UFS Land-DA Workflow

- Day 2: Structure and Features -

Chan-Hoo Jeon, Gillian Petro, Jong Kim, Eddie Snyder, Kris Booker

NOAA Earth Prediction Innovation Center (NOAA/EPIC)

https://epic.noaa.gov/

Contents

- Components of workflow
- NOAA operational implementation standards
- Structure of workflow
- Tasks of workflow
- Workflow manager: Rocoto
- Configuration of workflow
- Log files

Key Components of Land-DA Workflow

- **Land**: UFS (**U**nified **F**orecast **S**ystem) Weather Model
  - Coupled model: atmosphere (FV3+CCPP), ocean (MOM6/HYCOM), ice (CICE), land (**Noah-MP**), air quality (CMAQ), atmospheric data (DATM), wave (WaveWatch III)
  - Coupling options available in UFS land-DA workflow:
    - Option 1 (APP=**LND**): land (Noah-MP) + atmospheric data (DATM)
    - Option 2 (APP=**ATML**): land (Noah-MP) + atmosphere (FV3+CCPP)
- **DA**: JCSDA JEDI (**J**oint **E**ffort for **D**ata-assimilation **I**ntegration)
  - **External** component: built separately for efficiency (**not** included in workflow repo.)
  - Algorithms: 3D-Var, LETKF (Local Ensemble Transform Kalman Filter)
  - IODA (Interface for Observation Data Access) converters
  - Snow observation data options: GHCN, IMS, SFCSNO
- **Workflow**: Pre-processing / Post-processing Tools
  - JCB (JEDI Configuration Builder), tile2tile_converter, IODA converting scripts, and python scripts

NOAA Operational Implementation Standards

- NCEP Central Operations (NCO) WCOSS Implementation Standards
  - NCEP (**N**ational **C**enter for **E**arth **P**rediction) in NOAA/NWS (National Weather Service)
  - WCOSS (**W**eather & **C**limate **O**perational **S**upercomputing **S**ystem): NCO's operational machine
  - Version: 11.0.0 (January 19, 2022)
- All operational packages (models) must follow the NCO standards.
- The vertical directory structure of the operational packages is defined in the standards.
- A standard set of environment parameters/variables such as 'RUN', 'PDY', and 'envir' are defined in the standards to simplify the production workflow.
- The standard file naming conventions are defined in the standards.
- The production utilities are provided with the 'prod_util' module.
- In the production environment, all jobs are scheduled and submitted to the WCOSS resource manager 'ecFlow'.

The UFS land-DA workflow currently uses 'Rocoto' as a workflow manager.

Advantages of Following NCO Standards

- The NCO standards is the only **official standards** across the NOAA organizations. When multiple organizations collaborate for one model (package), they should have a code standards for consistency and efficiency.
- The **ultimate goal** of the NOAA models is to be one of the NOAA operational models. Following the NCO standards would be essential to the NOAA models.
- **R2O** (Research-to-Operation) and **O2R** (Operation-to-Research) are the important topics in NOAA. When the structure of a research model is the same as that of the operational models, the R2O/O2R process can be accelerated.
- The operational models run **multiple cycles** for weather forecast at NCO every day. To do this, the operational standards is designed to optimize the cycled runs with multiple models. Therefore, it would be the best option for a model intended for the reanalysis experiments running for multi-days/months.
- The models and tools developed by NOAA/EMC are supposed to follow the NCO standards. The UFS land-DA workflow uses the **components** (submodules) developed by EMC.

