Steps to Run the UFS Short-Range Weather App v1.0.1

Julie Schramm[1,2], Jeff Beck[1,3,5], Laurie Carson[1,2], Mike Kavulich[1,2], Gerard Ketefian[1,3,4], Linlin Pan[1,3,4] , Jamie Wolff[1,2]

[1]NCAR Research Applications Laboratory
[2] Developmental Testbed Center

[3] NOAA/ESRL Global Systems Laboratory
[4] University of Colorado Cooperative Institute for Research in Environmental Sciences

[5]Cooperative Institute for Research in the Atmosphere

1st UFS SRW App Training Sept 20, 2021

SRW App Documentation

- Documentation for the UFS Short-Range Weather (SRW) App can be found at:

  https://ufs-srweather-app.readthedocs.io/en/ufs-v1.0.1/

- These slides follow the steps in the "Generate the Workflow Experiment" section:

  https://ufs-srweather-app.readthedocs.io/en/ufs-v1.0.1/Quickstart.html#generate-the-workflow-experiment

  and "Run the Workflow" section:

  https://ufs-srweather-app.readthedocs.io/en/ufs-v1.0.1/Quickstart.html#run-the-workflow-using-rocoto

SRW App uses Rocoto

- Rocoto is a workflow management system that interfaces with the batch system on an HPC:
  - Runs and manages dependencies between the tasks
  - Submits jobs to the HPC batch system as the task dependencies allow
  - Queries the batch system to get the current state of the workflow
  - Automatically resubmits failed tasks up to a given number of tries
  - Provides tools to check the status of the workflow
- Important! Executing an entire workflow is iterative: the `rocotorun` command must be executed many times to run a workflow to completion.

Description of SRW App Tasks

Workflow task	**Task description**
make_grid	Pre-processing task to generate regional grid files.
make_orog	Pre-processing task to generate orography files.
make_sfc_climo	Pre-processing task to generate surface climatology files.
get_extrn_ics	Cycle-specific task to obtain external data for the initial conditions
get_extrn_lbcs	Cycle-specific task to obtain external data for the lateral boundary conditions
make_ics	Generate initial conditions from the external data
make_lbcs	Generate lateral boundary conditions from the external data
run_fcst	Run the forecast model (UFS weather model)
run_post	Run the post-processing tool (UPP)

Workflow Generation Overview

Generating a regional workflow experiment requires three steps:

1. Set experiment parameters in **`config.sh`**
2. Set Python and other environment parameters
3. Run the **`generate_FV3LAM_wflow.sh`** script

These steps can be done using scripts available with the regional workflow, which uses the Rocoto workflow manager.

Step 1: Set parameters in **`config.sh`**

We will start with a copy of the template **`config.community.sh`**

```
cd ufs-srweather-app/regional_workflow/ush
cp config.community.sh config.sh
```

`config.sh` contains parameters that can be set by the user:

```
MACHINE="cheyenne"
ACCOUNT="my_account"
EXPT_SUBDIR="my_expt_name"
PREDEF_GRID_NAME="RRFS_CONUS_25km"
CCPP_PHYS_SUITE="FV3_GFS_v15p2"
FCST_LEN_HRS="12"
LBC_SPEC_INTVL_HRS="6"
```

Step 2: Set Python Environment

- Generating the workflow requires the appropriate Python environment
- This includes packages **PyYAML**, **Jinja2**, and **f90nml**
- These packages can be activated using the following script:

```
cd ufs-srweather-app/env
source ./wflow_<platform>.env
```

On Cheyenne, you will see:

```
Now using NPL virtual environment at path:
    /glade/p/ral/jntp/UFS_SRW_app/ncar_pylib/regional_workflow

Use deactivate to remove NPL from environment
```

Step 3: Generate the workflow

- Simply run the script:

```
cd ufs-srweather-app/regional_workflow/ush
./generate_FV3LAM_wflow.sh
```

- There will be output to your screen, ending with:

```
To launch the workflow, change location to the experiment directory
(EXPTDIR) and issue the rocotrun command, as follows:
  > cd /glade/scratch/$USER/expt_dirs/test_community
  > rocotorun -w FV3LAM_wflow.xml -d FV3LAM_wflow.db -v 10
To check on the status of the workflow, issue the rocotostat command
(also from the experiment directory):
  > rocotostat -w FV3LAM_wflow.xml -d FV3LAM_wflow.db -v 10
*/3 * * * * cd /glade/scratch/$USER/expt_dirs/test_community && ./launch_FV3LAM_wflow.sh
Done.
```

Developmental Testbed Center

Step 4: Run the Workflow

- The workflow is usually run using Rocoto
- After generating the workflow, there will be a directory called **`/glade/scratch/$USER/expt_dirs/expt_name`**
- From this directory **`$EXPTDIR`**, you can launch the workflow:

```
cd $EXPTDIR
./launch_FV3LAM_wflow.sh
```

- Once the workflow is launched, a file named **`log.launch_FV3LAM_wflow`** will be created in **`$EXPTDIR`**

Step 4: Run the Workflow (cont.)

