Get Logged In

EPIC has set up 50 head-node instances in the Amazon cloud.

Every participant will log into *one* of the instances (we don't want people stepping on each other)

Everyone will use the same PEM file available – HERE and log into their head node as user "ubuntu"

The list of head node addresses next to each participant name is – HERE

The command to log into your head node is "ssh -i path-to/EPIC_RSA.pem ubuntu@your-head-node-address

Please make sure to log into your own head node. Once you are in, if you type "who" you should see only your own login.

Overview

- Explain how the SRW workflow operates
- Explain how the SRW workflow is configured
- Run 3 experiments
  - 6 hour 25 km CONUS forecast for 6/15/2019
  - 6 hour high resolution forecast over Indianapolis with v16 physics
  - 6 hour high resolution forecast over Indianapolis with RRFS physics
- Plot results and compare
- Describe the process for setting up a new Domain
- Describe how to build the SRW application (time permitting)

Objective: I can use the Short Range Weather application to run, modify, and compare forecast outputs.

Your AWS Environment

(# = comment, $ = command)

 # Start from your home directory (/home/ubuntu), which is pre-populated with a copy of the ufs-srweather-app repo

#cd to the ush directory

$ cd ufs-srweather-app/regional_workflow/ush

#activate the regional workflow conda environment

$ conda activate regional_workflow

Your AWS Environment

(# = comment, $ = command)

The instance that you are logging into has been provisioned with GNU/OpenMPI-based modules and a pre-built beta-version of the SRW Application/release-v2 (soon to be released). Each of your home directories have a copy of the application in it.

Conda is a packaging application that the SRW uses to set up python environments with all the required modules used by the workflow.

In this case, the regional_workflow environment has everything needed to run the workflow and make plots upon completion.

This AWS instance is set up with slurm and has access to a single compute node with 72 MPI cores available. All steps of the workflow will run on the compute node.

 # Start from your home directory (/home/ubuntu), which is pre-populated with a copy of the ufs-srweather-app repo

#cd to the ush directory

$ cd ufs-srweather-app/regional_workflow/ush

#activate the regional workflow conda environment

$ conda activate regional_workflow

Step 1: Run the Control Case

We will be using a pre-configured 25KM resolution CONUS (Continental US) forecast to start. This is a re-forecast for the CONUS from June 15, 2019, a day with particularly interesting weather.

Copy the config.sh.lowres file to config.sh

$ cp config.sh.CONUS config.sh

Make sure you have activated your regional workflow environment. (should see regional_workflow on left of your prompt) We'll talk about what is in the configuration file in a minute, but let's generate the workflow and get it started before we do that

$ ./generate_FV3LAM_wflow.sh

This script creates an experiment directory and populates it with all the data needed to run through the workflow. In this case, the experiment has been configured to automatically add an entry to your crontab which will run through the entire workflow using rocoto. If you run "crontab -l" (lower case L) you will see the actual commands that are being run.

Cron will run that command every minute.

$ crontab -l

***/01 * * * * cd /home/ubuntu/expt_dirs/GST_CONUS && ./launch_FV3LAM_wflow.sh called_from_cron="TRUE" >> ./log.launch_FV3LAM_wflow 2>&1**

Review of the Workflow (single forecast cycle)

- Starting from initial and boundary conditions extracted from a global model
- make grid for the region of interest (CONUS to start)
- get external initial conditions from global model
- get external lateral boundary conditions from global model
- make the orography to be used by the regional forecast
- make the surface climatology to be used by regional forecast if external model fields unavailable
- make the initial conditions (convert global data to be used on regional grid)
- make the lateral boundary conditions (convert the global data to be used on regional grid)
- run the forecast
- perform post processing on forecast data (convert from model grid to grib2 format)

