

Floability: Enabling Portable Scientific Workflows Across HPC Facilities

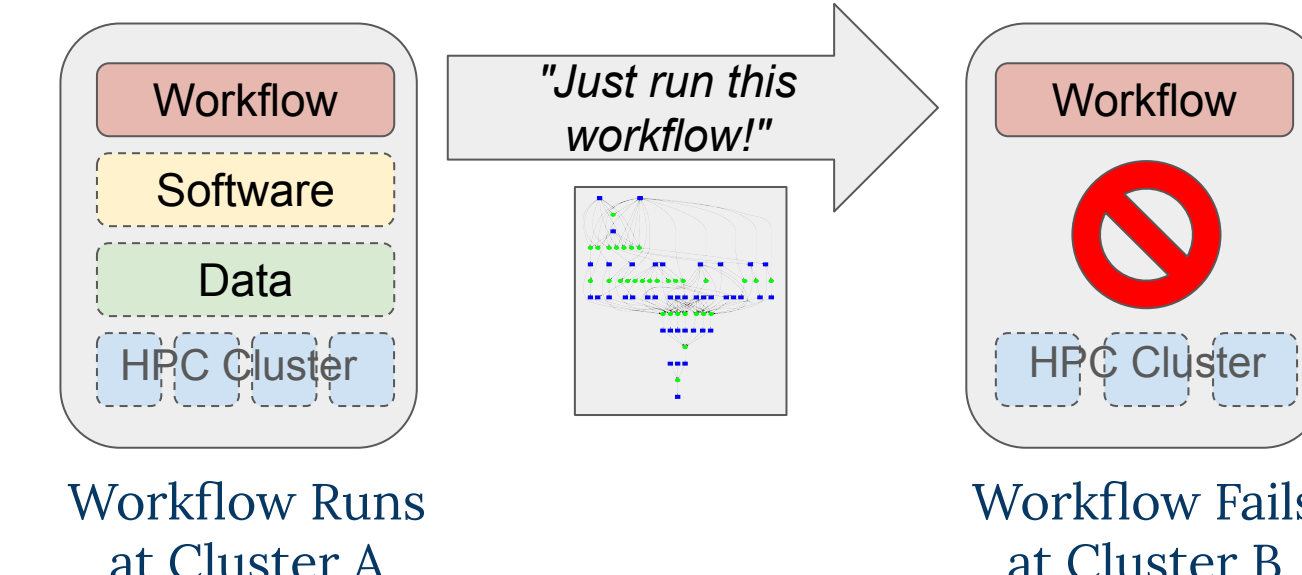
Md Saiful Islam (mislam5@nd.edu)

On behalf of the Floability team: Prof. Tanu Malik, Prof. Kevin Lannon, Prof. Shaowen Wang, Prof. Douglas Thain, Ben Tovar, Md Saiful Islam, Talha Azaz, Dr. Furqan Baig, A S M Shahadat Hossain, and Raza Ahmad

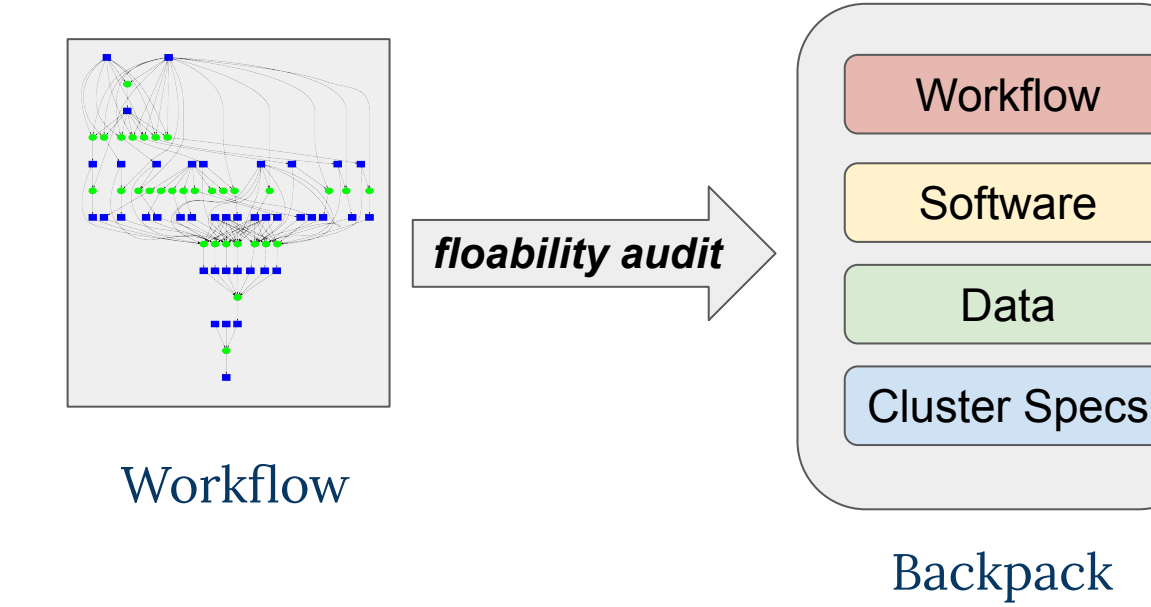
Scientific workflows often consist of millions of tasks arranged in a DAG.