Github Repository of UFS Land-DA Workflow

URL of authoritative GitHub repository

Develop branch

(hidden) CI/CD test: Jenkins nightly ctest

Documentation

Static input (fix) files

J-job scripts

Machine-specific module (library) files

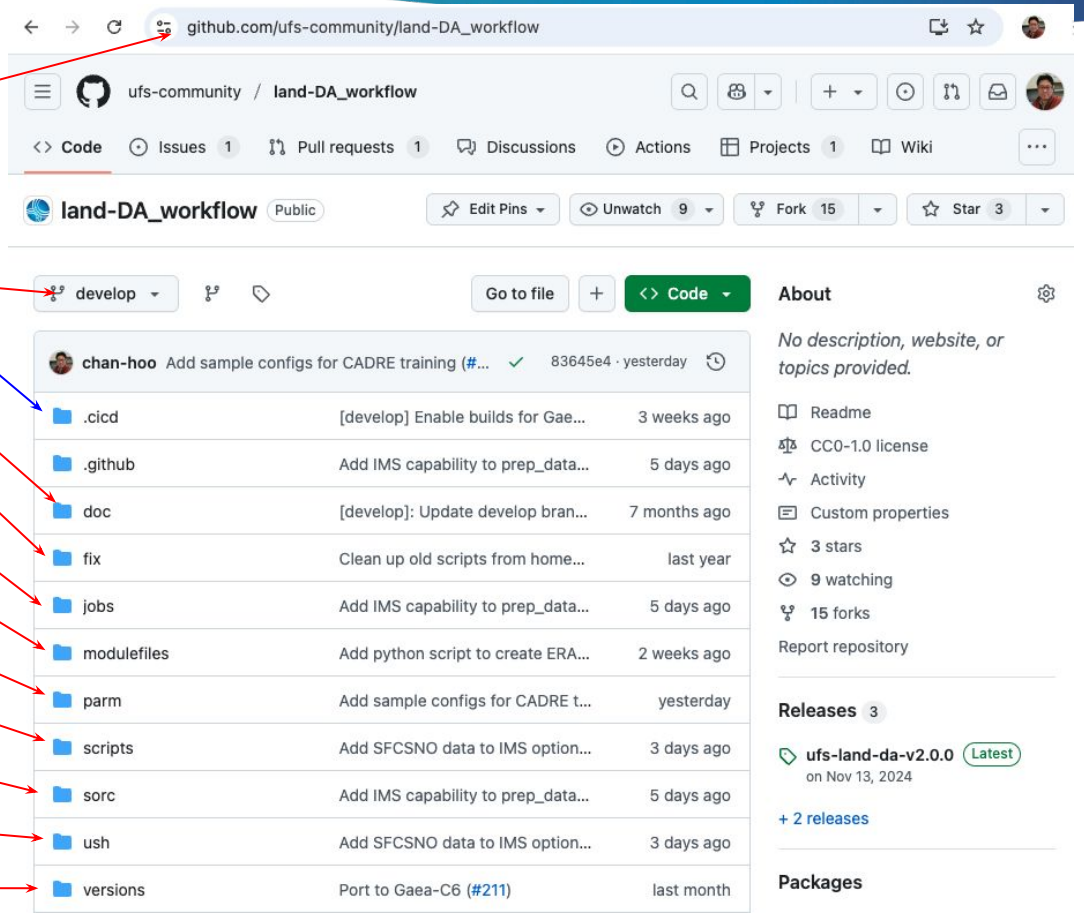
Parameter files and templates

Main script for each task

Submodule components and build scripts

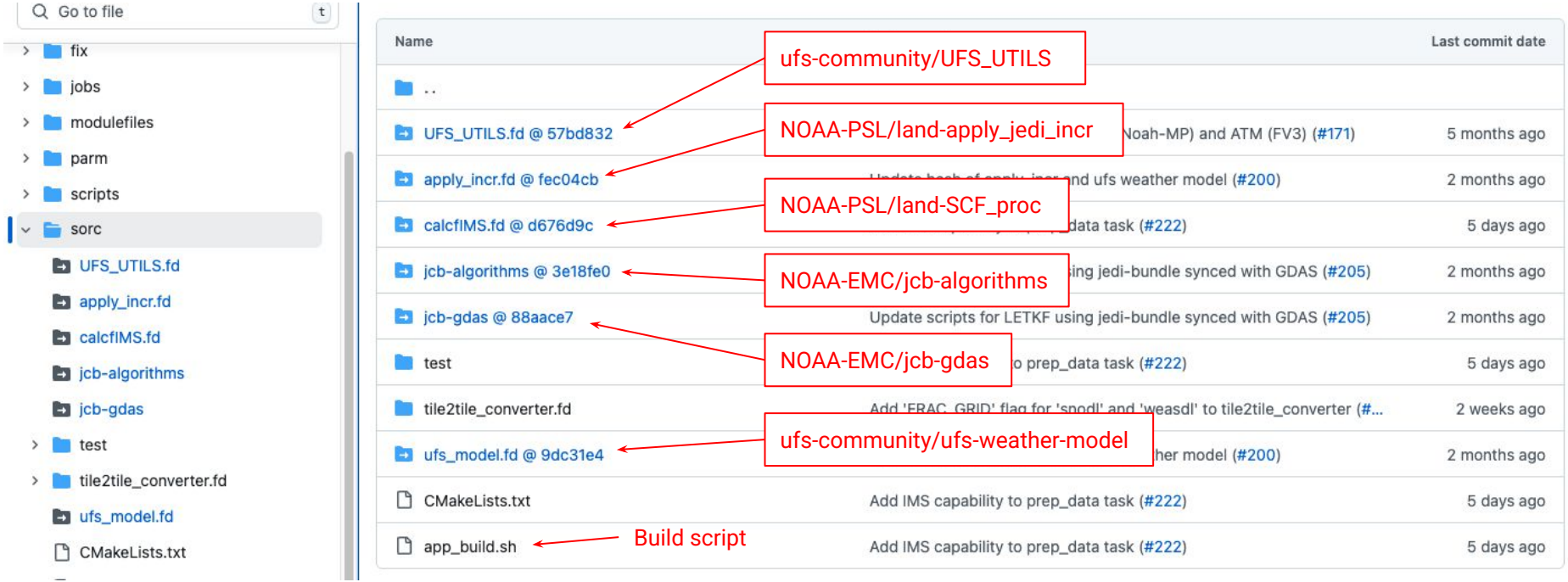
Utility scripts

Version files



Executables (submodules) of Workflow

- The Git submodules point to the specific commits of other external repositories.