Scripts, Rocoto, and launch_FV3LAM_wflow

- Each step of the workflow can be run manually using scripts provided in ush/wrappers
- The generate_FV3LAM_wflow.sh script creates an XML file used by rocoto
- XML file contains all the information about how to run each of the steps, including submission to the batch queuing system (slurm, in this case)
- A local database file is used to keep track of the progress of the workflow
- launch_FV3LAM_wflow.sh is a wrapper script around rocoto that can interact with crontab among other things

Rocoto basics

- Usually available as a module, but can be built in user space
- Run from the experiment directory
- Check status of wflow
  rocotostat -w FV3LAM_wflow.xml -d FV3LAM_wflow.db -v 10
- Run next step(s) of the workflow
  rocotorun -w FV3LAM_wflow.xml -d FV3LAM_wflow.db -v 10
- Rewind step(s) of the workflow
  rocotorewind -w FV3LAM_wflow.xml -d FV3LAM_wflow.db -v 10 -c CYCLE -t TASK_NAME

Try running rocotostat from ~/expt_dirs/GST_CONUS

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

```
(regional_workflow) ubuntu@ip-10-0-8-43:~/expt_dirs/GST_CONUS$ rocotostat -w FV3LAM_wflow.xml -d FV3LAM_wflow.db -v 10
       CYCLE                    TASK                       JOBID               STATE         EXIT STATUS     TRIES      DURATION
================================================================================================================================
201906151800               make_grid                          68           SUCCEEDED                   0         1           4.0
201906151800               make_orog                          71           SUCCEEDED                   0         1         361.0
201906151800          make_sfc_climo                          72           SUCCEEDED                   0         1         400.0
201906151800           get_extrn_ics                          69           SUCCEEDED                   0         1          24.0
201906151800          get_extrn_lbcs                          70           SUCCEEDED                   0         1          50.0
201906151800                make_ics                          73           SUCCEEDED                   0         1          39.0
201906151800               make_lbcs                          74           SUCCEEDED                   0         1          33.0
201906151800                run_fcst                          75             RUNNING                   -         0           0.0
201906151800           run_post_f000                          76              QUEUED                   -         0           0.0
201906151800           run_post_f001                           -                   -                   -         -             -
201906151800           run_post_f002                           -                   -                   -         -             -
201906151800           run_post_f003                           -                   -                   -         -             -
201906151800           run_post_f004                           -                   -                   -         -             -
201906151800           run_post_f005                           -                   -                   -         -             -
201906151800           run_post_f006                           -                   -                   -         -             -
```

Check on status of compute jobs using squeue

```
(regional_workflow) ubuntu@ip-10-0-8-43:~/ufs-srweather-app/regional_workflow/ush$ squeue
             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
                83       srw make_gri   ubuntu CF       0:24      1 srw-dy-c5n-1
                84       srw get_extr   ubuntu CF       0:24      1 srw-dy-c5n-1
                85       srw get_extr   ubuntu CF       0:24      1 srw-dy-c5n-1
(regional_workflow) ubuntu@ip-10-0-8-43:~/ufs-srweather-app/regional_workflow/ush$
```

Anyone not seeing jobs running or completed?

If you are having trouble, use the # slack channel and someone will work though the issue with you.

Let’s look at the configuration of what we are running

Open up config.sh with your favorite editor

Note that these variables override those in config_defaults.sh

```
#
# The machine on which to run, the account to which to charge computational
# resources, the base directory in which to create the experiment directory
# (if different from the default location), and the name of the experiment
# subdirectory.
#
MACHINE="AMI"
ACCOUNT="none"
EXPT_SUBDIR="GST_lowres"
EXPT_BASEDIR="$HOME/expt_dirs"

NNODES_MAKE_ICS="1"
NNODES_MAKE_LBCS="1"
NNODES_RUN_POST="1"
NNODES_MAKE_SFC_CLIMO="1"
OMP_NUM_THREADS_RUN_FCST="1"
OMP_STACKSIZE_RUN_FCST="1024m"
PPN_MAKE_ICS="2"
PPN_MAKE_LBCS="2"
PPN_RUN_POST="2"

RUN_ENVIR="community"
PREEXISTING_DIR_METHOD="rename"



# Flag specifying whether or not to automatically resubmit the worfklow
# to the batch system via cron and, if so, the frequency (in minutes) of
# resubmission.
#
USE_CRON_TO_RELAUNCH="TRUE"
CRON_RELAUNCH_INTVL_MNTS="01"
#
# Flag specifying whether to run in verbose mode.
#
```