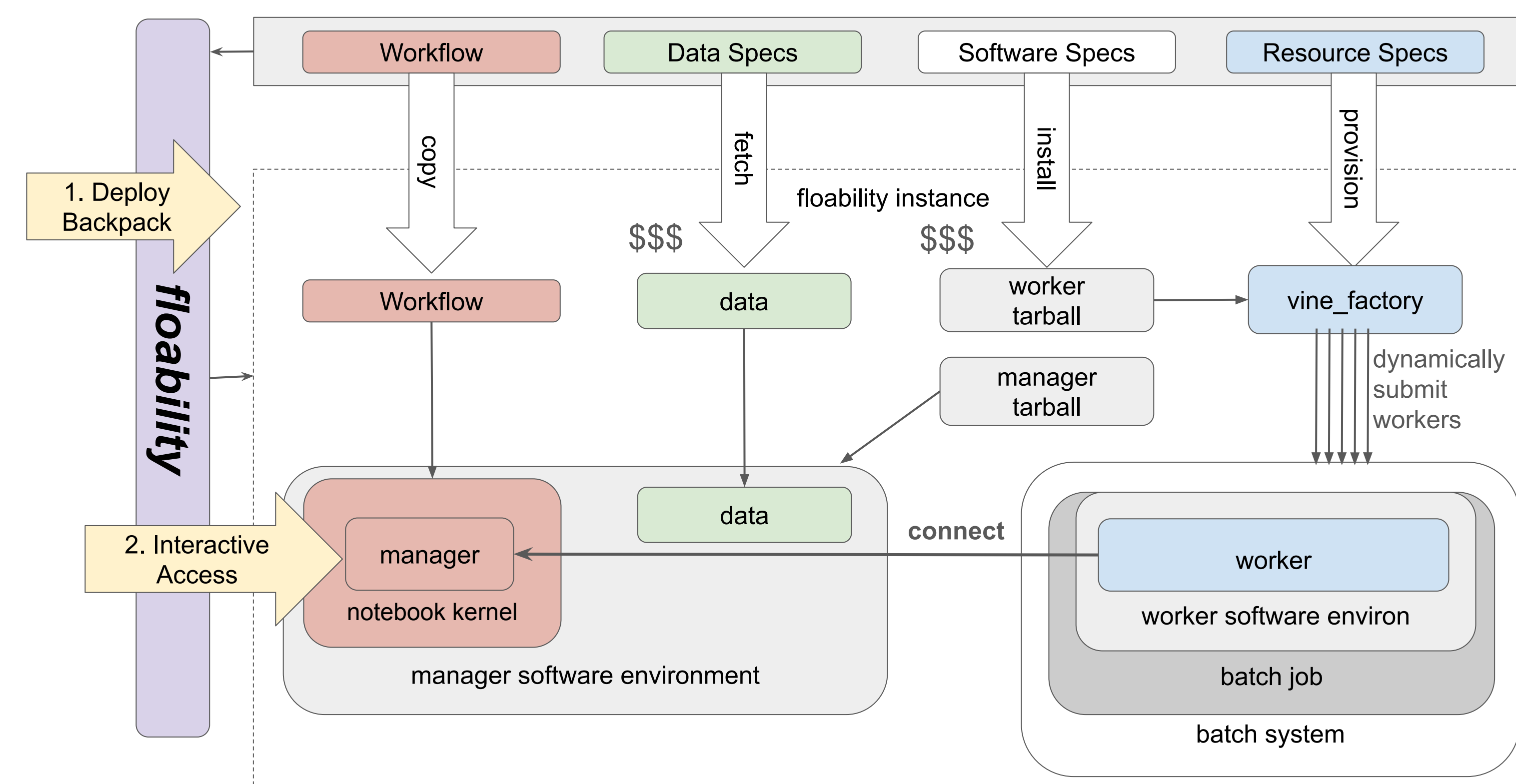
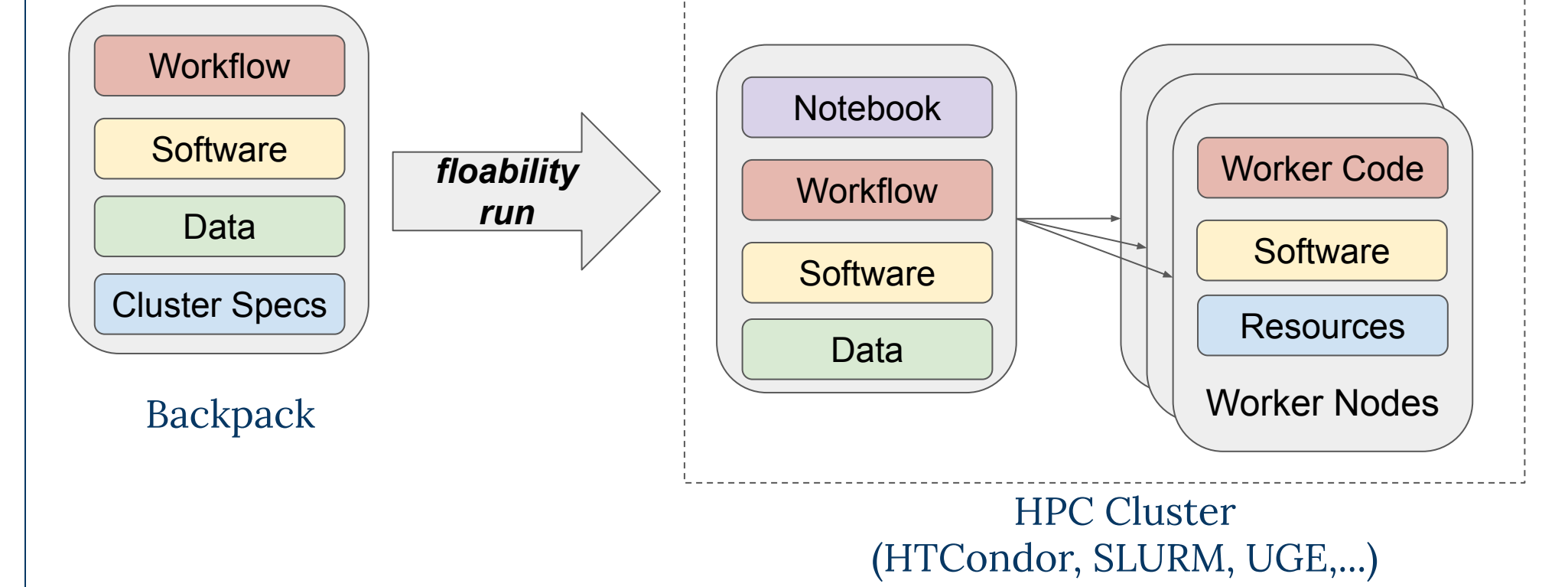
A workflow by itself cannot run without the data and its supporting environment.