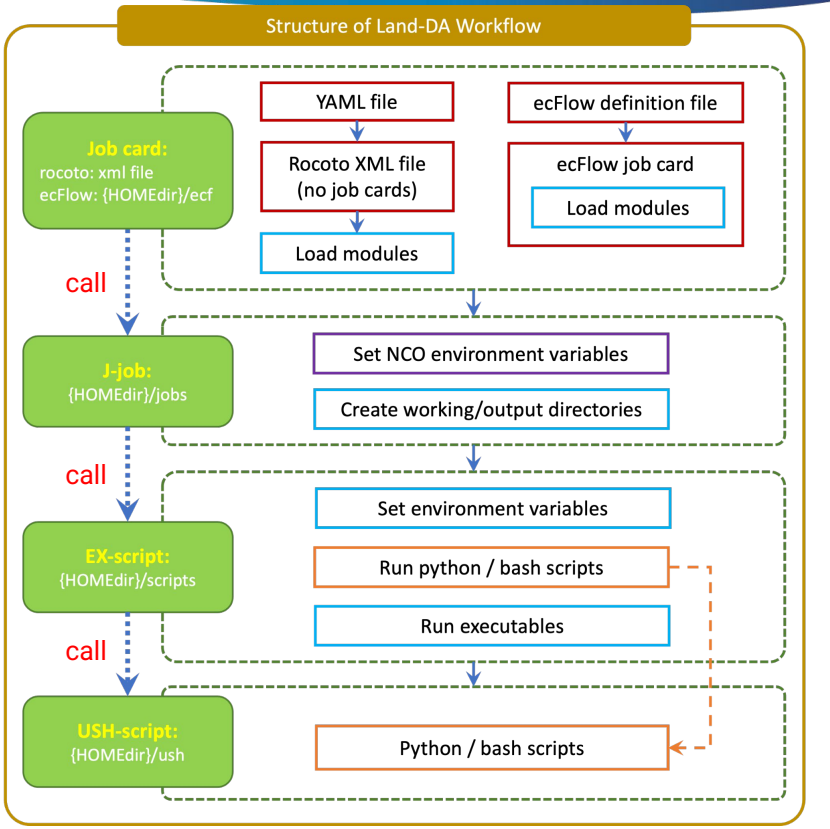


Note that the JEDI executables are not included in the repository.

https://epic.noaa.gov/

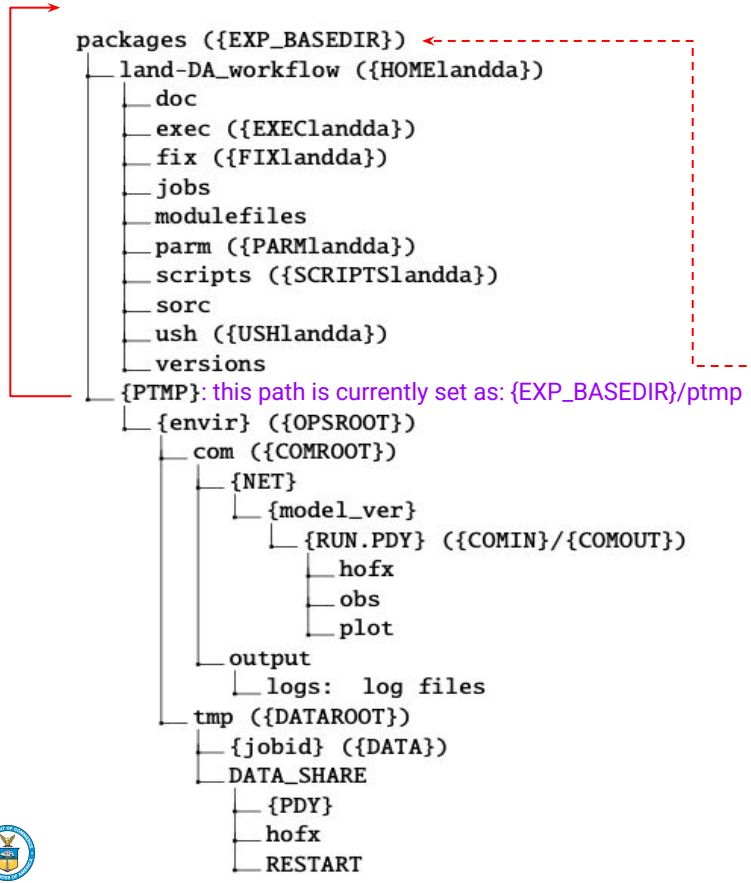
Structure of Workflow

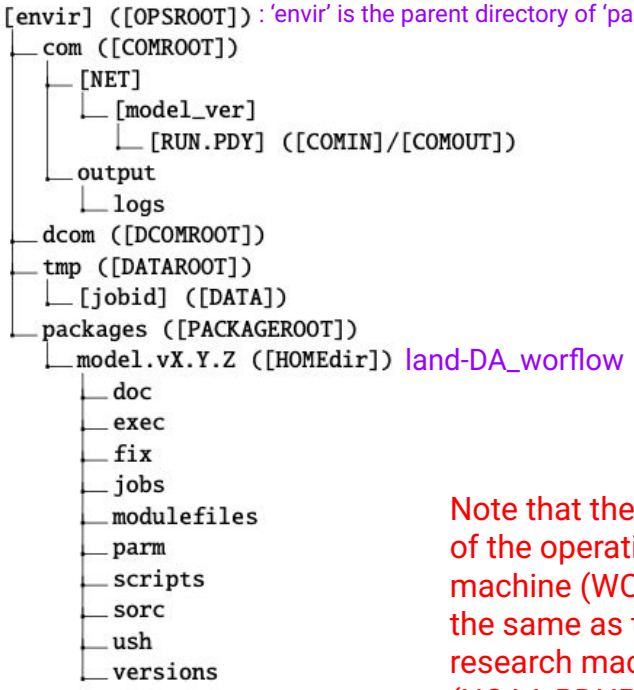
- According to the NCO standards:
  - The job card (submission script) is to set job scheduler directives and execution environment.
  - Each job (task) is associated with a single J-job.
  - The J-job sets up the environment and calls an ex-script.
  - Any sub-scripts to the ex-script will be located in the 'ush' directory.
- The UFS land-DA workflow uses Rocoto as a workflow manager, but ecFlow is **not** available yet.
- In Rocoto, job cards are not necessary, but a Rocoto XML file is required.