Experiment name and location + platform basics

```
#
# The machine on which to run, the account to which to charge computational
# resources, the base directory in which to create the experiment directory
# (if different from the default location), and the name of the experiment
# subdirectory.
#
MACHINE="AMI"
ACCOUNT="none"
COMPILER="gnu"
EXPT_SUBDIR="GST_CONUS"
EXPT_BASEDIR="/home/$USER/expt_dirs"
```

Define the computational parameters for workflow steps

```
NNODES_MAKE_ICS="1"
NNODES_MAKE_LBCS="1"
NNODES_RUN_POST="1"
NNODES_MAKE_SFC_CLIMO="1"
OMP_NUM_THREADS_RUN_FCST="1"
OMP_STACKSIZE_RUN_FCST="1024m"
PPN_MAKE_ICS="16"
PPN_MAKE_LBCS="16"
PPN_RUN_POST="2"
PPN_MAKE_GRID="16"
PPN_MAKE_OROG="36"
PPN_MAKE_SFC_CLIMO="36"
PARTITION_DEFAULT=""
QUEUE_DEFAULT=""
```

Define how/how often workflow will be run and

```
# Flag specifying whether or not to automatically resubmit the worfklow
# to the batch system via cron and, if so, the frequency (in minutes) of
# resubmission.
#
USE_CRON_TO_RELAUNCH="TRUE"
CRON_RELAUNCH_INTVL_MNTS="01"
#
# Flag specifying whether to run in verbose mode.
#
VERBOSE="TRUE"
#
# TEST PURPOSE/DESCRIPTION:
# ------------------------
#
# Starting with a 25km pre-defined grid for the AMS 2022 SRW Workshop.
#

RUN_ENVIR="community"
PREEXISTING_DIR_METHOD="rename"
```

Specify a physics suite along with other details of the forecast to be run

```
CCPP_PHYS_SUITE="FV3_GFS_v16"

EXTRN_MDL_NAME_ICS="FV3GFS"
EXTRN_MDL_NAME_LBCS="FV3GFS"
USE_USER_STAGED_EXTRN_FILES="TRUE"

DATE_FIRST_CYCL="20190615"
DATE_LAST_CYCL="20190615"
CYCL_HRS=( "18" )

FV3GFS_FILE_FMT_ICS="grib2"
FV3GFS_FILE_FMT_LBCS="grib2"

FCST_LEN_HRS="6"
LBC_SPEC_INTVL_HRS="6"
WTIME_RUN_FCST="00:30:00"
#
```

Locations of input files and description of the domain to be used.

```
#
USE_USER_STAGED_EXTRN_FILES="TRUE"
EXTRN_MDL_SOURCE_BASEDIR_ICS="/contrib/GST/model_data/FV3GFS"
EXTRN_MDL_FILES_ICS=( "gfs.pgrb2.0p25.f000" )
EXTRN_MDL_SOURCE_BASEDIR_LBCS="/contrib/GST/model_data/FV3GFS"
EXTRN_MDL_FILES_LBCS=( "gfs.pgrb2.0p25.f006" "gfs.pgrb2.0p25.f012" )

# use predefined CONUS 25km grid
PREDEF_GRID_NAME="RRFS_CONUS_25km"
GRID_GEN_METHOD="ESGgrid"
QUILTING="TRUE"

# Set the layout of the domain decomposition
LAYOUT_X="${LAYOUT_X:-8}"
LAYOUT_Y="${LAYOUT_Y:-8}"
```

