Options for producing OpenMC h5 cross sections

IAEA Consultancy Meeting on Model Code Output

&

Application Nuclear Data Form Structure

Jonathan Shimwell

Obtaining h5 cross section files

- Download compressed (tar) libraries from OpenMC.org
- Download individual isotopes using the openmc-data-downloader Python package https://github.com/openmc-data-storage/openmc_data_downloader
- Process h5 files using the data repository https://github.com/openmc-dev/data
  - Process ACE files using convert*.py scripts
  - Process ENDF files using generate*.py scripts

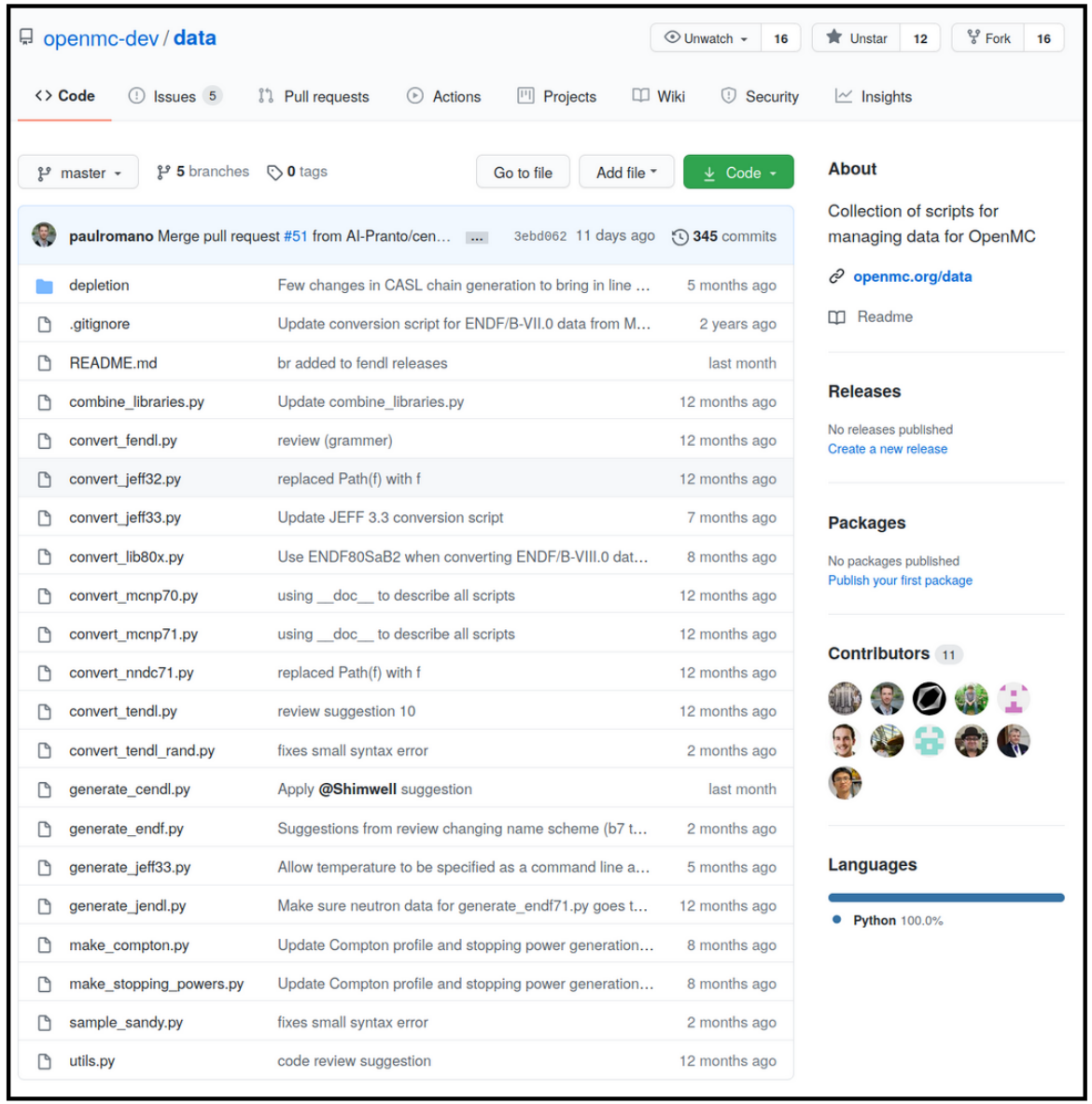
Obtaining h5 cross section files

There are several ways of obtaining the cross section data in the h5 file format required for neutronics simulations.

Library	Release	Processed by	Download from openmc.org	Download from openmc-data-downloader	Download ACE files and convert HDF5	Download ENDF files and generate HDF5	Convert local ACE files
CENDL	3.1 3.2					generate_cendl.py	
ENDF/B	VII.0	LANL	✓				convert_mcnp70.py
ENDF/B	VII.1	LANL	✓				convert_mcnp71.py
ENDF/B	VII.1	NNDC	✓	✓	convert_nndc71.py	generate_endf.py	
ENDF/B	VIII.0	LANL	✓				convert_lib80x.py
ENDF/B	VIII.0	NNDC	✓				
FENDL	2.1, 3.0, 3.1a, 3.1d			✓ **(3.1d)**	convert_fendl.py		
JENDL	4.0					generate_jendl.py	
JEFF	3.2		✓		convert_jeff32.py		
JEFF	3.3		✓		convert_jeff33.py		
TENDL	2015, 2017, 2019			✓ **(2019)**	convert_tendl.py		

OpenMC data repository

- Python scripts
- Permissive open source license (MIT)
- 11 contributors
- Storage and sharing opportunity to avoid reproducing the same scripts
- Useful for developers
- Generate scripts require NJOY install
- PyPi package with pip install on the horizon