- What is happening?
- **`launch_FV3LAM_wflow.sh`** has submitted the initial tasks to the queue and is monitoring the workflow

```
CYCLE                 TASK         JOBID                STATE     EXIT STATUS   TRIES   DURATION
=================================================================================================
201906150000       make_grid   druby://hfe01:45920   SUBMITTING      -           0         0.0
201906150000       make_orog   -                     -               -           -         -
201906150000  make_sfc_climo   -                     -               -           -         -
201906150000   get_extrn_ics   druby://hfe01:45920   SUBMITTING      -           0         0.0
201906150000  get_extrn_lbcs   druby://hfe01:45920   SUBMITTING      -           0         0.0
201906150000        make_ics   -                     -               -           -         -
201906150000       make_lbcs   -                     -               -           -         -
201906150000        run_fcst   -                     -               -           -         -
201906150000   run_post_f000   -                     -               -           -         -
...
```

Developmental Testbed Center

Step 4: Run the Workflow (cont.)

- **`launch_FV3LAM_wflow.sh`** runs one instance of the **`rocotorun`** and **`rocotostat`** commands:

```
rocotorun -w FV3LAM_wflow.xml -d FV3LAM_wflow.db -v 10
```

```
rocotostat -w FV3LAM_wflow.xml -d FV3LAM_wflow.db -v 10
```

- Until the launch script is run again, the workflow will remain in this state.
- There are two options to continue running the workflow:
  - Manually run **`launch_FV3LAM_wflow.sh`** until the workflow completes
  - Run the workflow via a crontab

Step 4: Run the Workflow (cont.)

- Manually run the **`launch_FV3LAM_wflow.sh`** one more time:

```
CYCLE                   TASK     JOBID     STATE    EXIT STATUS   TRIES   DURATION
================================================================================
201906150000       make_grid  21337987   QUEUED          -           0        0.0
201906150000       make_orog  -          -               -           -          -
201906150000   make_sfc_climo -          -               -           -          -
201906150000    get_extrn_ics 21337988   SUCCEEDED  0                1        9.0
201906150000   get_extrn_lbcs 21337989   SUCCEEDED  0                1        9.0
```

Step 4: Run the Workflow (cont.)

- The second option is to run the **launch_FV3LAM_wflow.sh** via your crontab (every 3 minutes):

```
crontab -e
*/3 * * * * cd /glade/scratch/$USER/expt_dirs/expt_name &&
./launch_FV3LAM_wflow.sh
```

- You can then monitor the workflow progress until all tasks are complete:

```
rocotostat -w FV3LAM_wflow.xml -d FV3LAM_wflow.db -v 10
```

```
CYCLE               TASK                JOBID          STATE        EXIT STATUS TRIES    DURATION
=================================================================================================
201906150000           make_grid   9458110      SUCCEEDED      0          1         30.0
201906150000           make_orog   9458169      SUCCEEDED      0          1         31.0
201906150000      make_sfc_climo   9458193      SUCCEEDED      0          1         45.0
201906150000       get_extrn_ics   9458111      SUCCEEDED      0          1         11.0
201906150000      get_extrn_lbcs   9458112      SUCCEEDED      0          1         11.0
201906150000            make_ics   9458252      SUCCEEDED      0          1         23.0
201906150000           make_lbcs   9458253      SUCCEEDED      0          1         21.0
201906150000            run_fcst   9458285      SUCCEEDED      0          1       1854.0
201906150000       run_post_f000   9458391      SUCCEEDED      0          1         15.0
201906150000       run_post_f001   9458610      SUCCEEDED      0          1         12.0
201906150000       run_post_f002   9458729      SUCCEEDED      0          1         16.0
201906150000       run_post_f003   9458913      SUCCEEDED      0          1         11.0

 . . .
```

Handy Tip

- If you notice the workflow is stuck in one state after modifying your crontab, but it advances when you manually run **`launch_FV3LAM_wflow.sh`**, there may be a typo in your crontab.
- Check path and experiment name.

```
CYCLE                   TASK    JOBID    STATE   EXIT STATUS   TRIES    DURATION
================================================================================
201906150000        make_grid  21337987   QUEUED      -         0           0.0
201906150000        make_orog  -          -           -         -             -
201906150000   make_sfc_climo  -          -           -         -             -
201906150000    get_extrn_ics  21337988   QUEUED      -         0           0.0
201906150000   get_extrn_lbcs  21337989   QUEUED      -         0           0.0
```

Additional Documentation

- Rocoto documentation: https://github.com/christopherwharrop/rocoto/wiki/documentation
- UFS UTILS: https://ufs-utils.readthedocs.io/en/latest/
- UFS Weather model: https://ufs-weather-model.readthedocs.io/en/ufs-v2.0.0
- Unified Post Processor (UPP): https://upp.readthedocs.io/en/upp-v9.0.0/

That’s it!

- Questions?