Define number of tries for each step.

```
# Set maximum number of retries in case of failure
MAXTRIES_MAKE_GRID="2"
MAXTRIES_MAKE_OROG="1"
MAXTRIES_MAKE_SFC_CLIMO="1"
MAXTRIES_GET_EXTRN_ICS="1"
MAXTRIES_GET_EXTRN_LBCS="1"
MAXTRIES_MAKE_ICS="2"
MAXTRIES_MAKE_LBCS="2"
MAXTRIES_RUN_FCST="1"
MAXTRIES_RUN_POST="2"
```

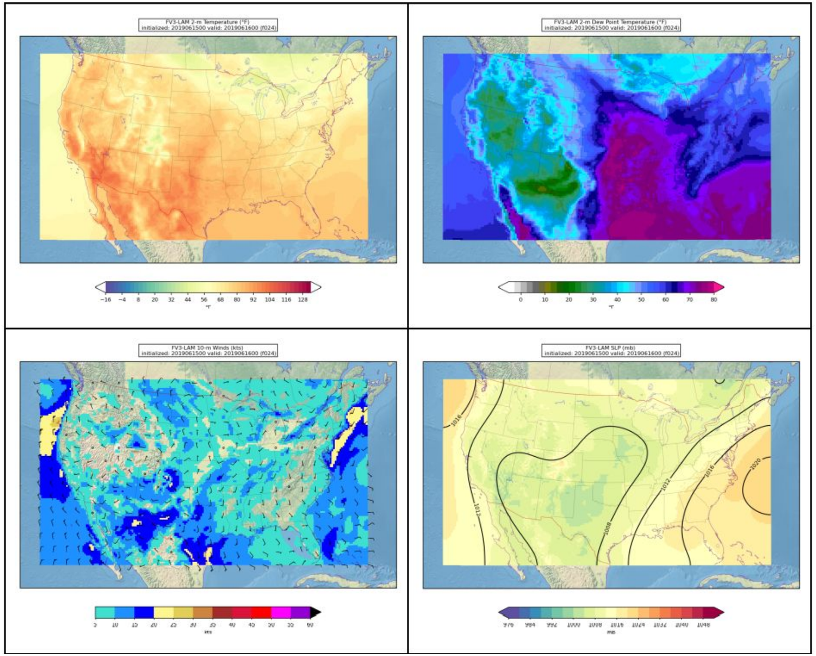
Plot the results!

When your experiment is complete, the subdirectory 2019061518/postprd will contain a series of grib2 files created by the UPP. We will use a plotting package that comes with the SRW to generate plots of those results.

#From your experiment directory run the following

$ export EXPTDIR=$PWD
$ export HOMErrfs=~/ufs-srweather-app/regional_workflow
$ HOMErrfs/ush/make_plots.sh

When the script completes there will be a series of PNG files in the postprd directory as well.



Objective: I can use the Short Range Weather application to **run**, modify, and compare forecast outputs.