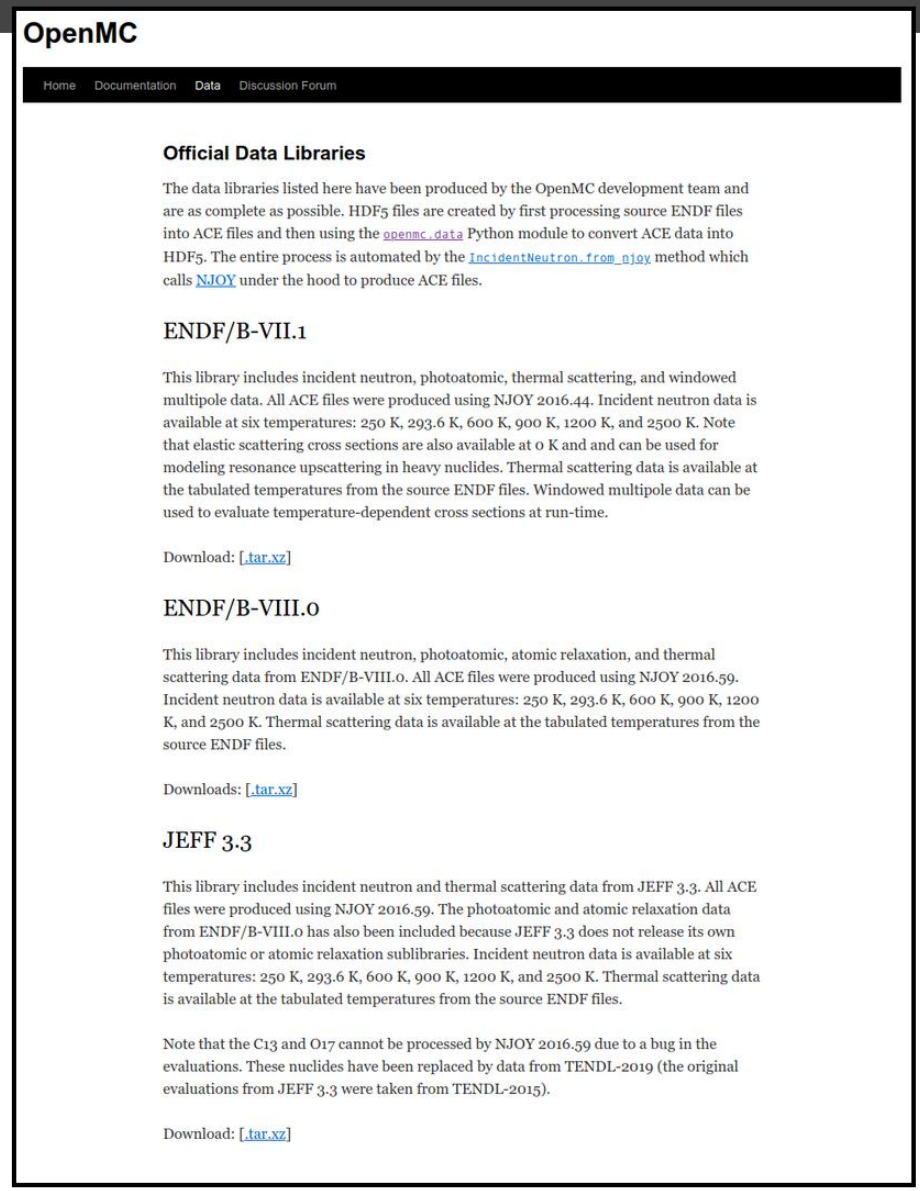
Obtaining h5 cross sections - data repo

- Example usage of one of the Python scripts
- This particular script downloads 429MB of compressed ACE and ENDF files
- Extracts the files
- Processes the files into h5 file for use in OpenMC

```
git clone https://github.com/openmc-dev/data.git

cd data

python convert_fendl.py --release 3.1d
>> downloading, extracting and processing ...

ls
>> fendl-3.1d-ace/
>> fendl-3.1d-endf/
>> fendl-3.1d-download/
>> fendl-3.1d-hdf5/

ls fendl-3.1d-hdf5/
>> cross_sections.xml
>> neutron/
>> photon/

ls fendl-3.1d-hdf5/neutron
>> Ag107.h5
>> Ag109.h5
>> Al27.h5
>> Ar36.h5
>> ...

export OPENMC_CROSS_SECTIONS=fendl-3.1d-hdf5/cross
```

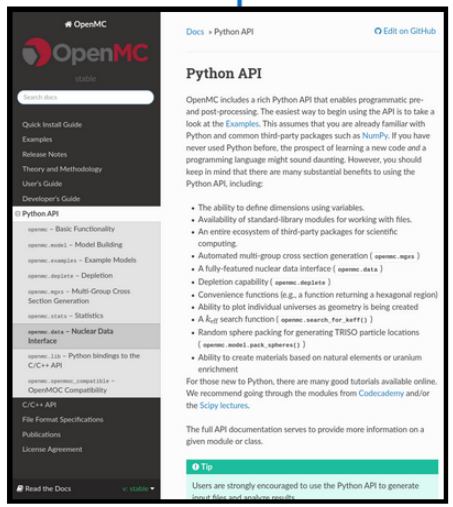
OpenMC.org

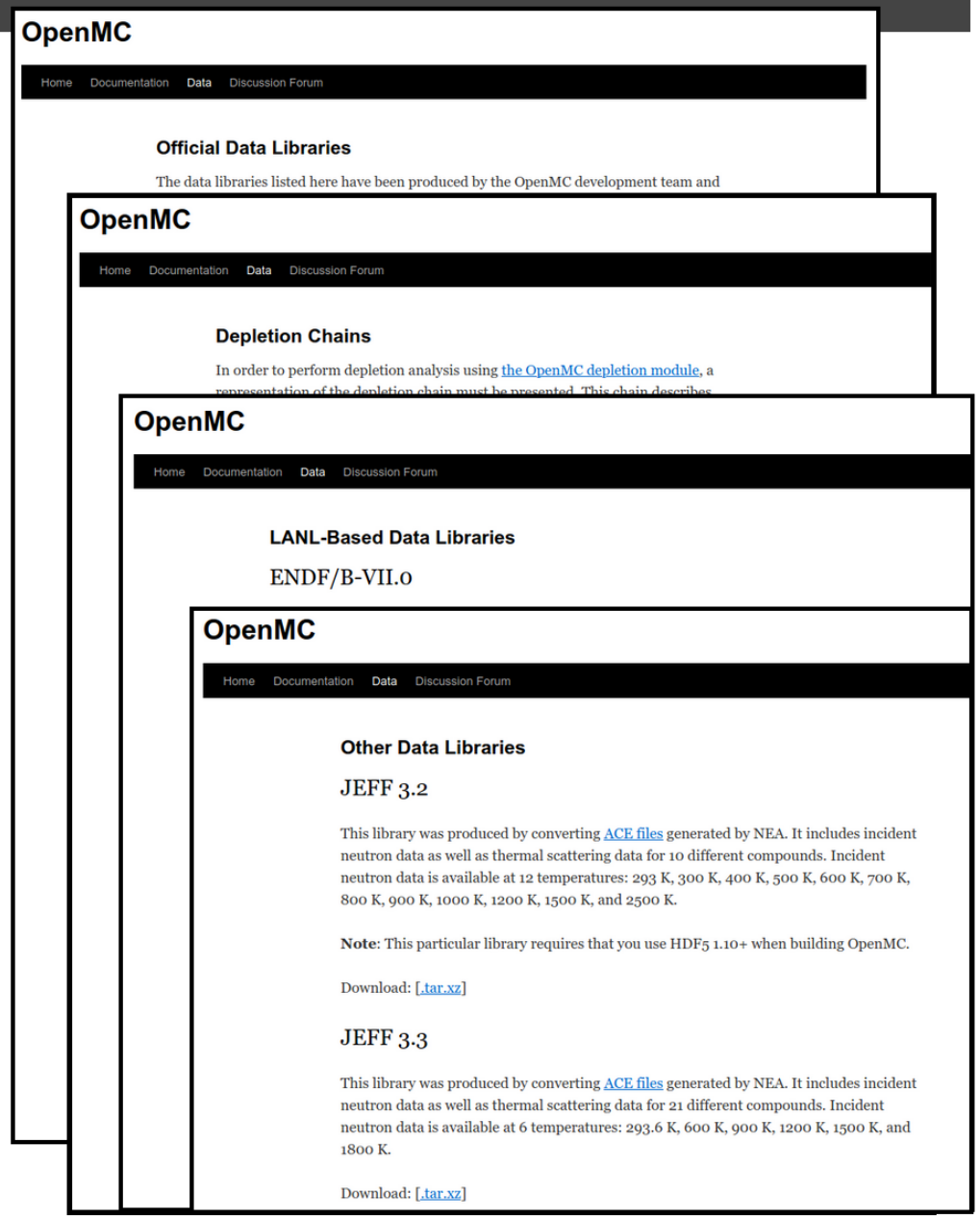
- https://openmc.org includes a data section with downloads
  - Official Libraries
  - LANL-Based Data Libraries
  - Other Libraries
  - Depletion chains



OpenMC.org

- https://openmc.org includes a data section with downloads
  - Official Libraries
  - LANL-Based Data Libraries
  - Other Libraries
  - Depletion chains
- OpenMC documentation pages includes information on the nuclear data interface https://docs.openmc.org