Vertical Directory Structure of Workflow

UFS land-DA workflow





NCO standards

Note that the structure of the operational machine (WCOSS) is not the same as that of the research machines (NOAA RDHPCS).

https://epic.noaa.gov/

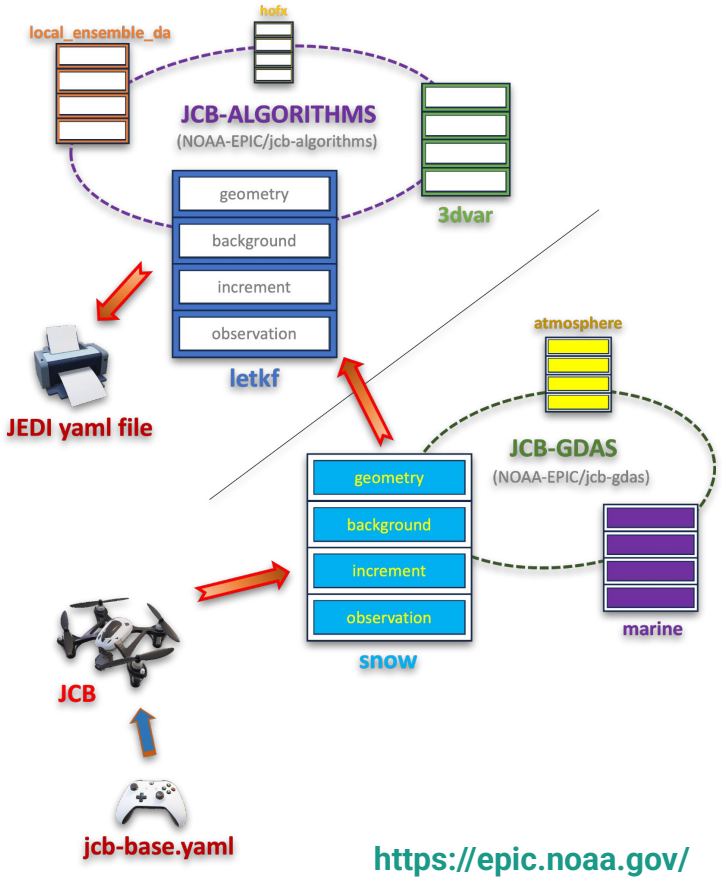
Workflow Tasks

- Tasks of UFS land-DA workflow:

Task name	Description	Application
JCB	Generate JEDI configuration YAML file	LND / ATML
PREP_DATA	Prepare observation / DATM forcing data files	LND / ATML
PRE_ANAL	Transfer snow depth data from restart files to surface data files	LND
ANALYSIS	Run JEDI and add increment to surface data files	LND / ATML
POST_ANAL	Transfer snow depth data from surface data files to restart files	LND / ATML
FORECAST	Run forecast model	LND / ATML
PLOT_STATS	Plot results of ANALYSIS and FORECAST	LND / ATML
FCST_IC	Generate initial condition (IC) files only for APP = ATML	ATML

Task: JCB (JEDI Configuration Builder)

- Generate a JEDI input YAML file.
- Components
  - **JCB**: python package
  - **JCB-algorithms**: collection of JEDI DA algorithms
    - 3dvar (for 3D-Var)
    - letkf (for LETKF)
    - etc.
  - **JCB-gdas**: items for each analysis
    - Snow
    - Marine
    - Atmosphere