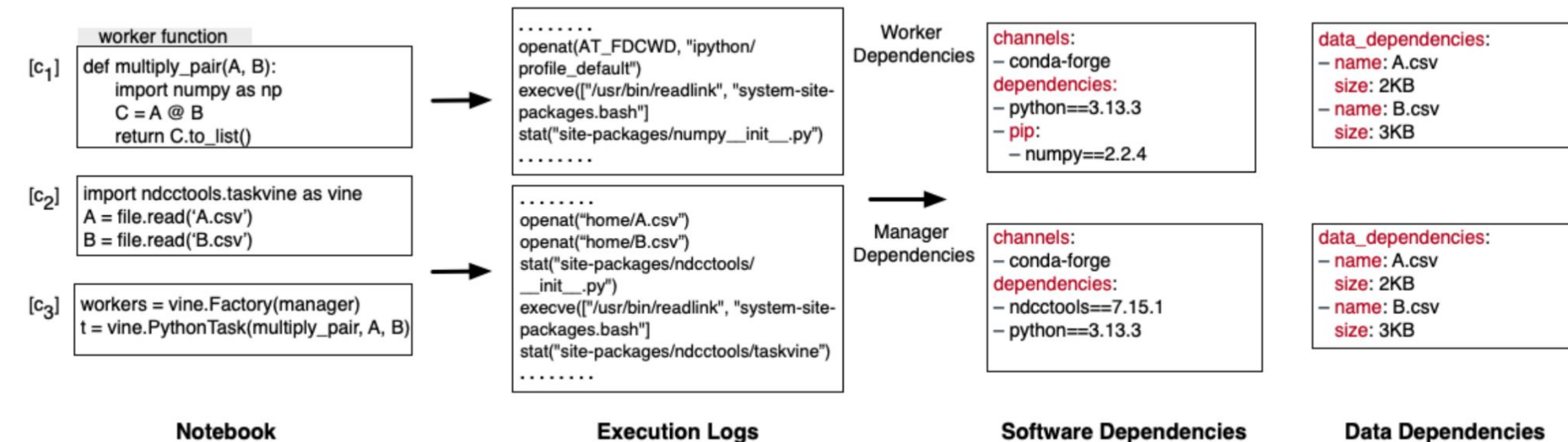
A backpack specifies all dependencies needed to execute a notebook-based workflow.