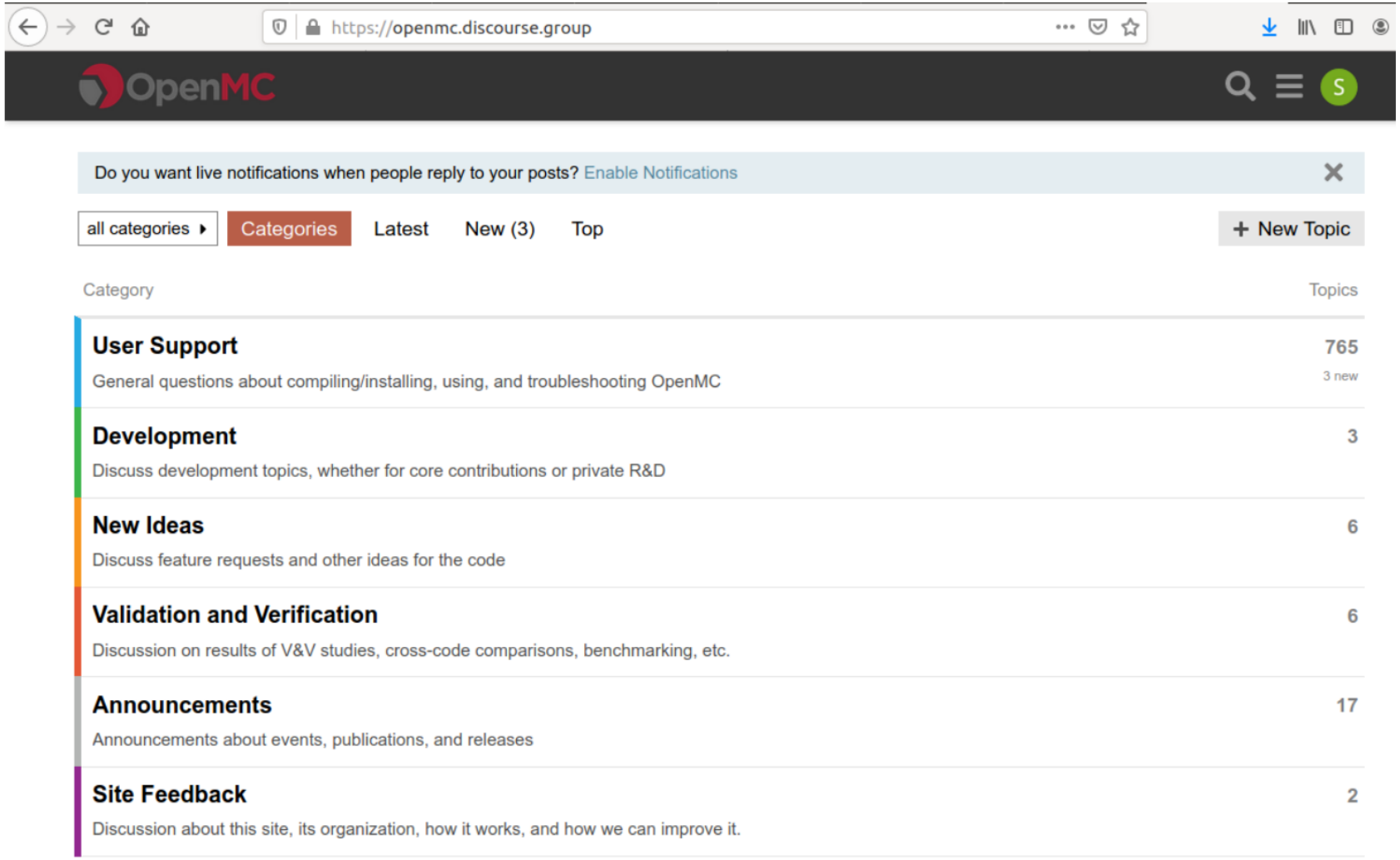


Obtaining h5 cross sections - OpenMC.org

- Downloads can be scripted using wget
- The h5 files can be used in simulations once the OPENMC_CROSS_SECTION environmental variable has been set

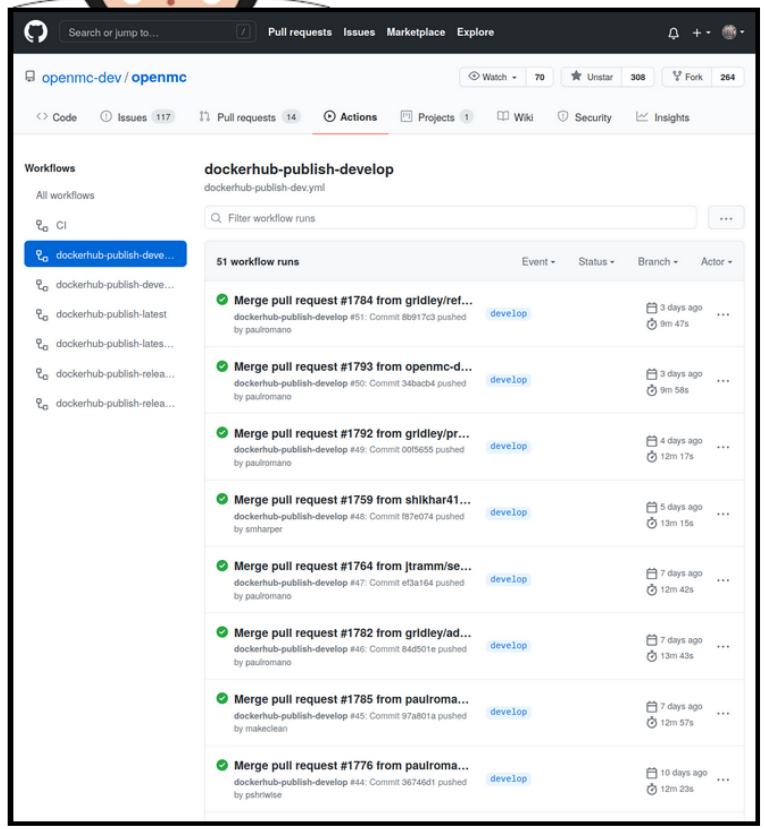
```
wget -O nndc-b7.1.tar.xz https://anl.box.com/shared/static/9igk353zpy8fn9ttvtrqgzvw1vtejoz6.xz
>> dowloading compressed file ...

mkdir nndc-b7.1-hdf5

tar -xf nndc-b7.1.tar.xz -C nndc-b7.1-hdf5
>> extracting compressed file ...

ll nndc-b7.1-hdf5
>>cross_sections.xml
>>neutron/
>>photon/

ll nndc-b7.1-hdf5/neutron
>>Ac225.h5
>>Ac226.h5
>>Ac227.h5
>>Ag107.h5
>>Ag109.h5
>>...

export OPENMC_CROSS_SECTIONS=nndc-b7.1-hdf5/cross_sections.xml
```