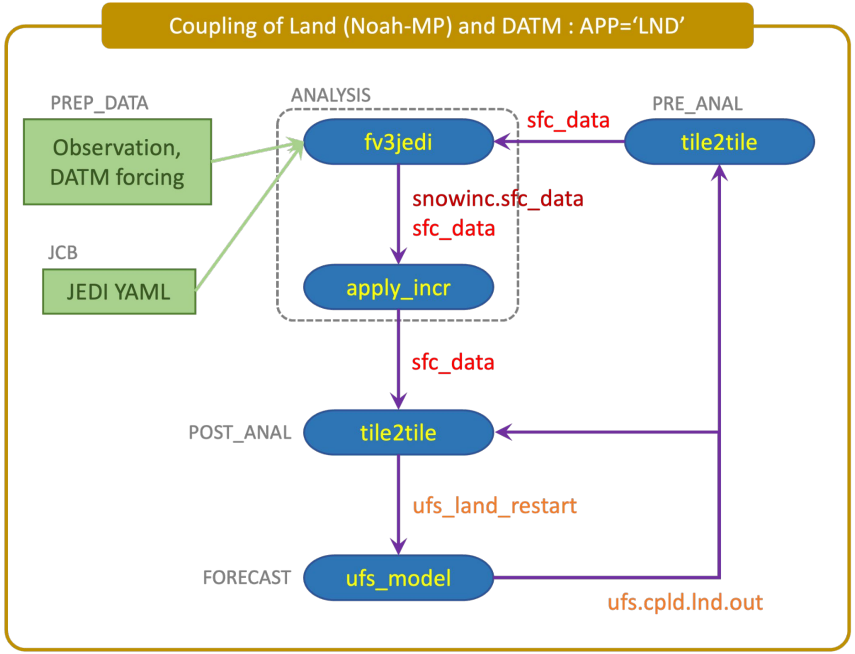


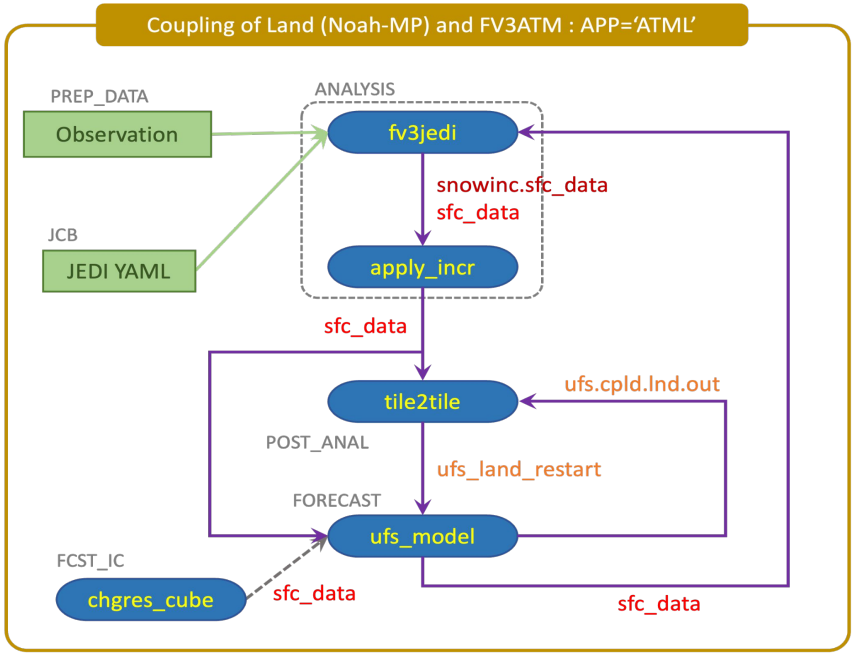
DATM Forcing Data / Snow Observation Data in 'PREP_DATA'

- DATM forcing data in case of APP=**LND** (for UFS weather model)
  - **ERA5** (**E**CMWF **R**e-**A**nalysis v**5**): can be downloaded from Climate Data Store (CDS).
    (European Centre for Medium-range Weather Forecasts)
  - **GSWP3** (**G**lobal **S**oil **W**etness **P**roject phase **3**): Available only for 1901-2010.
- Snow observation data (for JEDI; available in JCB)
  - **GHCN** (**G**lobal **H**istorical **C**limatology **N**etwork): can be created from the fix files within the UFS land-DA workflow.
  - **IMS** (**I**nteractive **M**ultisensor **S**now and **I**ce **M**apping **S**ystem): The raw data ASCII files can be obtained from GDAS/GFS (NOAA/EMC).
  - **SFCSNO** (Global Telecommunication System data): can be obtained from GDAS/GFS.
- Will be discussed in detail on Day 3 (pre/post-processing)

Flow of Data/Restart Files by Workflow Tasks

- The surface data ('sfc_data') and restart files play an important role in running both the UFS weather model and JEDI over cycles.





Note that the workflow manager 'Rocoto' will manage all tasks automatically.

