Leveraging ServiceX to Transform PHYSLITE Data into Flat N-tuples with Systematics

Alexander Schmidt

The Problem at Hand:

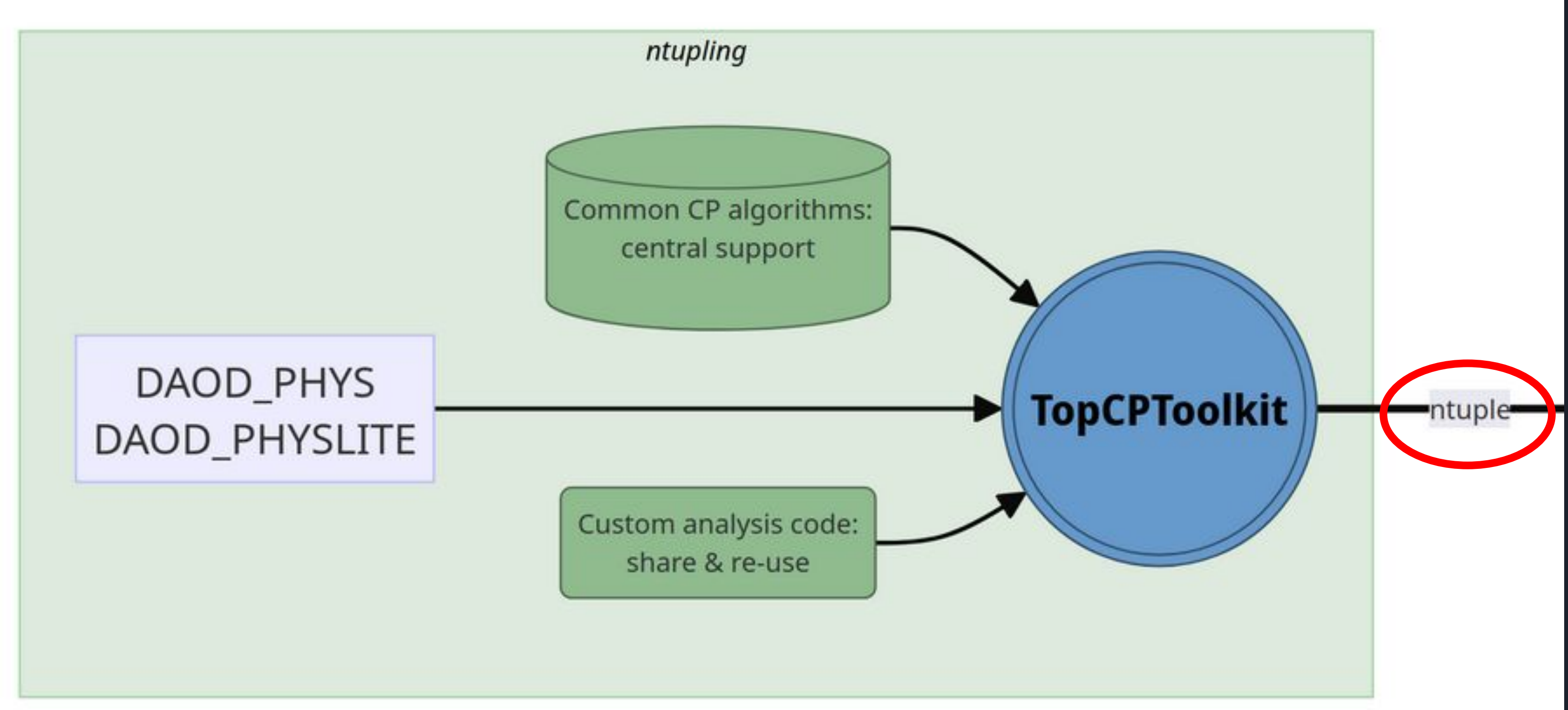
The intended main storage format for ATLAS is a format called PHYSLITE. The issue with PHYSLITE is that we need to be able to apply systematic variations to its data, but the code to do so is currently not compatible with columnar analysis or the Python ecosystem. One way to address this, at least in the short term, is to run those systematic variations as part of creating flat ntuples out of PHYSLITE, which is something we have existing software (TopCPToolkit) to do. So…

The Idea:

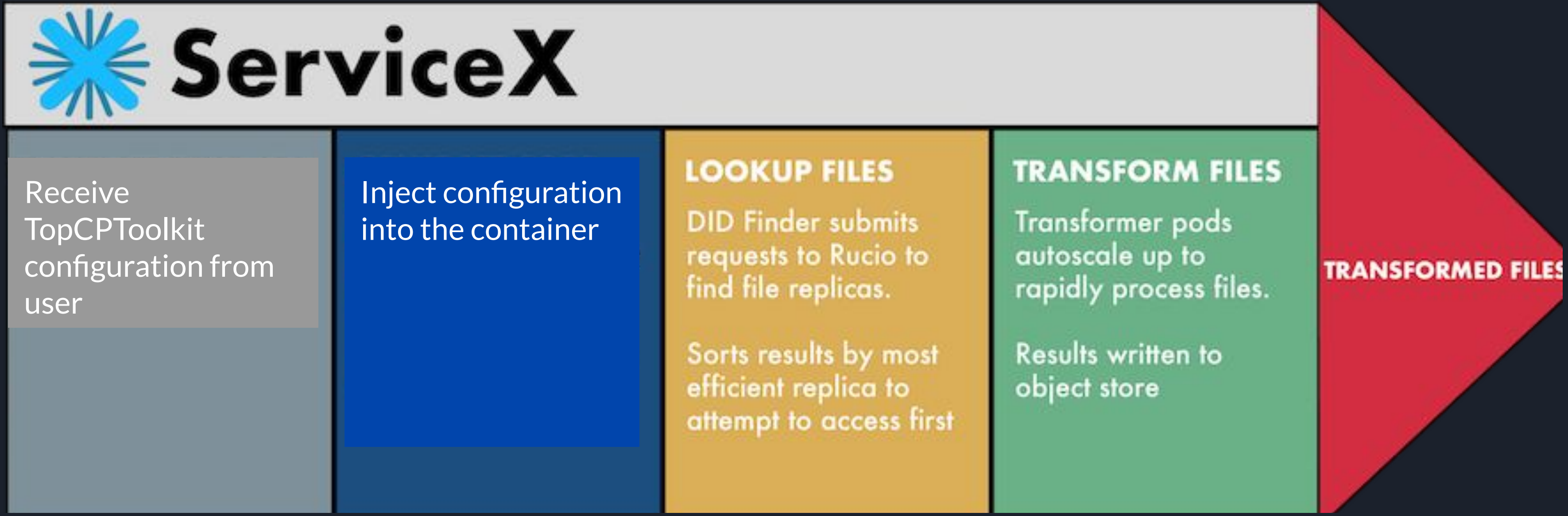
- Encapsulate TopCPToolkit as a ServiceX transformation
- Access the grid via ServiceX and run analyses with TopCPToolkit all in one go
- Should be easy to use
- Should work on any system
- Should include any desired data especially systematics

What does this look like in practice?

TopCPToolkit



ServiceX



ServiceX

Receive TopCPToolkit configuration from user

Inject configuration into the container

LOOKUP FILES

DID Finder submits requests to Rucio to find file replicas.