Floability deploys a backpack into an HPC cluster.



Floability architecture consists of a manager running on the head node and multiple worker processes running on compute nodes. A factory process is responsible for launching and dynamically scaling workers in the cluster.

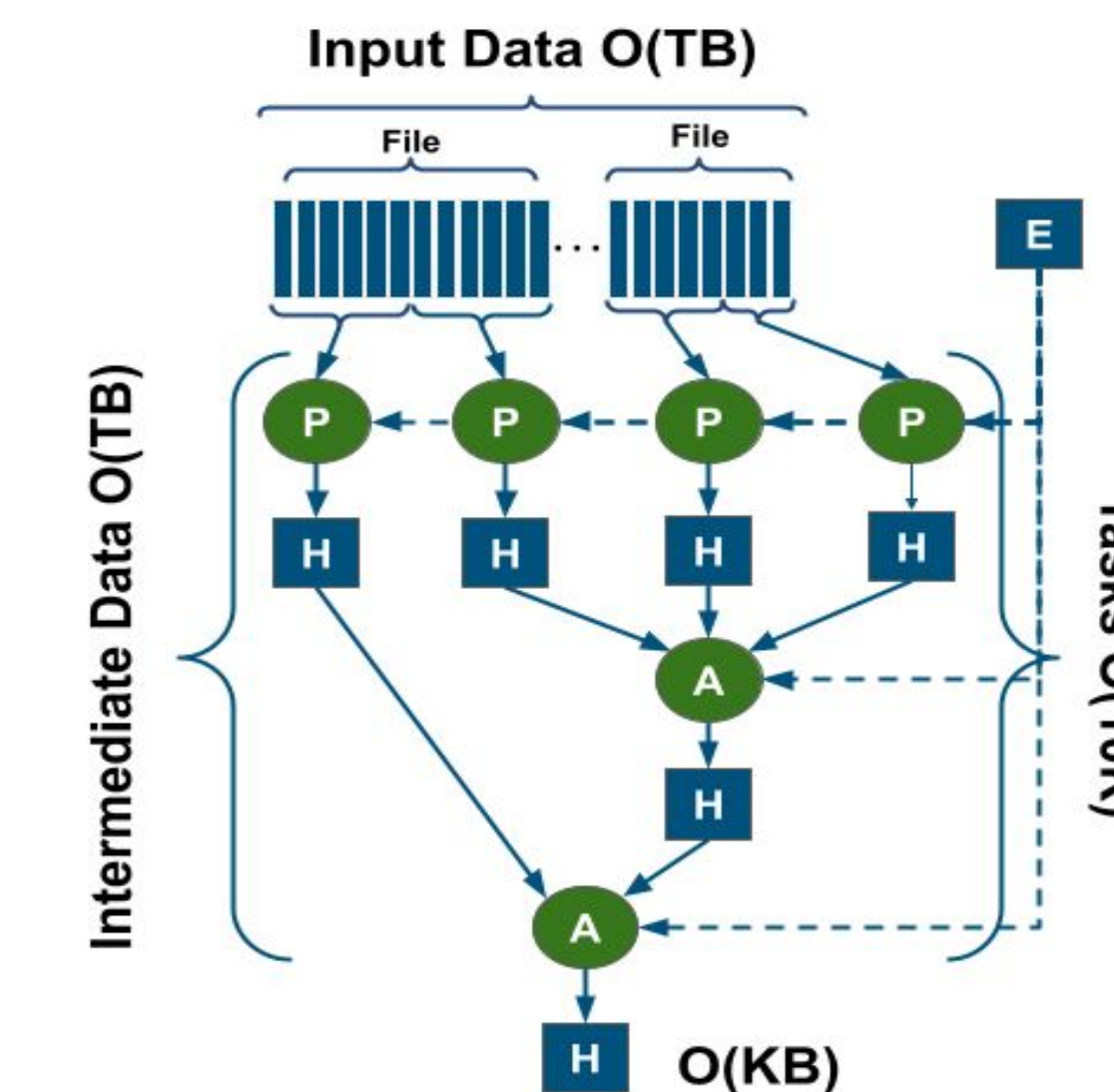


Floability *audit* uses system call tracing to extract manager and worker software packages, which are then validated to produce an environment file with precise versions.