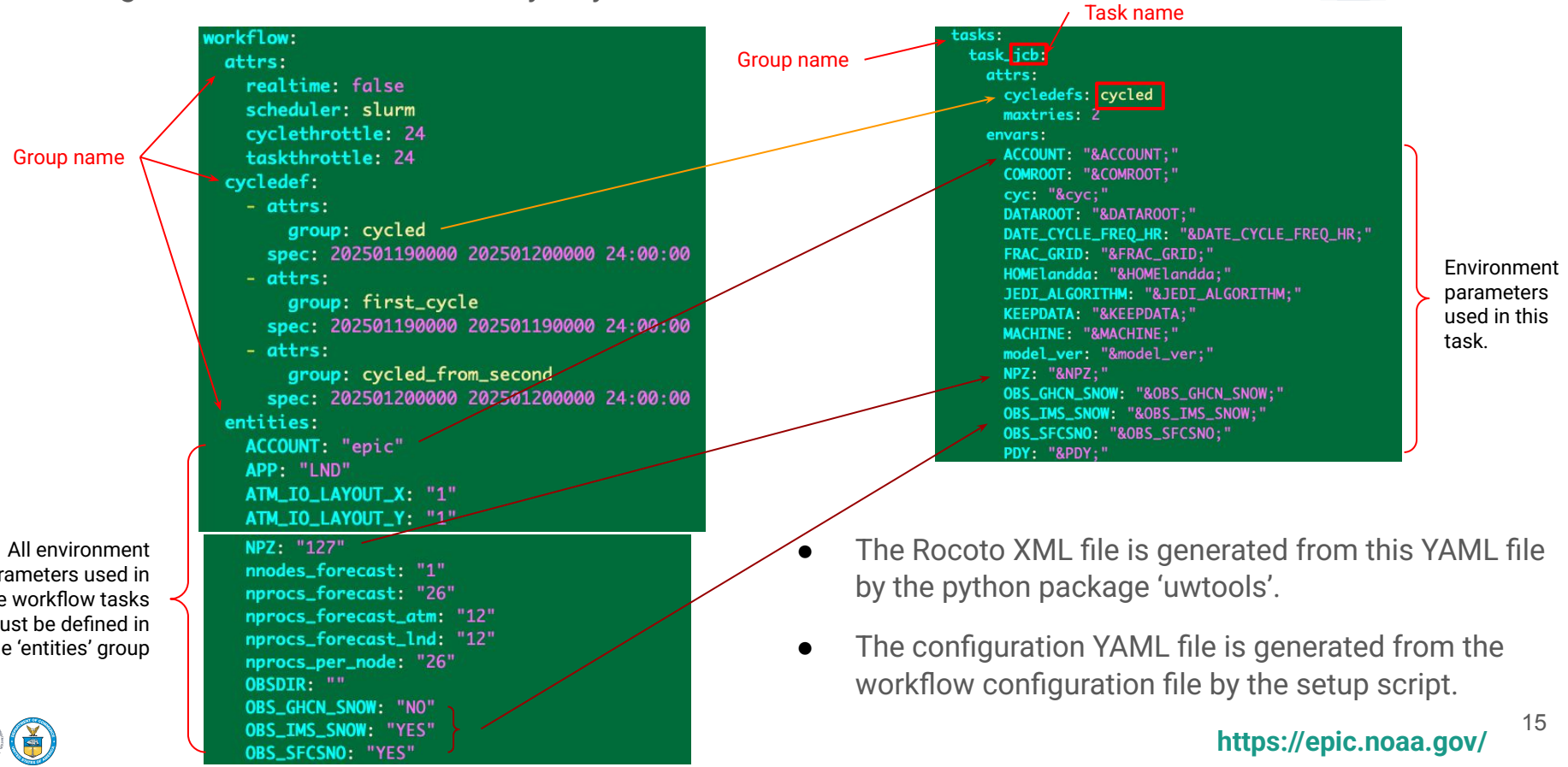
https://epic.noaa.gov/

Workflow Manager: Rocoto

- Rocoto is a workflow management tool developed by NOAA/GSL.
- The workflow environment variables, tasks, and their dependencies are defined in the Rocoto XML file ‘land_analysis.xml’.
- Rocoto submits the workflow tasks when their dependencies are satisfied and tracks the progress of the workflow tasks.
- The template of the Rocoto XML file is in the ‘parm/templates’ directory.
- In the UFS land-DA workflow, the Rocoto run/stat commands can be launch by the bash script created in the experimental case directory by the setup script.
- When ‘cron’ is available on a machine, the submission can be automated by the launch script.

Rocoto XML File: Configuration

- Configuration YAML file 'land_analysis.yaml' for Rocoto XML file

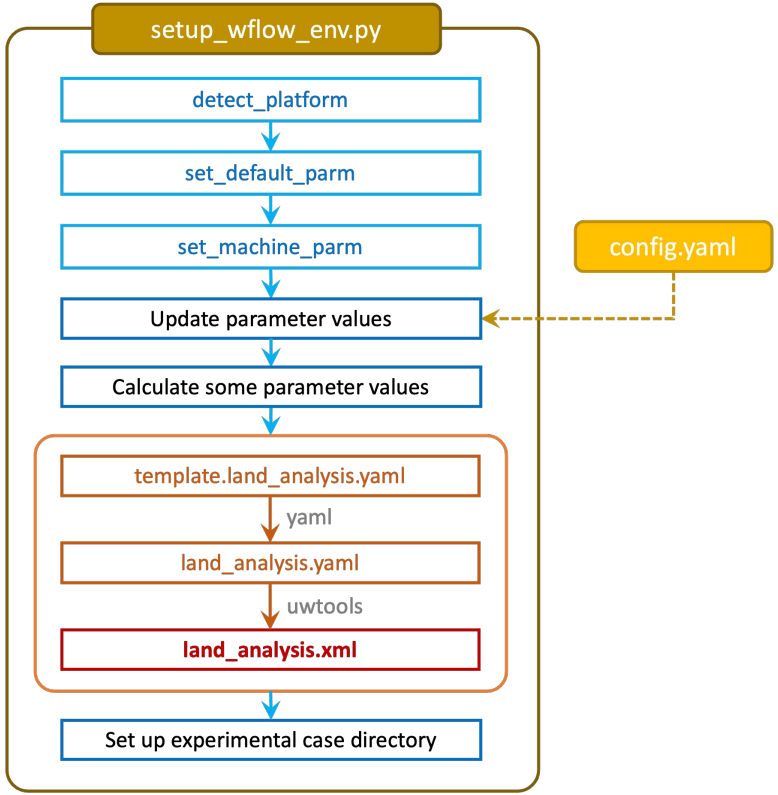


- The Rocoto XML file is generated from this YAML file by the python package 'uwtools'.
- The configuration YAML file is generated from the workflow configuration file by the setup script.

