`nbscuid`: Towards a CESM Diagnostics Workflow Built on Jupyter Notebooks

Elena Romashkova, Matt Long, Deepak Cherian,
Gustavo Marques, Keith Lindsay

This material is based upon work supported by the National Center for Atmospheric Research, which is a major facility sponsored by the National Science Foundation under Cooperative Agreement No. 1852977.

Introducing the workflow

Goals

- Notebook-based for easy sharing and annotating, with support for scripts for back-compatibility
- Flexible diagnostic framework - run out of the box or customize
- Catalog-friendly for simpler data access
- Multiple options for computational resources

Intro

README.md

nbscuid

Notebook-Based, Super CUstomizable Infrastructure for Diagnostics

Installation

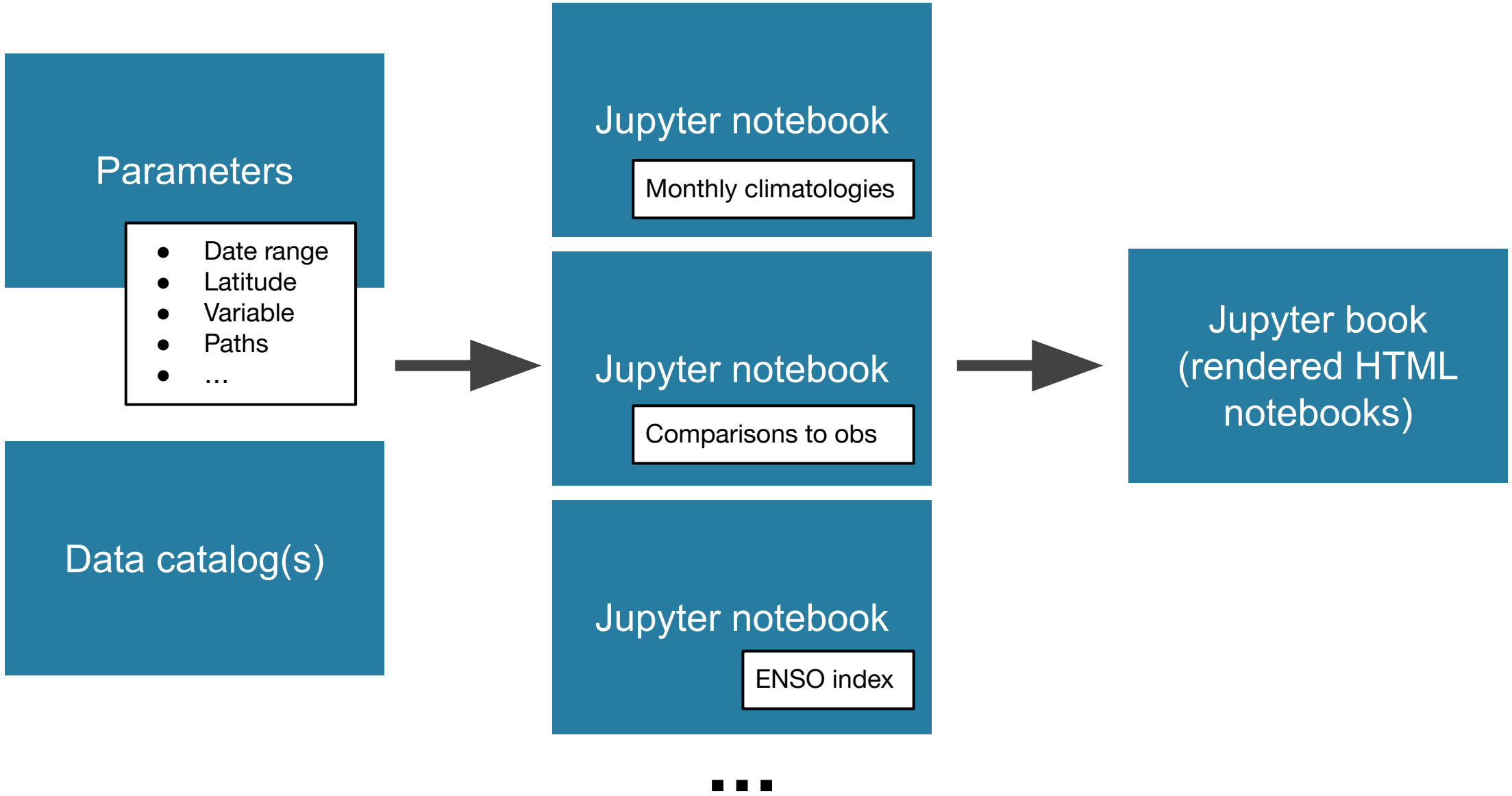
- Run:

```
pip install nbscuid
```

This is a package to enable running notebook-based diagnostic workflows. Based on my-cesm-experiment by matt-long: https://github.com/matt-long/my-cesm-experiment.

See some examples of workflows at https://github.com/rmshkv/nbscuid-examples. For a basic tutorial, follow https://nbscuid.readthedocs.io/en/latest/tutorialsetup.html.