Obtaining h5 cross sections - OpenMC.org

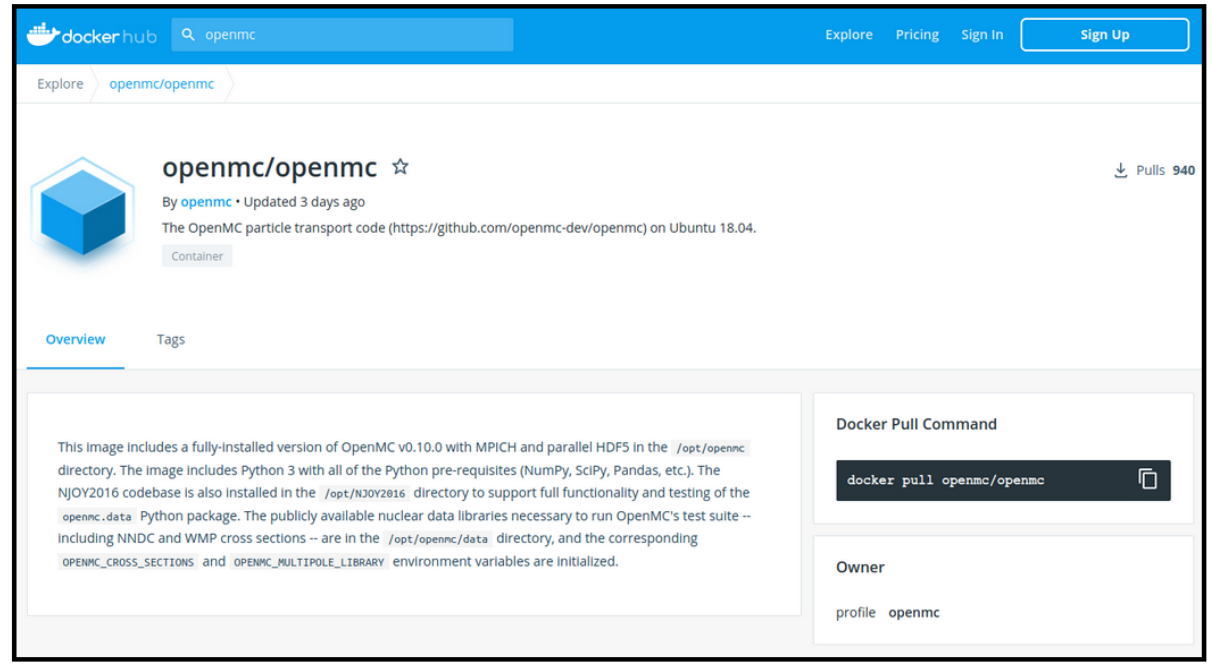


OpenMC Docker image production

- OpenMC automatically produces a new docker image with every code merge
- Github Actions is used to build the docker image and upload it to Dockerhub



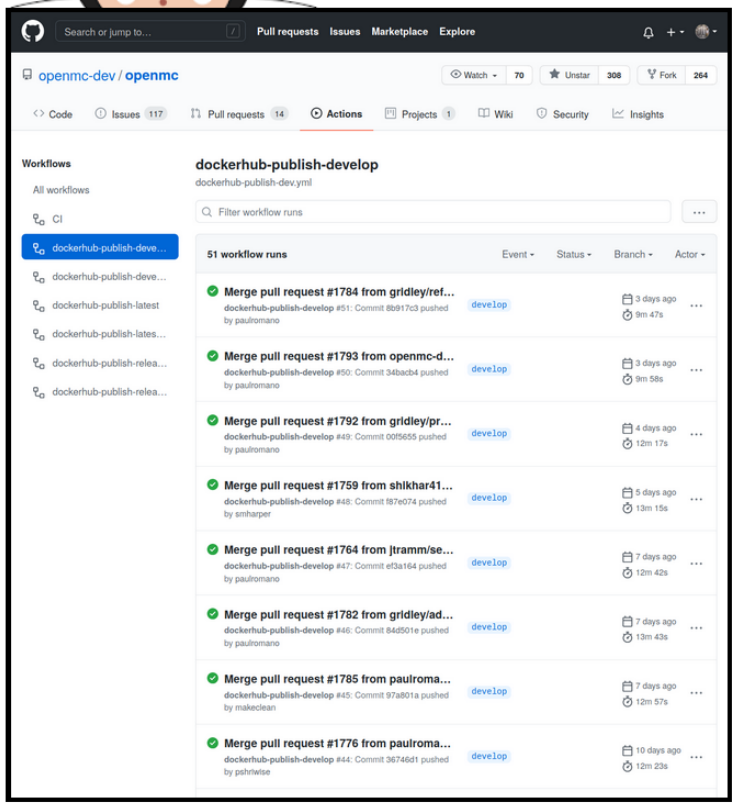
https://github.com/openmc-dev/openmc/actions



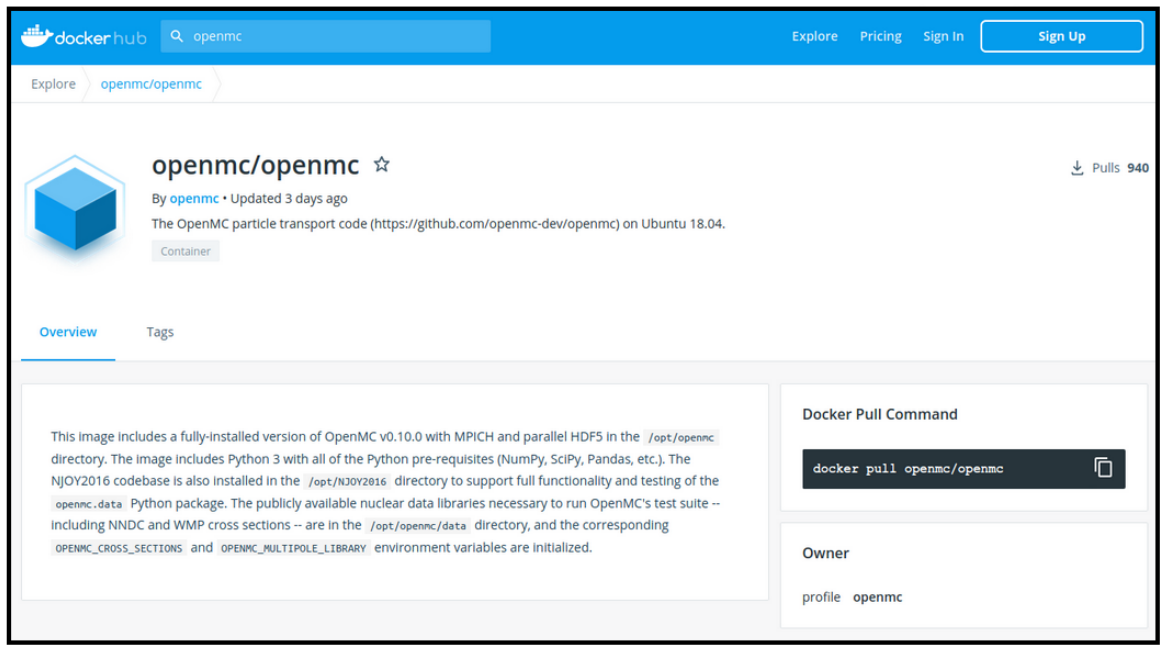
https://hub.docker.com/r/openmc/openmc

OpenMC Docker image production

- OpenMC automatically produces a new docker image with every code merge
- Github Actions is used to build the docker image and upload it to Dockerhub