https://epic.noaa.gov/

Configuration of Workflow

- The XML file of the workflow manager ‘Rocoto’ can be generated by the setup python script ‘setup_wflow_env.py’ in ‘land-DA_workflow/parm’.
- All parameters, which are necessary for the Rocoto XML file and workflow scripts, are defined in this script with the default values.
- The default values pre-defined in the ‘setup_wflow_env.py’ script are replaced with those in the configuration file ‘config.yaml’. This means that users do not have to modify the setup script for their experiments. They can add any parameters they want to change to the ‘config.yaml’ file.
- This setup script finally generates a Rocoto XML file using the python package ‘uwtools’ within a new experimental case directory.



Configuration Script ‘config.yaml’

```
ACCOUNT: epic
APP: LND
ATMOS_FORC: era5
COLDSTART: 'NO'
COUPLER_CALENDAR: 2
DATE_CYCLE_FREQ_HR: 24
DATE_FIRST_CYCLE: 2025011900
DATE_LAST_CYCLE: 2025012500
DT_ATMOS: 900
DT_RUNSEQ: 3600
envir: test_cadre1
EXP_CASE_NAME: cadre1_lnd_era5_ims
FCSTHR: 24
JEDI_ALGORITHM: 3dvar
LND_CALC_SNET: .true.
LND_IC_TYPE: custom
LND_INITIAL_ALBEDO: 0.25
LND_LAYOUT_X: 1
LND_LAYOUT_Y: 2
LND_OUTPUT_FREQ_SEC: 21600
MED_COUPLING_MODE: ufs.nfrac.aoflux
model_ver: v2.1.0
net: landda
NPROCS_ANALYSIS: 6
NPZ: 127
OBS_GHCN_SNOW: 'NO'
OBS_IMS_SNOW: 'YES'
OBS_SFCSNO: 'YES'
RES: 96
RUN: landda
WE2E_TEST: 'NO'
```

HPC project account name

Application (coupling) option: LND / ATML

DATM forcing data option: era5 / gswp3

Flag for cold-start: ‘YES’ / ‘NO’

Cycle frequency in hours

First date of cycles

Last date of cycles

Experimental case name

Forecast length in hours for each cycle

JEDI algorithm: 3dvar / letkf

Parameters of land model (Noah-MP) in UFS weather model

Number of vertical layers (+1) in forecast (UFS weather model)

Flag for GHCN observation: ‘YES’ / ‘NO’

Flag for IMS observation: ‘YES’ / ‘NO’

Flag for SFCSNO observation: ‘YES’ / ‘NO’

Grid resolution per tile (C96=1 degree≈100km)

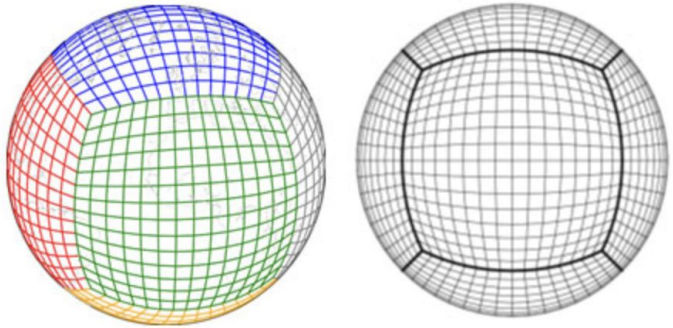
Sample Configurations

- Sample configuration files for CADRE DA-training:
  - land-DA_workflow/parm/config_samples/samples_cadre/

Filename prefix	App	DATM forcing	JEDI algorithm	Observation	Cold/Warm start	Number of cycles	Note
cadre0	LND	ERA5	LETKF	GHCN	Cold start	2	To prepare restart files for CADRE1
cadre1	**LND**	**ERA5**	**3D-Var**	**IMS + SFCSNO**	**Warm start**	**7**	**Benchmark case**
cadre2	**LND**	**GSWP3**	**LETKF**	**GHCN**	**Warm start**	**2**	**Another forcing and observation options**
cadre3	**ATML**	**N/A**	**3D-Var**	**GHCN**	**Cold start**	**2**	**Option 2: Noah-MP + FV3ATM**
cadre4	LND	ERA5	3D-Var	GHCN	Warm start	7	Same as CADRE1 but for **GHCN** stand-alone
cadre5	LND	ERA5	3D-Var	IMS	Warm start	7	Same as CADRE1 but for **IMS** stand-alone

