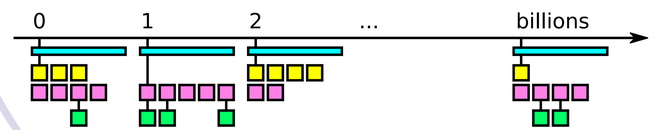
Awkward Arrays: Accelerating scientific data analysis on irregularly shaped data

Framework, #2103945 PI: Jim Pivarski

Awkward Array

An array library for **nested, variable-sized data**, including arbitrary-length lists, records, mixed types, and missing data, using **NumPy-like idioms**.



Arrays are **dynamically typed**, but operations on them are **compiled and fast**.

Coincides with NumPy when arrays are regular; generalizes when they're not.

What it is, what it's for

Example of an "awkward" array

```
array = ak.Array([
    [{"x": 1.1, "y": [1]}, {"x": 2.2, "y": [1, 2]}, {"x": 3.3, "y": [1, 2, 3]}],
    [],
    [{"x": 4.4, "y": [1, 2, 3, 4]}, {"x": 5.5, "y": [1, 2, 3, 4, 5]}]
])
```

Record structures with differently typed fields

Array of variable-length lists ("ragged" or "jagged" arrays)

Nested variable-length lists

Missing data

Heterogenous data (union/variant types)

NumPy-like expression

```
output = np.square(array["y", ..., 1:])
```

Result

```
output.to_list()
    [
        [[], [4], [4, 9]],
        [],
        [[4, 9, 16], [4, 9, 16, 25]]
    ]
```

Equivalent Python

```
output = []
for sublist in python_objects:
    tmp1 = []
    for record in sublist:
        tmp2 = []
        for number in record["y"][1:]:
            tmp2.append(np.square(number))
        tmp1.append(tmp2)
    output.append(tmp1)
```

(for a similar calculation 10 million times larger, single threaded)

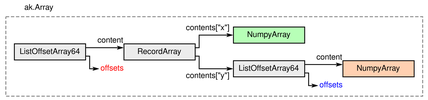
1.5 seconds
2.1 GB of memory

140 seconds
22 GB of memory

How it works

An entire array consists of **one small tree** with large, contiguous data buffers attached to each node (color coded with the above).

Compiled operations are performed on these data buffers, not the objects they represent.



Broader impacts

High school teachers and students

As a first step toward our "Data science and machine learning" outreach project, we've partnered with Mark Adams (Fermilab, University of Illinois at Chicago) to contribute to their QuarkNet cosmic ray project for high school teachers on August 8–12, 2022. We will help with data analysis, which currently handles ragged arrays in Excel.

We hope to learn from teachers what pedagogy works and what doesn't, in the context of an established program.

Undergraduates and graduate students

We continue to employ undergradate and graduate students through Google Summer of Code and IRIS-HEP fellowships, transitioning to CSSI funding.

In addition, this was the first year we brought a student (Aryan Roy) to SciPy, the scientific Python community's flagship conference. Jim Pivarski also presented a tutorial ("Loopy and unloopy programming"). We intend to scale this up to at most 8 students each year, and to include students beyond our direct collaborators.

Other opportunities

Scientific Python (scientific-python.org) has a strong outreach for new open-source developers, and is looking for partners. We may be able to contribute and collaborate.

Applications in the sciences

CloudDrift: a platform for accelerating research with Lagrangian climate data

NSF #2126413 **PI: Shane Elipot (University of Miami)**

We're helping the authors, particularly Philippe Miron (Florida State University), accelerate "Lagrangian" analyses of ocean currents, which follow measurement probes as they drift on the ocean. It is a situation in which variable-length lists naturally arise, as each probe has a different number of measurements.

Status: subject of a publication presented at EarthCube 2022: DOI 10.5281/zenodo.6792049

launch binder

2022 EarthCube Annual meeting

Accelerating Lagrangian analyses of oceanic data: benchmarking typical workflows

Philippe Miron[1], Shane Elipot[2], Rick Lumpkin[3], Bertrand Dano[2]

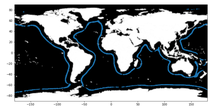
- [1] Florida State University, USA
- [2] University of Miami, USA
- [3] NOAA's Atlantic Oceanographic and Meteorological Laboratory, USA

(Prospective) Improving access to Argo oceanic probe data through Parquet and Awkward Array

The Argo project also has variable-length data from oceanic probes, which are currently served to the community as NetCDF files, partitioned by date, position, and probe number. Conversion to a columnar format shrinks the dataset by 20×, such that the entire historical dataset could fit in RAM (42 GB uncompressed).

The demonstration to the right selects probe measurements by distance from coastline.

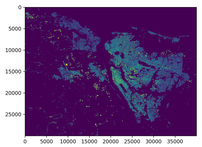
Status: networking; presented to argopy developers and getting in touch with the Argo data manager.

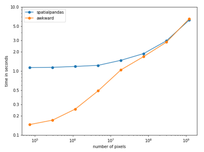


Refactoring spatialpandas using Awkward Arrays and Numba

James Bednar and Ian Thomas (Anaconda, Inc.) want to remove spatialpandas's reliance on a custom ragged array implementation and a custom Pandas schema, in favor of GeoPandas and GeoParquet. Their algorithms are accelerated with Numba, which Awkward Array already supports.

Status: project has started. The map of buildings (polygons) in New York City was generated with Awkward Arrays, faster than the old implementation.