Example Applications

DV5 CMS Analysis Application

Consumes 1.5TB Data
Submits 17K Tasks, Uses 2400 cores, 200 nodes.



```
def analysis(events):
    warnings.filterwarnings("ignore", module="coffea.*")
    dataset = events.metadata["dataset"]
    events["PFCandidates", "pt"] = events.PFCandidates.pt * events.PFCandidates.puppiWeight
    cut_to_fix_softdrop = ak.num(events.FatJet.constituents.pt, axis=2) > 0
    events = events[ak.all(cut_to_fix_softdrop, axis=1)]

    trigger = ak.zeros_like(ak.firsts(events.FatJet.pt), dtype="bool")
    for t in triggers["2017"]:
        if t in events.HLT.fields:
            trigger = trigger | events.HLT[t]
    trigger = ak.fill_none(trigger, False)
    events["FatJet", "num_fatjets"] = ak.num(events.FatJet)

    tasks = dataset_tools.apply_to_fileset(
        analysis,
        samples_dict,
        uproot_options={},
        schemaclass=PFNanoAODSchema,
    )

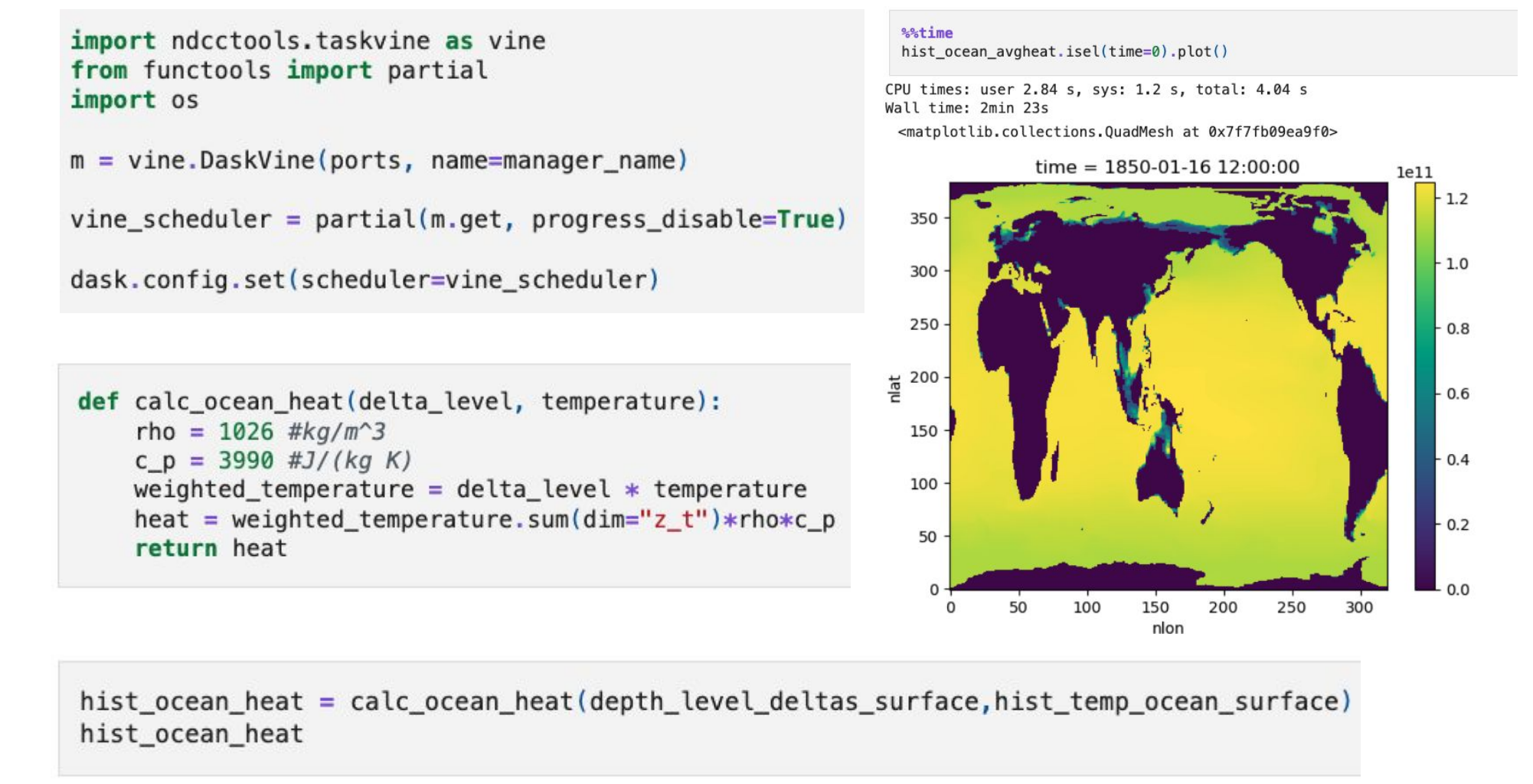
    m = DaskVine(ports, name=manager_name)
```



Provided By: Kevin Lannon and Connor Moore

Surface Ocean Heat (CESM2)