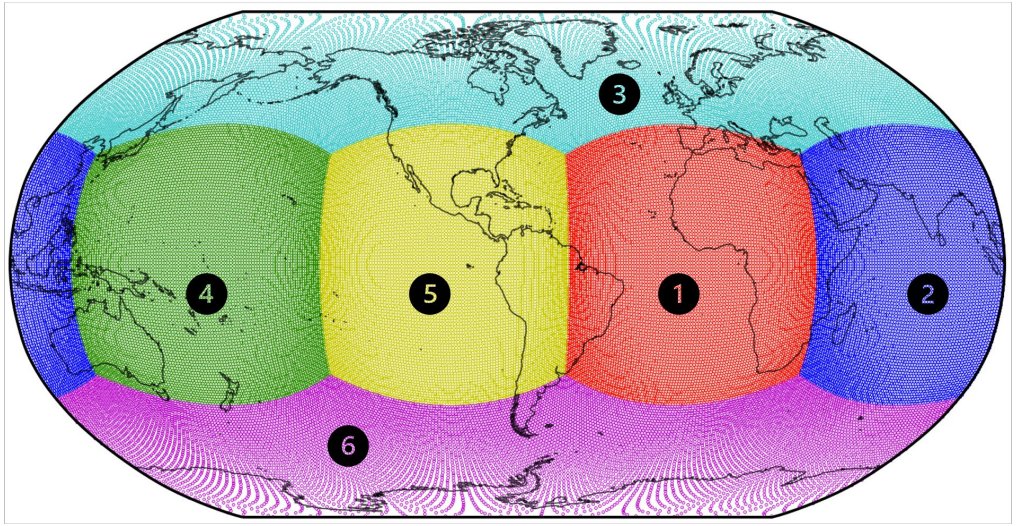
Six Tiles in Input / Output

- Some input / output files such as ‘sfc_data’ and ‘restart’ files have six tiles. This is because the FV3 dynamical core of the UFS weather model uses the cubed-sphere grid that represents the globe with six tiles.



Cubed-sphere grid of FV3 (NOAA/GFDL)

(gfdl.noaa.gov/fv3/fv3-grids/)



How to Check Log Files

ptmp/{envir}/com

- The log files of the workflow tasks can be found in 'com/output/logs/'.
- Some log files generated by executables are in the work directory of each workflow task under the 'tmp' directory:

task name cycle date job id (queue number)

```
[Chan-hoo.Jeon@hfe10 tmp_dir]$ ls
analysis.2000020200.10255800  forecast.2000020200.10256023  jcb.2000020300.10255309
analysis.2000020300.10256549  forecast.2000020300.10256983  plot_stats.2000020200.10256289
DATA_SHARE                    jcb.2000020200.10255306       plot_stats.2000020300.10257011
```

- For users' convenience, the symbolic links to the above log/work directories are provided in the experimental case directory:

symlink to LOG directory

```
[Chan-hoo.Jeon@hfe10 cadre1_lnd_era5_ims]$ ls
com_dir          land_analysis_lock.db  land_analysis.yaml      log_dir          log.rocoto_run
land_analysis.db land_analysis.xml      launch_rocoto_wflow.sh  log.rocoto_launch tmp_dir
```

symlink to COM (input/output) directory

symlink to WORK directory

Status of Workflow Tasks in Log File

- APP='**LND**'; warm-start

Log file: log.rocoto_launch

CYCLE	TASK	JOBID	STATE	EXIT STATUS	TRIES	DURATION
202501190000	jcb	10256999	SUCCEEDED	0	1	11.0
202501190000	prep_data	10256998	SUCCEEDED	0	1	66.0
202501190000	pre_anal	10257000	SUCCEEDED	0	1	13.0
202501190000	analysis	10257028	SUCCEEDED	0	1	150.0
202501190000	post_anal	10257121	SUCCEEDED	0	1	10.0
202501190000	forecast	10257199	SUCCEEDED	0	1	106.0
202501190000	plot_stats	10257229	SUCCEEDED	0	1	90.0
202501200000	jcb	10257001	SUCCEEDED	0	1	11.0
202501200000	prep_data	10257226	SUCCEEDED	0	1	35.0
202501200000	pre_anal	10257222	SUCCEEDED	0	1	6.0
202501200000	analysis	10257408	SUCCEEDED	0	1	105.0
202501200000	post_anal	10257507	SUCCEEDED	0	1	9.0
202501200000	forecast	10257523	SUCCEEDED	0	1	104.0
202501200000	plot_stats	10257686	SUCCEEDED	0	1	83.0

- APP='**ATML**'; cold-start

CYCLE	TASK	JOBID	STATE	EXIT STATUS	TRIES	DURATION
202212210000	prep_data	10263602	SUCCEEDED	0	1	15.0
202212210000	fcst_ic	10263603	SUCCEEDED	0	1	110.0
202212210000	forecast	10263819	SUCCEEDED	0	1	561.0
202212220000	jcb	10263604	SUCCEEDED	0	1	10.0
202212220000	prep_data	10263605	SUCCEEDED	0	1	307.0
202212220000	analysis	10264158	SUCCEEDED	0	1	139.0
202212220000	post_anal	10264305	SUCCEEDED	0	1	9.0
202212220000	forecast	10264306	SUCCEEDED	0	1	558.0
202212220000	plot_stats	10264610	SUCCEEDED	0	1	70.0