https://github.com/openmc-dev/openmc/actions



https://hub.docker.com/r/openmc/openmc

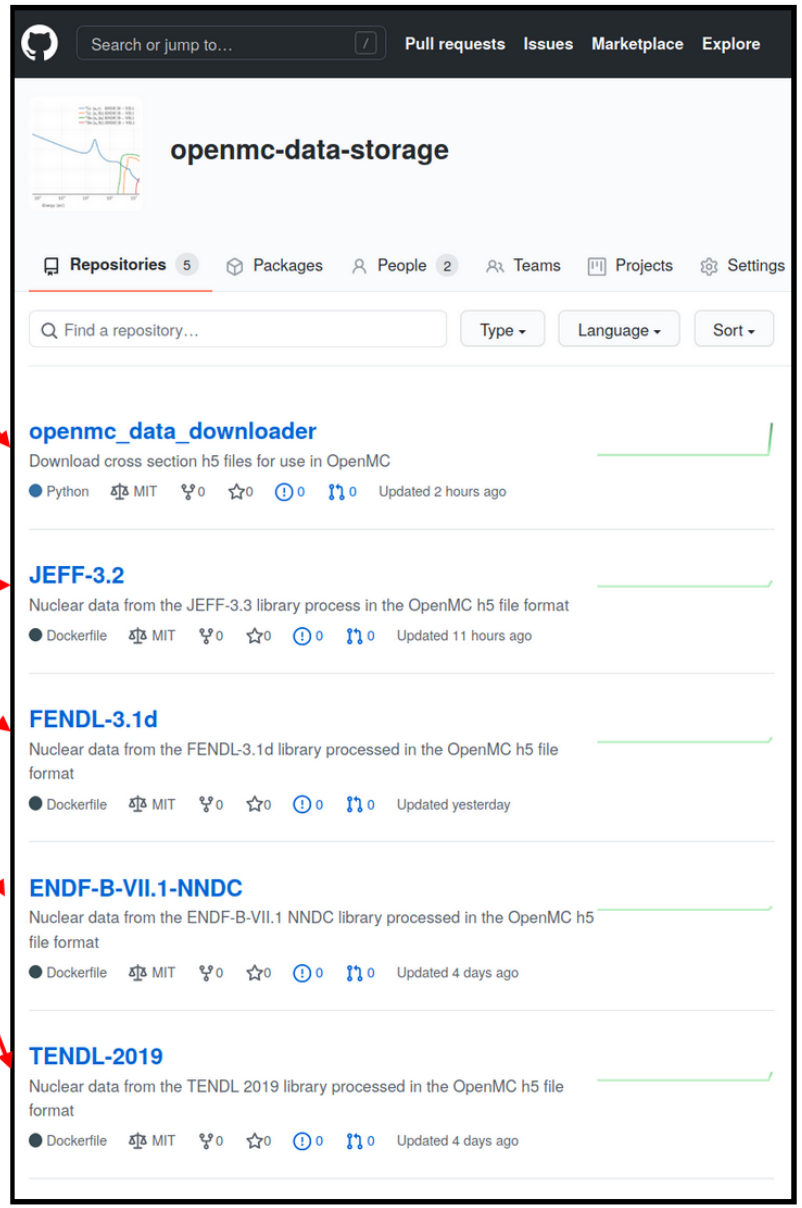
Data storage repo - processing

Contains repository for the openmc-data-downloader Python package.

Contains separate repositories for each library release

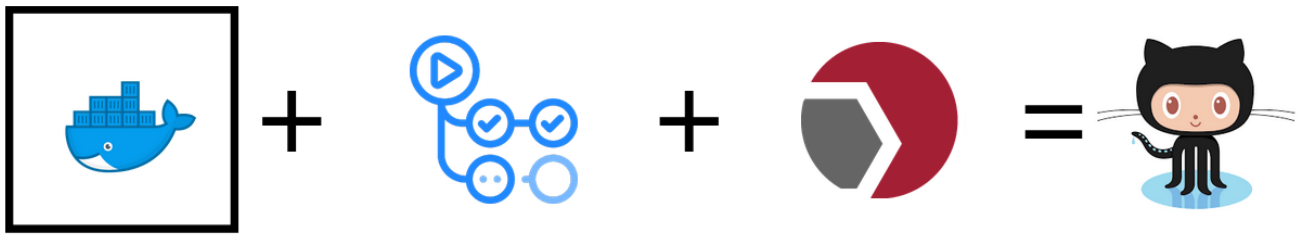
Repositories can be templated

Open source, permissively licensed (MIT)
https://github.com/openmc-data-storage



Data storage repo - enviroment

4 Lines



Use of the openmc/openmc-dev image
this contains NJOY as well

Addition of the openmc-data
repo to the base image

Copying test suite

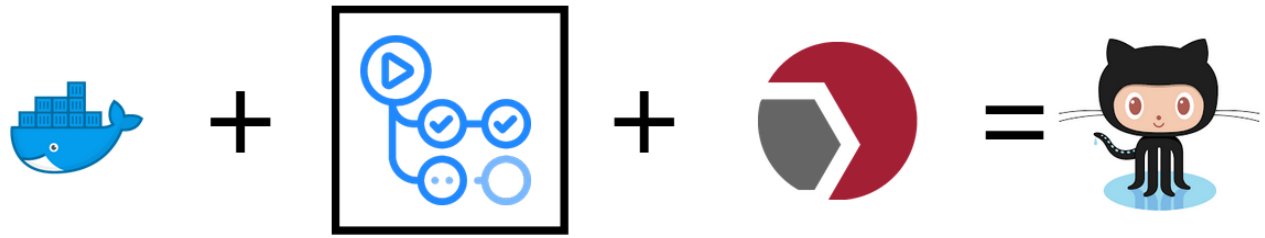
See environmental variable

```
FROM openmc/openmc:develop

RUN git clone https://github.com/openmc-dev/data.git

COPY tests tests/

ENV OPENMC_CROSS_SECTIONS=/share/cross_sections.xml
```

Data storage repo - processing

4 Lines

40 Lines



Github actions (CI) build and run the new docker image

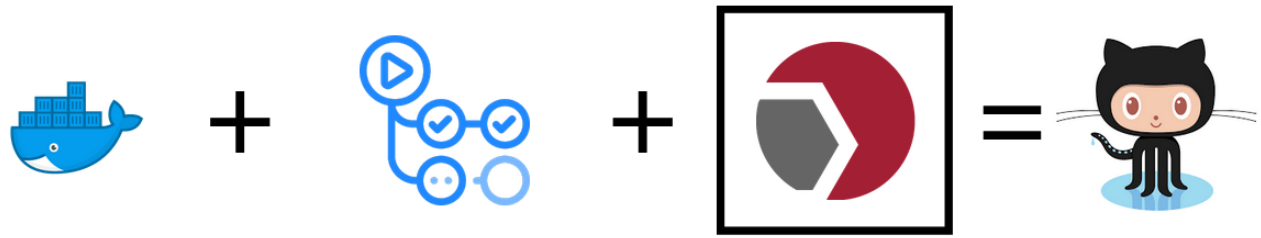
The new docker image is then run externally with a shared volume mount

Github action (CI) then commits the new h5 files from the share folder to the repository

```
    - name: Create custom readme
      run: |
          python readme_writer.py
          cat README.md
    - name: Build and test with Docker
      run: |
          docker build --tag nuc_data_maker .
          rm -rf h5_files
          docker run -v /home/runner/work/${{ github.event.repos
    - name: Commit files
      run: |
        git config --local user.email "action@github.com"
        git config --local user.name "GitHub Action"
        git add h5_files/*.h5
        git add h5_files/cross_sections.xml
        git add README.md
        git commit -m "Added readme with repo name (bot commit)"
    - name: Push changes
      uses: ad-m/github-push-action@master
      with:
        github_token: ${{ secrets.GITHUB_TOKEN }}
        branch: ${{ github.ref }}
```