Data: CESM2 LENS 1850-2100
Tasks: 2800+ parallel jobs for each plot

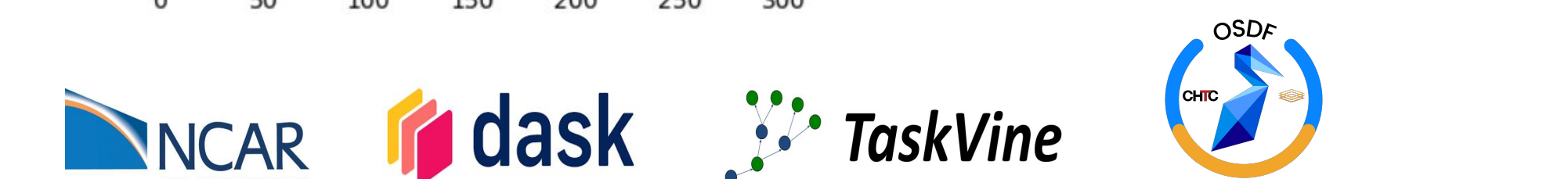
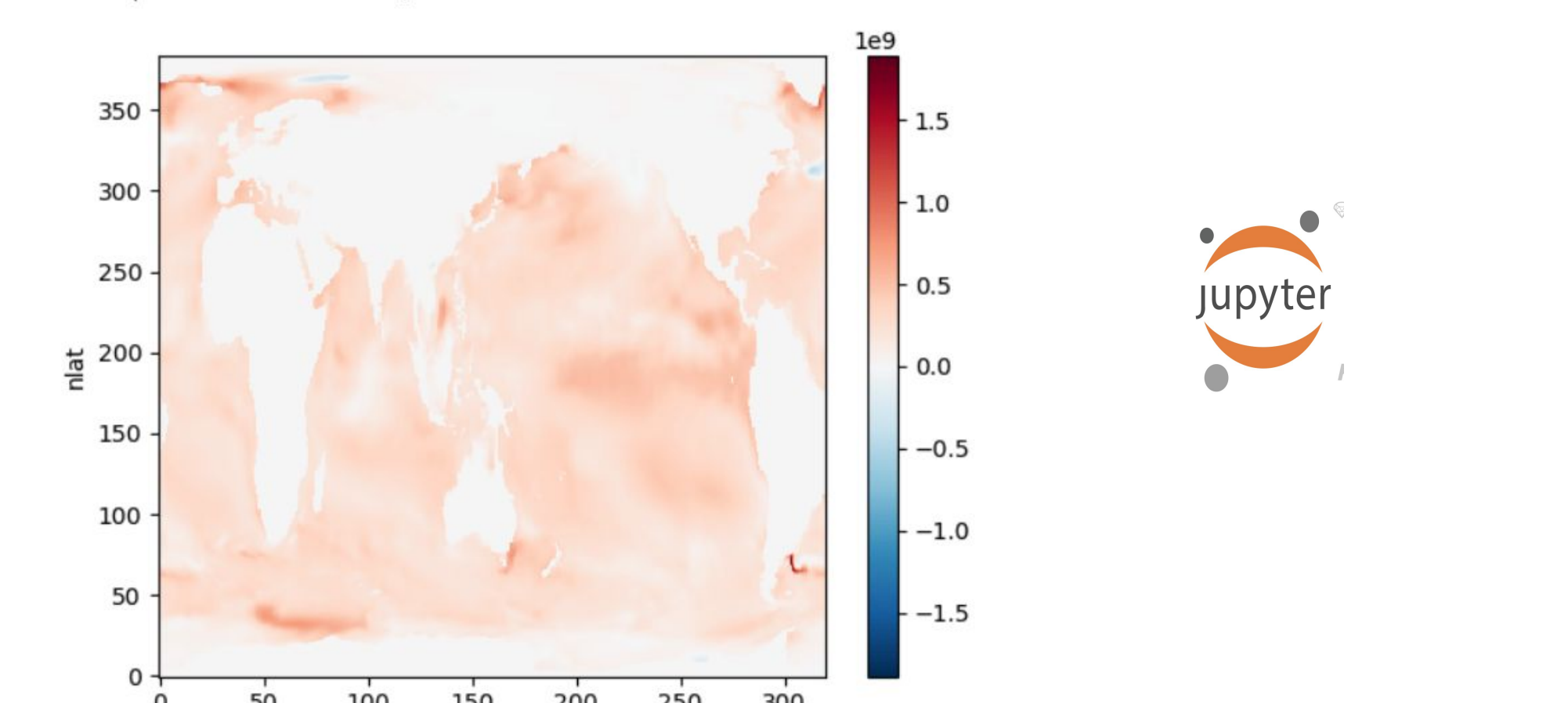


Has the surface ocean heat content increased with time for January? (Due to Global Warming!)

```
hist_ocean_avgheat_ano = hist_ocean_avgheat.isel(time=1) - hist_ocean_avgheat.isel(time=0)

%time
hist_ocean_avgheat_ano.plot()
```

CPU times: user 3.75 s, sys: 997 ms, total: 4.75 s
Wall time: 1min 1s
~matplotlib.collections.QuadMesh at 0x7f7fb158a120~



Provided By: Harsha Hampapura