(Prospective) Parquet file format for IceCube observatory

Benedikt Riedel (University of Wisconsin, WIPAC) is considering future file formats for IceCube, which is currently non-standard.

We are testing Parquet, which is an industry standard and can represent the deeply nested/variable-length structures of IceCube data.

Status: acquiring IceCube datasets for initial study.

(Prospective) Accelerating analysis of cosmic air showers

Karl Kosack (CEA Saclay) and Maximilian Nöthe (TU Dortmund) are refactoring the Python interface to the EventIO cosmic air shower data format, which is deeply nested/variable-length.

They will be the first external users of header-only C++ data ingest for Awkward Arrays, at least in a first feasibility study.

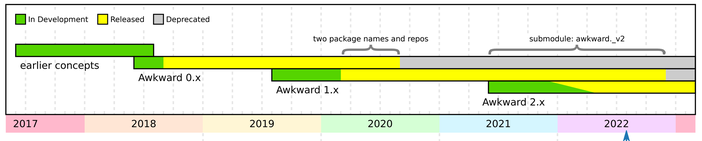
Status: waiting on development of C++ data ingest.

History and usage

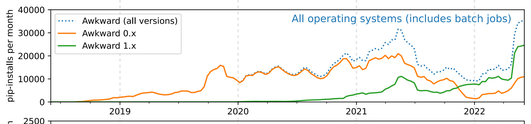
Currently, most users are particle physicists, analyzing data from the LHC.

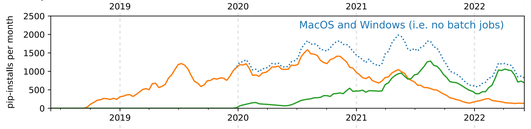
The problems this library solves are not domain-specific, though, and Awkward Array is expanding to other scientific fields.

Development cycle

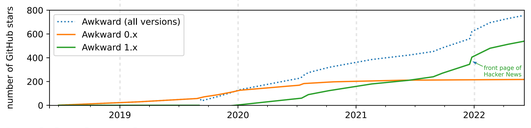


Download statistics





GitHub stars



PyPI critical project

awkward CRITICAL PROJECT
Last released Jul 9, 2022
Manipulate JSON-like data with NumPy-like idioms.

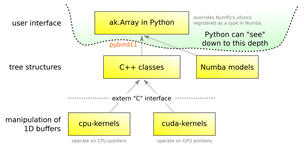
Manage View

Integration with the ecosystem

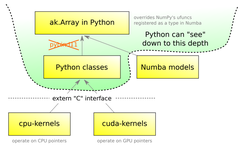
A large part of the codebase was refactored: 60k C++ and 20k Python lines of code were rewritten as 40k Python lines of code (mostly Ioana Ifrim).

Writing the "tree structures" in Python, rather than C++, allows us to use other Python libraries at this level., opening the door to interoperability.

Awkward 1.x



Awkward 2.x



For instance, the "contiguous data buffers," ordinarily NumPy arrays, can be replaced with CuPy to get Awkward Arrays on GPUs, or JAX to get Awkward Arrays that can be used in autodifferentiated functions (Anish Biswas).

CuPy

In doing so, Awkward Array does not lose its ability to interoperate with C++, but the C++ must be JIT-compiled, e.g. in cppyy.

In fact, the same code that implements iteration over Awkward Arrays in Numba (JIT-compiled Python) is now used to iterate over Awkward Arrays in JIT-compiled C++ (Ianna Osborne and Manasvi Goyal).

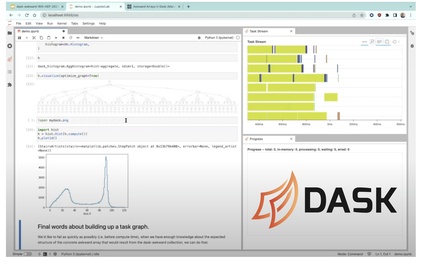
The C++ code that builds Awkward Arrays is in the process of being isolated in a header-only library without Python or Awkward Array dependencies.

Numba

Dask-Awkward Array collection

We also replaced Awkward Array's lazy arrays with a new Dask collection type for Awkward Arrays (Douglas Davis).

Dask is an industry standard for delayed and distributed computation, enabling horizontal scale-out across a cluster.



Relationship to Apache Arrow

Arrow has a similar data structure, and most Awkward Arrays are zero-copy convertible to and from Arrow arrays.

Awkward Array can be viewed as one way to perform computations on Arrow data, in a way that is unlike anything else in the Arrow ecosystem (which focuses on SQL).

Parquet, through pyarrow, is now the default file format for Awkward Arrays.

Parquet

And more...

We're participating in the scientific-python.org community organization for better communication with developers of other libraries.

Scientific Python

Team

Regular contributors to Awkward Array during the time of this grant, September 2021 – present.

PRINCETON UNIVERSITY ANACONDA.

(research scientists)

Jim Pivarski
Princeton University

Henry Schreiner
Princeton University

(research software engineers)

Ianna Osborne
Princeton University

Ioana Ifrim
Princeton University

(postdoc)

Angus Hollands
University of Birmingham → Princeton University

(undergraduates)

Anish Biswas
Manipal Institute of Technology

Manasvi Goyal
Delhi Technological University

Aryan Roy
Manipal Institute of Technology

(contractors)

Douglas Davis
Anaconda, Inc.

Martin Durant
Anaconda, Inc.

24 more contributors on GitHub