Data storage repo - testing

4 Lines + 40 Lines + 39 Lines =



```
import xml.etree.ElementTree as ET

import openmc

os.environ["OPENMC_CROSS_SECTIONS"] = '/share

class TestH5FilesInSimulations(unittest.TestC

    def test_simulation_runs_with_isotopes(se
        tree = ET.parse('/share/h5_files/cross
        root = tree.getroot()

        for elem in root:
            if elem.attrib['type'] == 'neutro
                isotope_name = elem.attrib['m

                sett = openmc.Settings()
                sett.batches = 100
                sett.inactive = 0
                sett.particles = 500
                sett.run_mode = 'fixed source'
```

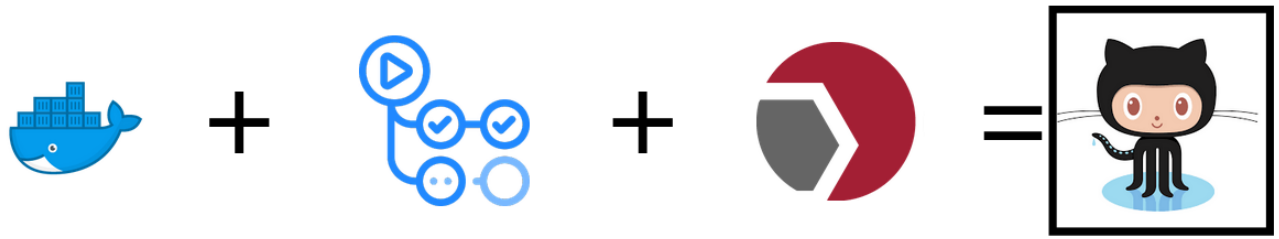
Each h5 cross section file is then used in an OpenMC simulation

Testing the h5 file produced has converted successfully and does not cause errors during usage

```
build
succeeded 1 hour ago in 8m 14s
Search logs

Build and test with Docker 6m 16s

                                %%%%%%%%%%%%%%%%%%%%%%
                             %%%%%%%%%%%%%%%%%%%%%%%%%%
                          %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                       %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                                      %%%%%%%%%%%%%%%%%%%%%%%
                                   %%%%%%%%%%%%%%%%%%%%%%%%%%
                  ###############     %%%%%%%%%%%%%%%%%%%%%%%
                ################      %%%%%%%%%%%%%%%%%%%%%%%
               ##################     %%%%%%%%%%%%%%%%%%%%%%%
               ###################    %%%%%%%%%%%%%%%%%%%%%%%
               ####################   %%%%%%%%%%%%%%%%%%%%%%%
               ####################   %%%%%%%%%%%%%%%%%%%%%%%
               ###################    %%%%%%%%%%%%%%%%%%%%%%%
                ##################    %%%%%%%%%%%%%%%%%%%%%%
                 ################     %%%%%%%%%%%%%%%%%%%%%
                 ################     %%%%%%%%%%%%%%%%%%%%%
                  ###############     %%%%%%%%%%%%%%%%%%%%
                   ##############     %%%%%%%%%%%%%%%%%%
                    ############      %%%%%%%%%%%%%%%%
                      ########        %%%%%%%%%%%%%%
                                      %%%%%%%%%%

                    | The OpenMC Monte Carlo Code
          Copyright | 2011-2021 MIT and OpenMC contributors
            License | https://docs.openmc.org/en/latest/license.html
            Version | 0.12.1-dev
          Git SHA1 | c4e0881d07e79042d5a69f9cae6e373963cfcdf1
         Date/Time | 2021-03-17 10:16:02
       MPI Processes | 1
      OpenMP Threads | 2

Reading settings XML file...
Reading cross sections XML file...
Reading materials XML file...
Reading geometry XML file...
Reading He4 from /share/h5_files/neutron/He4.h5
Minimum neutron data temperature: 300.0 K
Maximum neutron data temperature: 300.0 K
Preparing distributed cell instances...
Writing summary.h5 file...
Maximum neutron transport energy: 60000000.0 eV for He4

===============>     FIXED SOURCE TRANSPORT SIMULATION     <===============

Simulating batch 1
Simulating batch 2
```

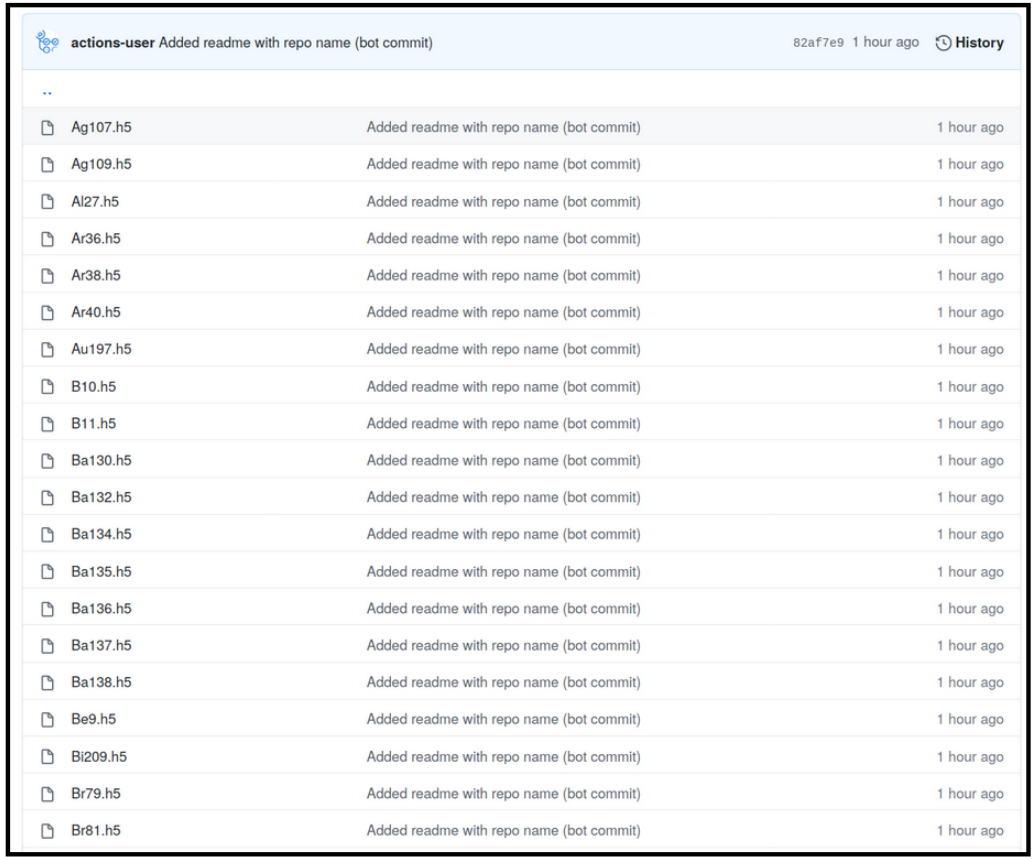
Data storage repo - storage

4 Lines 40 Lines 39 Lines



Gigabytes of h5 files each with their own URL endpoint

https://github.com/openmc-data-storage/FENDL-3.1d/blob/main/h5_files/neutron/Ag107.h5?raw=true



actions-user Added readme with repo name (bot commit) 82af7e9 1 hour ago **History**

..