Workflow overview



Tool stack

- **papermill** for parameterizing and executing notebooks
- **jinja** for parameterizing Markdown cells
- **dask** for parallelization
- **intake-esm** for catalog parsing
- **esm_catalog_utils** (cc Keith Lindsay) for catalog creation
- **jupyter book** for turning Jupyter notebooks into publishable HTML

Executing a diagnostic collection

config.yml

- data catalog
- path to template notebooks
- global params
- notebook-specific params
- Jupyter Book config

```
nbscuid-run
config.yml
```

- passes params to notebooks
- executes notebooks through **papermill**

Set of executed notebooks

```
nbscuid-build
config.yml
```

Jupyter book

Demo

https://github.com/rmshkv/nbscuid-examples

Demo - backup

```
(base) crhtc47:~/codes/nbsc...smworkshop-demo> nbscuid-run config.yml
/glade/work/eromashkova/miniconda3/lib/python3.9/site-packages/distributed/node.py:183: UserWarning: Port 8787 is already in use.
Perhaps you already have a cluster running?
Hosting the HTTP server on port 41350 instead
  warnings.warn(
Waiting for Dask workers....

Executing /glade/u/home/eromashkova/codes/nbscuid-examples/cesmworkshop-demo/nblibrary/index.ipynb
Executing: 100%|                                                                                | 4/4 [00:02<00:00,  1.84cell/s]
Executing /glade/u/home/eromashkova/codes/nbscuid-examples/cesmworkshop-demo/nblibrary/global_TS_levels.ipynb
Executing:  43%|                                                                                | 10/23 [00:07<00:06,  2.02cell/s]
```

```
(base) crhtc47:~/codes/nbsc...smworkshop-demo> nbscuid-build config.yml

================================================================================

Your _build directory has been emptied except for .jupyter_cache

================================================================================

Running Jupyter-Book v0.15.1
Source Folder: /glade/u/home/eromashkova/codes/nbscuid-examples/cesmworkshop-demo/computed_notebooks/cesmworkshop-demo-run1
Config Path: /glade/u/home/eromashkova/codes/nbscuid-examples/cesmworkshop-demo/computed_notebooks/cesmworkshop-demo-run1/_config.yml
Output Path: /glade/u/home/eromashkova/codes/nbscuid-examples/cesmworkshop-demo/computed_notebooks/cesmworkshop-demo-run1/_build/html
Running Sphinx v4.5.0
making output directory... done
```

Demo - backup

Example project

Diagnostic notebooks

Temperature and salinity biases at selected depth levels

```
import warnings
warnings.filterwarnings("ignore")
```

```
Basemap module not found. Some regional plots may not function properly
```

```
# Empty cell with "parameters" tag, papermill-provided parameters will be inserted below.
```

```
# Parameters
diag_config_yml = {
    "Avg": {"end_date": "0061-12-31", "start_date": "0031-01-01"},
    "Case": {
        "CASEROOT": "/glade/work/gmarques/cesm.cases/G/gmom.e23.GJRAv4.TL319_t061_zstar_N65.nuopc.HBD.
        "CIMEROOT": "/glade/work/gmarques/cesm.sandboxes/cesm2_3_alpha08a.sbx/cime/",
        "OCN_DIAG_ROOT": "/glade/work/gmarques/Notebooks/for_elena/ncfiles/",
        "SNAME": "HBD_zstar",
    },
}
sname = "cesmworkshop-demo-run1"
test_param = "This parameter was inserted!"
woa_path = "/glade/u/home/gmarques/Notebooks/CESM_MOM6/WOA18_remapping/"
WOA18_temp_path = "/glade/u/home/gmarques/Notebooks/CESM_MOM6/WOA18_remapping/WOA18_TEMP_tx0.66v1_34le
WOA18_salt_path = "/glade/u/home/gmarques/Notebooks/CESM_MOM6/WOA18_remapping/WOA18_SALT_tx0.66v1_34le
cluster_scheduler_address = None
subset_kwargs = {}
```