Sorts results by most efficient replica to attempt to access first

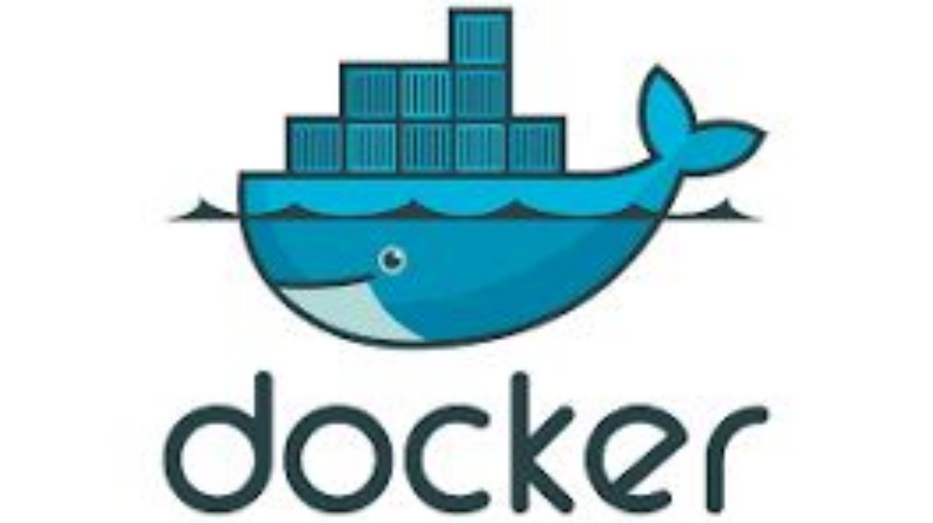
TRANSFORM FILES

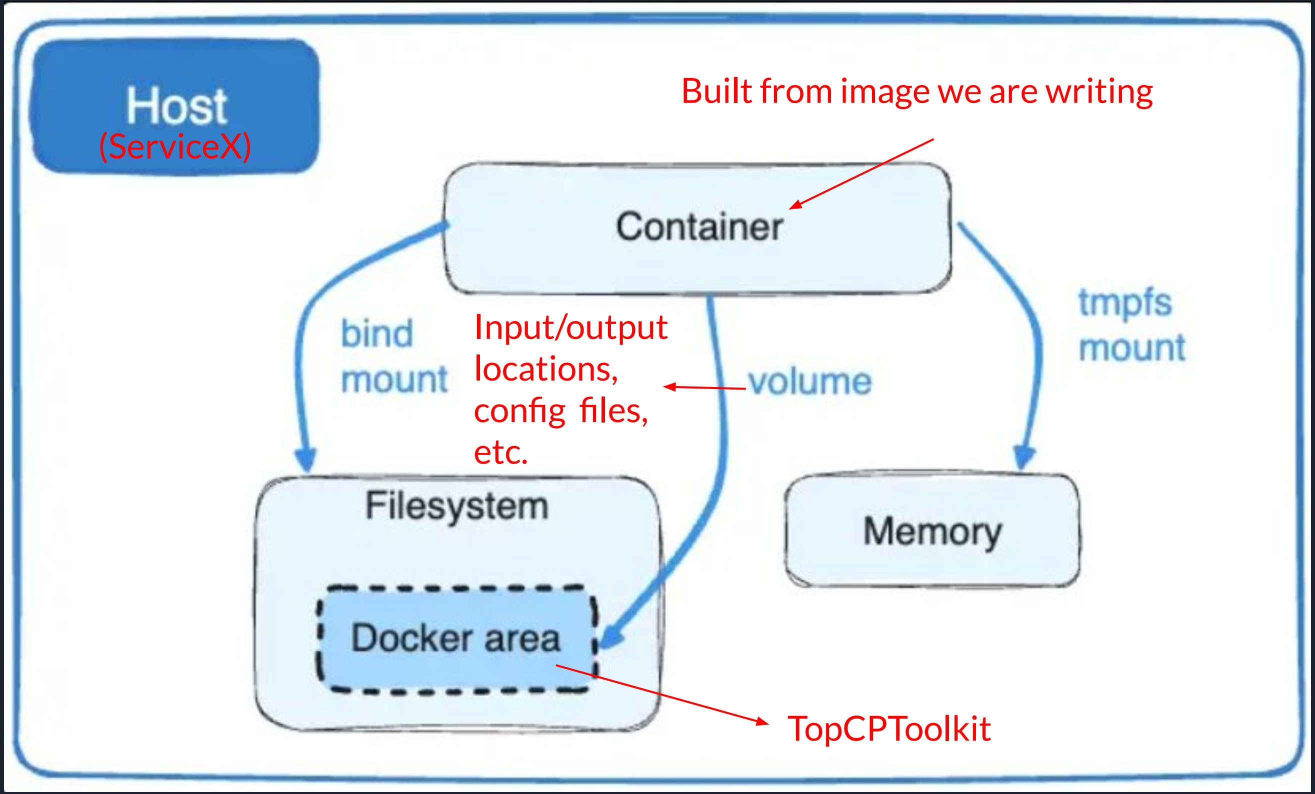
Transformer pods autoscale up to rapidly process files.

Results written to object store

TRANSFORMED FILES

Docker





(ServiceX)

Built from image we are writing

Input/output locations, config files, etc.

TopCPToolkit

TopCPToolkit

+ Docker

+ ServiceX

= Final Product 👍

Questions?