Ag107.h5	Added readme with repo name (bot commit)	1 hour ago
Ag109.h5	Added readme with repo name (bot commit)	1 hour ago
Al27.h5	Added readme with repo name (bot commit)	1 hour ago
Ar36.h5	Added readme with repo name (bot commit)	1 hour ago
Ar38.h5	Added readme with repo name (bot commit)	1 hour ago
Ar40.h5	Added readme with repo name (bot commit)	1 hour ago
Au197.h5	Added readme with repo name (bot commit)	1 hour ago
B10.h5	Added readme with repo name (bot commit)	1 hour ago
B11.h5	Added readme with repo name (bot commit)	1 hour ago
Ba130.h5	Added readme with repo name (bot commit)	1 hour ago
Ba132.h5	Added readme with repo name (bot commit)	1 hour ago
Ba134.h5	Added readme with repo name (bot commit)	1 hour ago
Ba135.h5	Added readme with repo name (bot commit)	1 hour ago
Ba136.h5	Added readme with repo name (bot commit)	1 hour ago
Ba137.h5	Added readme with repo name (bot commit)	1 hour ago
Ba138.h5	Added readme with repo name (bot commit)	1 hour ago
Be9.h5	Added readme with repo name (bot commit)	1 hour ago
Bi209.h5	Added readme with repo name (bot commit)	1 hour ago
Br79.h5	Added readme with repo name (bot commit)	1 hour ago
Br81.h5	Added readme with repo name (bot commit)	1 hour ago

Data storage repo - openmc-data-downloader

Easy to install Python package

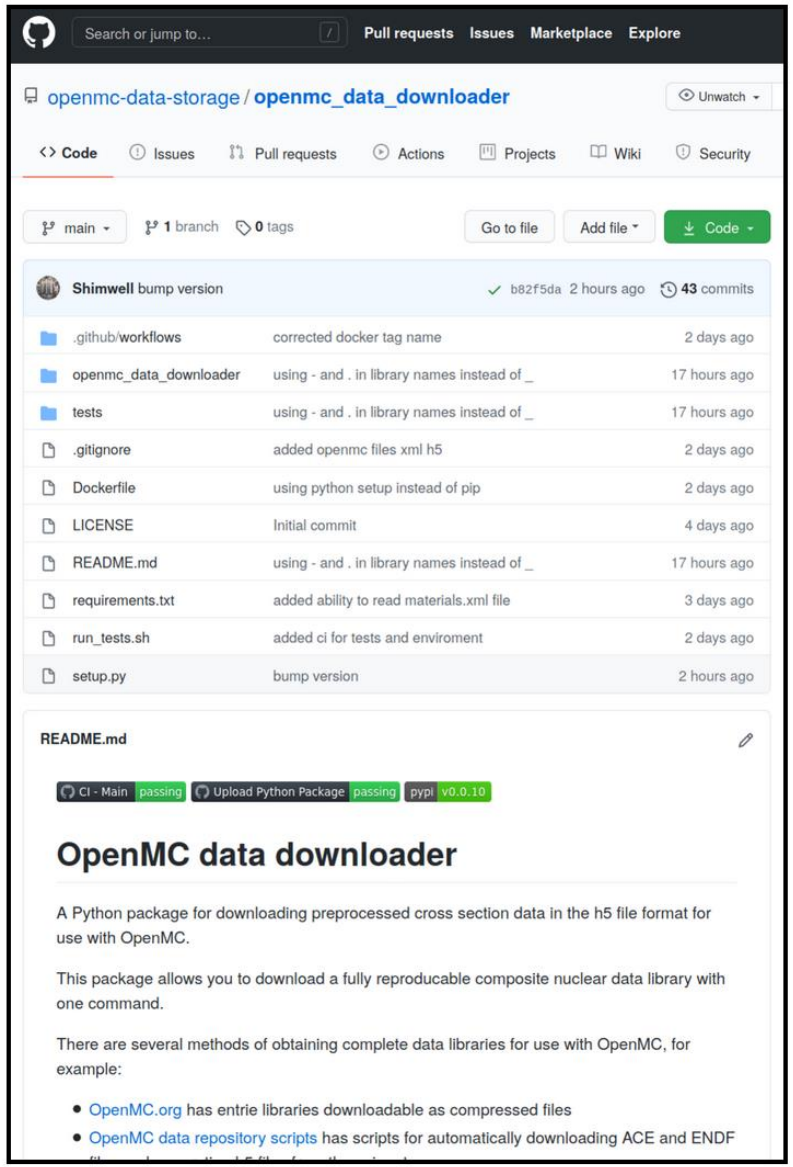
```
pip install openmc_data_downloader
```

Knowledge of all the URLs of all the h5 files available.

```
{
  'element': 'Zn',
  'isotope': 'Zn66',
  'library': 'FENDL-3.1d',
  'local_file': 'FENDL-3.1d_Zn66.h5',
  'remote_file': 'Zn66.h5',
  'url': 'https://github.com/openmc-data-storage/FEN
 },
 ...
```

Internal data based that gets filtered by library / isotopes etc

Translation from user instructions to a list of h5 files to download



https://github.com/openmc-data-storage/openmc_data_downloader

Data storage repo - openmc-data-downloader

Command line usage `openmc_data_downloader --help`

```
(base) jshim@jshim-desktop:~$ openmc_data_downloader --help
usage: openmc_data_downloader [-h] -l [{ENDFB-7.1-NNDC,TENDL-2019,FENDL-3.1d} [{ENDFB-7.1-NNDC,TENDL-2019,FENDL-3.1d} ...]] [-i [ISOTOPES [ISOTOPES ...]]]
                              [-e [ELEMENTS [ELEMENTS ...]]] [-m [MATERIALS_XML [MATERIALS_XML ...]]] [-d DESTINATION]

optional arguments:
  -h, --help            show this help message and exit
  -l [{ENDFB-7.1-NNDC,TENDL-2019,FENDL-3.1d} [{ENDFB-7.1-NNDC,TENDL-2019,FENDL-3.1d} ...]], --libraries [{ENDFB-7.1-NNDC,TENDL-2019,FENDL-3.1d} [{ENDFB-7.1-NN
DC,TENDL-2019,FENDL-3.1d} ...]]
                        The nuclear data libraries to search through when searching for cross sections. Multiple libaries are acceptable and will be
                        preferentially utilised in the order provided
  -i [ISOTOPES [ISOTOPES ...]], --isotopes [ISOTOPES [ISOTOPES ...]]
                        The isotopes to download
  -e [ELEMENTS [ELEMENTS ...]], --elements [ELEMENTS [ELEMENTS ...]]
                        The elements to download
  -m [MATERIALS_XML [MATERIALS_XML ...]], --materials_xml [MATERIALS_XML [MATERIALS_XML ...]]
                        The filename of the materials.xml file to provide cross sections for
  -d DESTINATION, --destination DESTINATION
                        Directory to create new library in
```

Data storage repo - openmc-data-downloader

```
openmc_data_downloader -l FENDL-3.1d -i Li6
```

Downloads one isotope

```
openmc_data_downloader -l TENDL-2019 -i Li6 Li7
```

Downloads two isotopes

```
openmc_data_downloader -l TENDL-2019 -e Li
```

Downloads one element

```
openmc_data_downloader -l TENDL-2019 -e Li Si Na
```

Downloads two elements

```
openmc_data_downloader -l ENDFB-7.1-NNDC -i Be9 -d my_h5_files
```

Downloads to a directory

```
openmc_data_downloader -l TENDL-2019 -e Li Si Na -i Fe56 U235
```

Downloads a combination of elements and isotopes

```
openmc_data_downloader -l TENDL-2019 -m materials.xml
```

Downloads all isotopes in a materials.xml file

```
openmc_data_downloader -l FENDL-3.1d TENDL-2019 -i Li6 U235
```

