package landing page on ESS-DIVE

Errors? Fix any errors found by validation using documentation

Create new datasets

- Populate required metadata fields in JSON-LD schema
- The API will **validate and submit JSON-LD**
- If the JSON-LD is invalid, details about the errors will be given
- The API can submit data files up to **500 GB** in size


```
{
    "@context": "http://schema.org/",
    "@type": "Dataset",
    "@id": "http://dx.doi.org/10.3334/CDIAC/spruce.001",
    "name": "Correct Package with a one liner description",
    "description": ["This is a very long paragraph This is a very
    "creator": [
        {
            "@id": "http://orcid.org/0000-0001-7293-3561",
            "givenName": "Paul J",
            "familyName": "Hanson",
            "affiliation": "Oak Ridge National Laboratory",
            "email": "hansonpj@ornl.gov"
        },
        {
            "givenName": "Jeffrey",
            "familyName": "Riggs",
            "affiliation": "Oak Ridge National Laboratory"
        },
        {
            "givenName": "C",
            "familyName": "Nettles",
            "affiliation": "Oak Ridge National Laboratory"
```

Update existing dataset

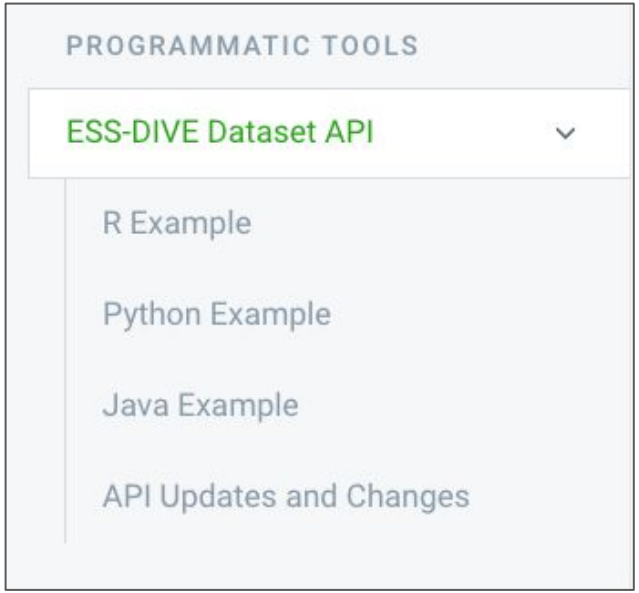
- **Metadata Updates**
  - Submit only the fields you want to update using JSON-LD
- **Data File Updates**
  - Upload additional files to an existing dataset

Dataset API How-To Guides

To get started with the Dataset API, we recommend using ESS-DIVE's **Dataset API How-To Guide**.

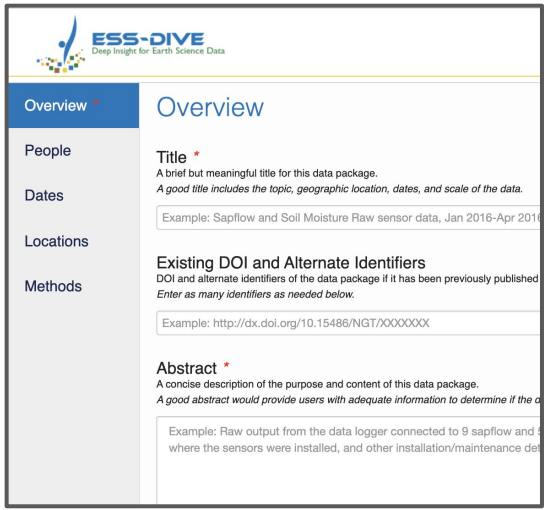
This Guide includes:

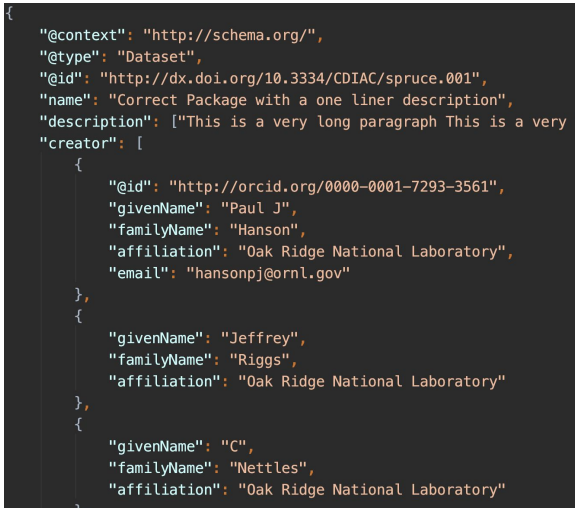
- All **prerequisites** needed to use the API
- **Code examples** of how to submit, update, and search data packages
- Complete examples in three **different coding languages**:
  - R
  - Python
  - Java



Accessible Coding

- API Guides give complete walkthrough of API submission
- JSON-LD metadata fields are equivalent to the fields in online submission form
- Quick learning curve for users of varying experience





Metadata Crosswalk example

ESS-DIVE Field	JSON-LD
Title	name
Alternative Identifiers	alternateName
Abstract	description
Keywords	keywords
Data Variables	variablesMeasured
Publication Date	datePublished

bit.ly/ess-dive-crosswalk