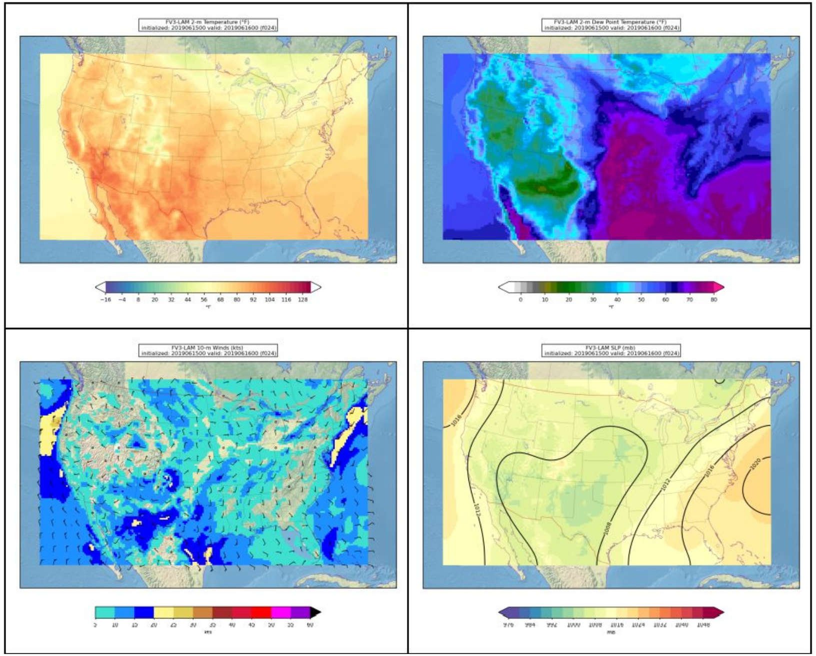
Download CONUS plots

Create a plots directory on your local machine and then create a conus subdirectory under that.

$ cd plots/conus

$ scp ubuntu@your-ip-address:"expt_dirs/GST_CONUS/20191518/postprd/*.png" .

Open with a browser or your favorite image viewer



Objective: I can use the Short Range Weather application to **run**, modify, and compare forecast outputs.

Workflow 2: Change the horizontal resolution & re-run the experiment over Indianapolis

#We can now go generate a new experiment that will re-run the same forecast, but at a higher resolution focusing on a small area around Indianapolis, which saw some extreme weather that day.

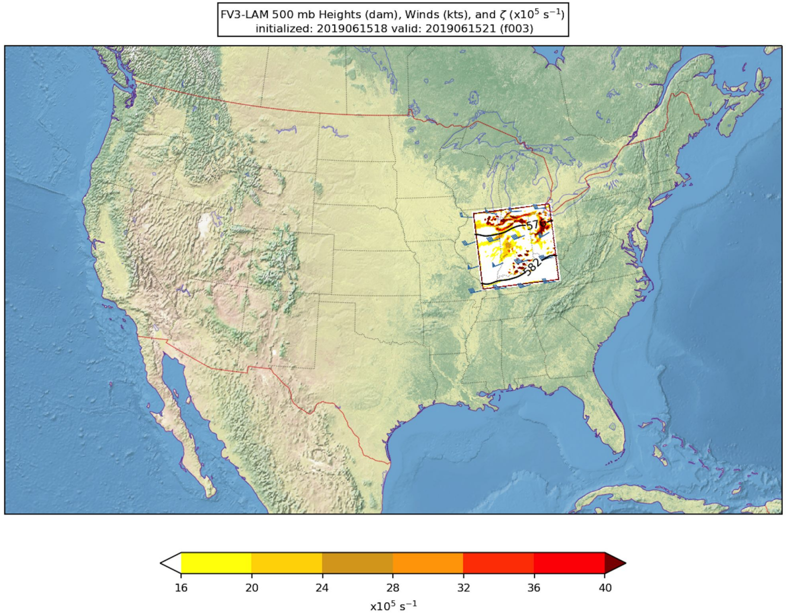
$ cd back to your ush directory

$ cd $HOME/ufs-srweather-app/regional_workflow/ush

#Now copy the config.sh.INDY.v16 to config.sh

$ cp config.sh.INDY.v16 config.sh

#Again, we will look at what this contains in a minute, but let's generate a new experiment and get it running