Downloads isotopes with library preference

Just in time library generator

```python
import openmc
import openmc_data_downloader as odd

mat1 = openmc.Material(1, "breeder_mat")
mat1.add_element('Pb', 84.2, percent_type='ao')
mat1.add_element('Li', 15.8, percent_type='ao')
mat1.set_density('g/cm3', 11.3)
mats = openmc.Materials([mat1])

odd.just_in_time_library_generator(
    libraries='FENDL-3.1d',
    materials=mat1
)
```

Extra import

Material

Download h5 cross section files for the materials, accepts lists or individual items

Resulting download of h5 files and setting of OPENMC_CROSS_SECTIONS environmental varible

```
(base) jshim@jshim-desktop:~$ python iaea_example.py
Searching libraries with the following priority {'FENDL-3.1d': 1}
Isotopes found matching library requirements 180
Isotopes found matching isotope requirements 18
Isotopes found matching all requirements 6
Downloading FENDL-3.1d_Li6.h5...
Downloading FENDL-3.1d_Li7.h5...
Downloading FENDL-3.1d_Pb204.h5...
Downloading FENDL-3.1d_Pb206.h5...
Downloading FENDL-3.1d_Pb207.h5...
Downloading FENDL-3.1d_Pb208.h5...
/home/jshim/cross_sections.xml written
setting OPENMC_CROSS_SECTIONS /home/jshim/cross_sections.xml
```

Re-running the same Python script and h5 downloading can be skipped as they already exist

```
(base) jshim@jshim-desktop:~$ python iaea_example.py
Searching libraries with the following priority {'FENDL-3.1d': 1}
Isotopes found matching library requirements 180
Isotopes found matching isotope requirements 18
Isotopes found matching all requirements 6
Skipping FENDL-3.1d_Li6.h5, already downloaded
Skipping FENDL-3.1d_Li7.h5, already downloaded
Skipping FENDL-3.1d_Pb204.h5, already downloaded
Skipping FENDL-3.1d_Pb206.h5, already downloaded
Skipping FENDL-3.1d_Pb207.h5, already downloaded
Skipping FENDL-3.1d_Pb208.h5, already downloaded
/home/jshim/cross_sections.xml written
setting OPENMC_CROSS_SECTIONS /home/jshim/cross_sections.xml
```

Just in time library generator - in simulation

```python
import openmc
import openmc_data_downloader as odd

mat1 = openmc.Material(1, "breeder_mat")
mat1.add_element('Pb', 84.2, percent_type='ao')
mat1.add_element('Li', 15.8, percent_type='ao')
mat1.set_density('g/cm3', 11.3)
mats = openmc.Materials([mat1])

outer_surface = openmc.Sphere(r=500, boundary_type='vac
cell1 = openmc.Cell(region=-outer_surface)
cell1.fill = mat1
universe = openmc.Universe(cells=[cell1])
geom = openmc.Geometry(universe)

sett = openmc.Settings()
sett.batches = 100
sett.particles = 500
sett.run_mode = 'fixed source'

source = openmc.Source()
source.space = openmc.stats.Point((0, 0, 0))
source.angle = openmc.stats.Isotropic()
source.energy = openmc.stats.Discrete([14e6], [1])
sett.source = source

odd.just_in_time_library_generator(
    libraries='FENDL-3.1d',
    materials=mat1
)

model = openmc.model.Model(geom, mats, sett)
model.run()
```

Extra import

Materials

CSG

Settings

Point source

Download h5 cross section files for the materials

Run the simulation

```
(base) jshim@jshim-desktop:~$ python iaea_example.py
Searching libraries with the following priority {'FENDL-3.1d': 1}
Isotopes found matching library requirements 180
Isotopes found matching isotope requirements 18
Isotopes found matching all requirements 6
Downloading FENDL-3.1d_Li6.h5...
Downloading FENDL-3.1d_Li7.h5...
Downloading FENDL-3.1d_Pb204.h5...
Downloading FENDL-3.1d_Pb206.h5...
Downloading FENDL-3.1d_Pb207.h5...
Downloading FENDL-3.1d_Pb208.h5...
/home/jshim/cross_sections.xml written
setting OPENMC_CROSS_SECTIONS /home/jshim/cross_sections.xml
                                %%%%%%%%%%%%%%%
                           %%%%%%%%%%%%%%%%%%%%%%%%
                        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                   %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                                    %%%%%%%%%%%%%%%%%%%%%%%
                                    %%%%%%%%%%%%%%%%%%%%%%%
               ###############      %%%%%%%%%%%%%%%%%%%%%%%
              ################      %%%%%%%%%%%%%%%%%%%%%%%
              ################      %%%%%%%%%%%%%%%%%%%%%%%
              ##################    %%%%%%%%%%%%%%%%%%%%%%%
              ###################   %%%%%%%%%%%%%%%%%%%%%%%
              ###################   %%%%%%%%%%%%%%%%%%%%%%%
              ###################   %%%%%%%%%%%%%%%%%%%%%%%
               ##################   %%%%%%%%%%%%%%%%%%%%%%
               ##################   %%%%%%%%%%%%%%%%%%%%%%
                ################    %%%%%%%%%%%%%%%%%%%%%
                 ###############    %%%%%%%%%%%%%%%%%%%%%
                  ##############    %%%%%%%%%%%%%%%%%%%%%
                   ############     %%%%%%%%%%%%%%%%%%%%
                     ########       %%%%%%%%%%%%%%%%%%
                                    %%%%%%%%%%%%%%%%%%

                     | The OpenMC Monte Carlo Code
           Copyright | 2011-2020 MIT and OpenMC contributors
             License | https://docs.openmc.org/en/latest/license.html
             Version | 0.12.1-dev
            Git SHA1 | f952fa9cbcdf00f585f42cc605daf6fe0e44b447
           Date/Time | 2021-03-15 13:52:06
      OpenMP Threads | 64

 Reading settings XML file...
 Reading cross sections XML file...
 Reading materials XML file...
 Reading geometry XML file...
 Reading Pb204 from /home/jshim/FENDL-3.1d_Pb204.h5
 Reading Pb206 from /home/jshim/FENDL-3.1d_Pb206.h5
 Reading Pb207 from /home/jshim/FENDL-3.1d_Pb207.h5
 Reading Pb208 from /home/jshim/FENDL-3.1d_Pb208.h5
 Reading Li6 from /home/jshim/FENDL-3.1d_Li6.h5
 Reading Li7 from /home/jshim/FENDL-3.1d_Li7.h5
 Minimum neutron data temperature: 300.0 K
 Maximum neutron data temperature: 300.0 K
 Reading tallies XML file...
 Preparing distributed cell instances...
 Writing summary.h5 file...
 Maximum neutron transport energy: 200000000.0 eV for Pb204

 ===============>     FIXED SOURCE TRANSPORT SIMULATION     <===============

 Simulating batch 1
 Simulating batch 2
```

Scale using repository templates

Less time spent making a repo for new nuclear data

~ 1 min human time

Architecture + customization = new nuclear data repo

Live demo

Building blocks

OpenMC Python API

IncidentNeutron, from_endf and from_ace

+URL endpoints for ACE and ENDF libraries

Openmc-dev data repository

convert*.py and generate*.py

+ Run commands and containerized enviroment

Automated processing and testing via CI

docker build, docker run and pytest

+URL end points for h5 files and libraries

Downloading of reproducible collections of isotopes

Summary

There are several methods of obtaining the h5 cross section files needed to run OpenMC simulations.

Each method has a specific use case.

- OpenMC.org - GUI, script, entire libraries, h5 data
- OpenMC-dev data repo - scripts for processing libraries
- openmc-data-downloader - reproducible mixed micro libraries

Future work

- Ability to specify particle type to allow photons
- Ability to specify temperature