Demo - backup

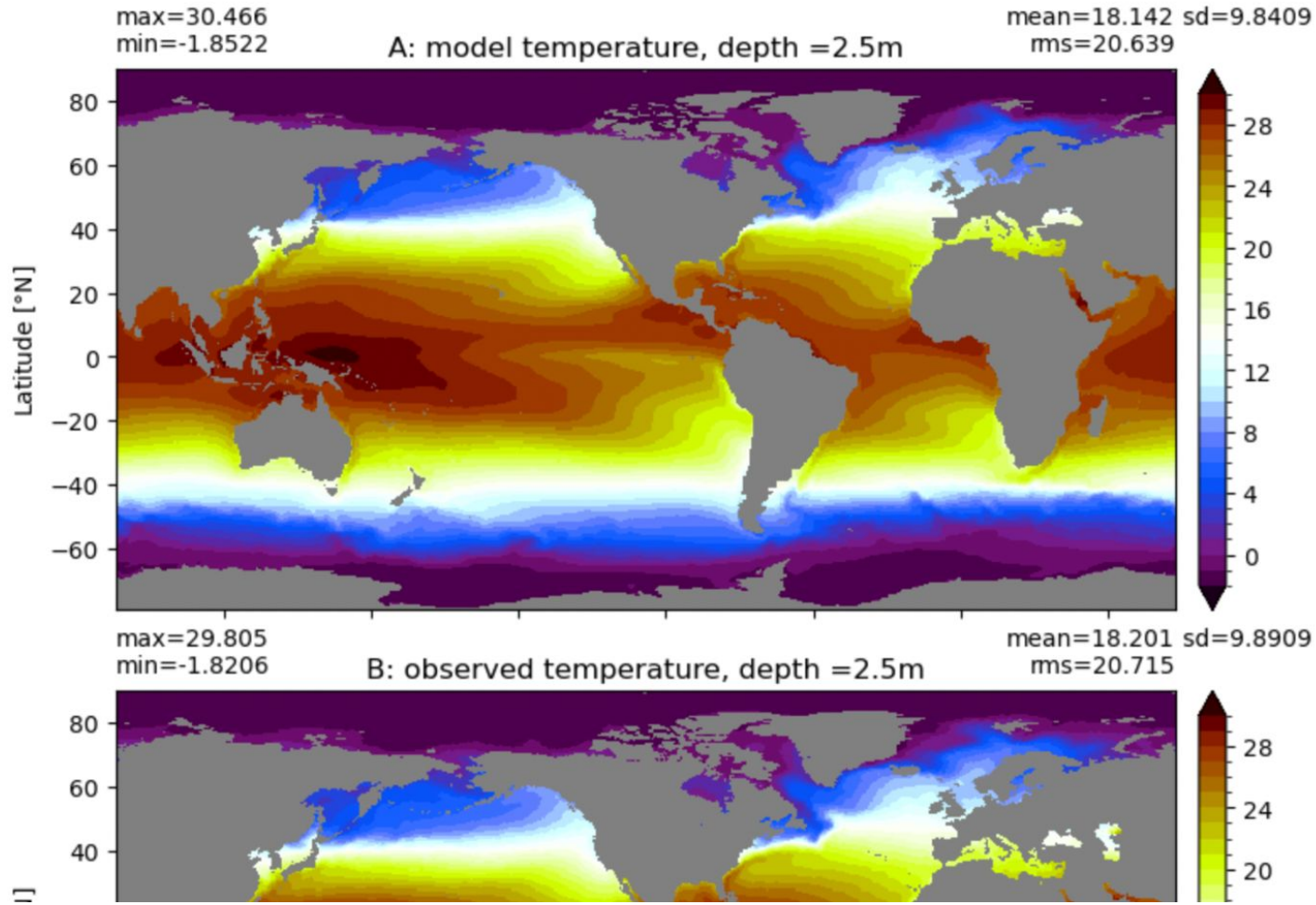
Example project

Diagnostic notebooks

Temperature and salinity biases at selected depth levels

```
            suptitle=dcase.casename + ', averaged '+str(start_date)+ ' and ' +str(end_date),
            clim=(-1.9,30.), dcolormap=plt.cm.bwr,
            extend='both', dextend='neither', dlim=(-2,2),
            show= True)
```

gmom.e23.GJRAv4.TL319_t061_zstar_N65.nuopc.HBD.002, averaged 0031-01-01 and 0061-12-31



Other features

- Run out of the box with a premade config.yml, or customize your own
- Can run any kind of notebook, not just CESM diagnostics
- Run a single notebook on different sets of parameters
- Run notebooks in different environments
- Cache results

Current work and open questions

Experimenting with MOM6 diagnostics

- Current workflow:
  - Diagnostic functions: **mom6-tools**
  - Series of python scripts configured by a yaml file and submitted via bash script through **qsub**
  - Create output files that are displayed through notebooks in **mom6_solutions**
- Goal: converting these diagnostics to be compatible

Next steps to implement

- Executing diagnostics in parallel
- Running non-notebook diagnostics (like .py files)

Managing computational resources

Currently:

- Create a “global” dask cluster on Casper
  - Wait for at least one worker to appear
  - Pass its scheduler address to each notebook
- Each notebook creates a client and attaches it to the global cluster
- Notebooks run in serial

Want:

- Notebooks run in parallel
- Running locally and on non-NCAR machines

Leveraging existing data pipeline packages?

- **Ploomber** - parallelizing notebooks, capability to run Python scripts, creating a more complex task graph to pass data more flexibly between diagnostics elements
- Potentially others - suggestions welcome!

Joining efforts with other CESM diagnostics?

- Currently several diagnostics efforts around NCAR
- Potential collaboration with ADF

Links

Main repo:
https://github.com/rmshkv/nbscuid

Docs: https://nbscuid.readthedocs.io

Usage examples:
https://github.com/rmshkv/nbscuid-examples

Contact me: eromashkova@ucar.edu