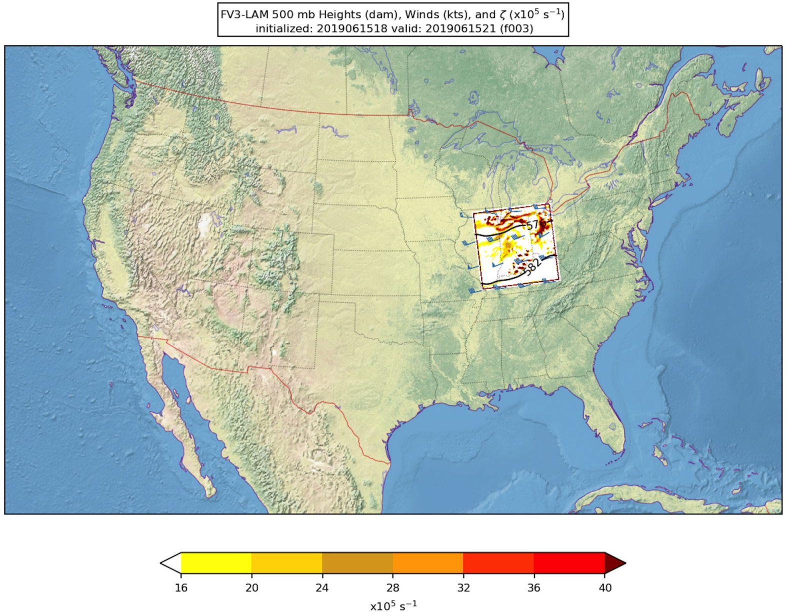
Workflow 2: Run the horizontal resolution change experiment

$ ./generate_FV3LAM_wflow.sh

This experiment is also set to add a slightly different entry to your crontab. You can check it out by listing the crontab contents.

$ crontab -l

Now we will look at the differences between the two configurations



Let's look at the configuration of what we are running

Open up the new config.sh with your favorite editor

This version will be somewhat different than the CONUS experiment

```
#
# The machine on which to run, the account to which to charge computational
# resources, the base directory in which to create the experiment directory
# (if different from the default location), and the name of the experiment
# subdirectory.
#
MACHINE="AMI"
ACCOUNT="none"
COMPILER="gnu"
EXPT_SUBDIR="GST_INDY_v16"
EXPT_BASEDIR="/home/$USER/expt_dirs"

NNODES_MAKE_OROG="1"
NNODES_MAKE_ICS="1"
NNODES_MAKE_LBCS="1"
NNODES_RUN_POST="1"
NNODES_MAKE_SFC_CLIMO="1"
OMP_NUM_THREADS_RUN_FCST="1"
OMP_STACKSIZE_RUN_FCST="1024m"
PPN_MAKE_ICS="36"
PPN_MAKE_LBCS="36"
PPN_RUN_POST="2"
```

We are now defining a custom grid centered over Indiana

```
# Define custom grid.
GRID_GEN_METHOD="ESGgrid"
QUILTING="TRUE"

ESGgrid_LON_CTR="-86.16"
ESGgrid_LAT_CTR="39.77"

ESGgrid_DELX="3000.0"
ESGgrid_DELY="3000.0"

ESGgrid_NX="200"
ESGgrid_NY="200"

ESGgrid_PAZI="0.0"
```

```
WRTCMP_write_groups="1"
WRTCMP_write_tasks_per_group=$(( 1*LAYOUT_Y ))
WRTCMP_output_grid="lambert_conformal"
WRTCMP_cen_lon="${ESGgrid_LON_CTR}"
WRTCMP_cen_lat="${ESGgrid_LAT_CTR}"
WRTCMP_stdlat1="${ESGgrid_LAT_CTR}"
WRTCMP_stdlat2="${ESGgrid_LAT_CTR}"
WRTCMP_nx="197"
WRTCMP_ny="197"
WRTCMP_lon_lwr_left="-89.47120417"
WRTCMP_lat_lwr_left="37.07809642"
WRTCMP_dx="${ESGgrid_DELX}"
WRTCMP_dy="${ESGgrid_DELY}"
```

Plot the results!

When your experiment is complete, the subdirectory 2019061518/postprd will again contain a series of grib2 files created by the UPP.

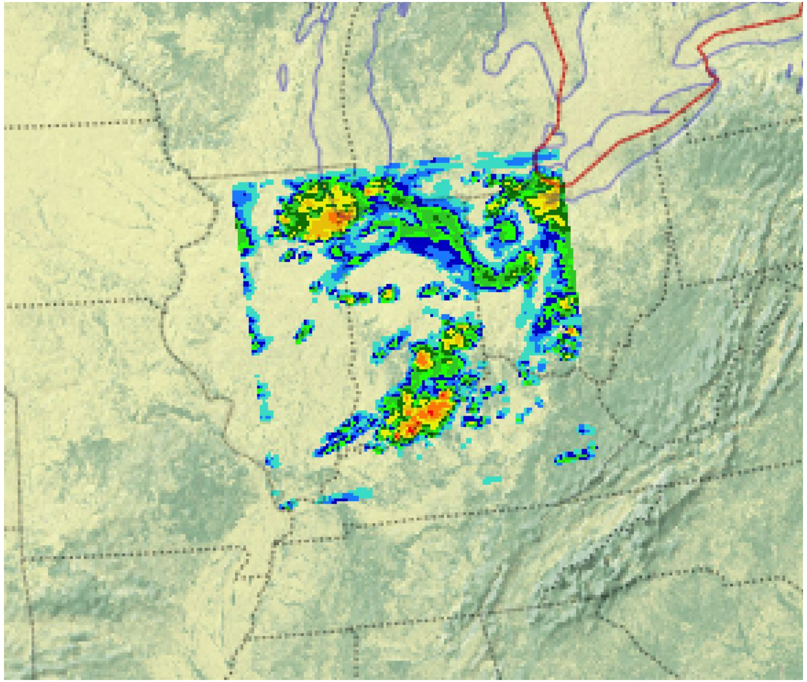
#From your experiment directory run the following

$ export EXPTDIR=$PWD

If you are using the same terminal, you can skip the next step
$ export HOMErrfs=~/ufs-srweather-app/regional_workflow

$ HOMErrfs/ush/make_plots.sh

When the script completes there will be a series of PNG files in the postprd directory as well.



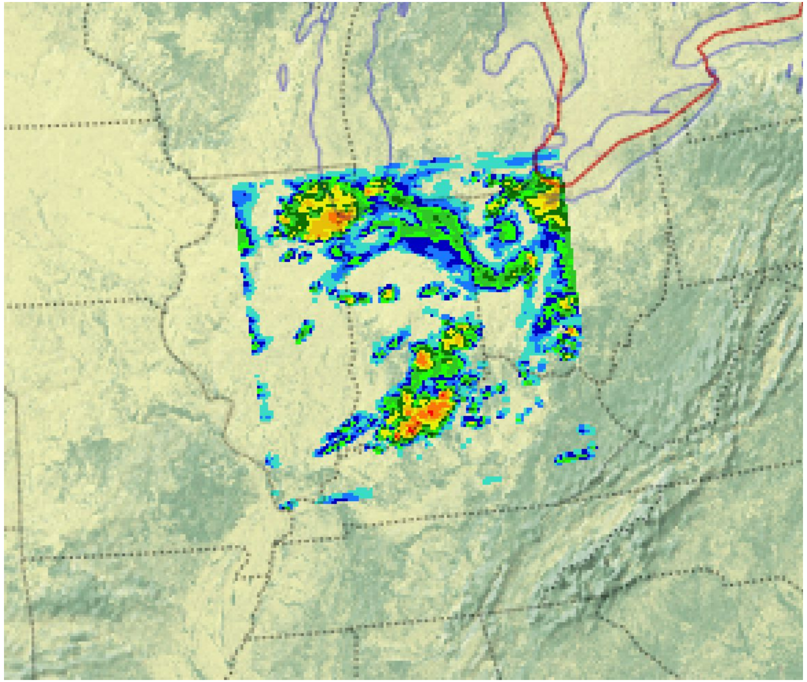
Download Indy v16 plots

Create an INDY-v16 subdirectory in your plots directory

$ cd plots/INDY-v16

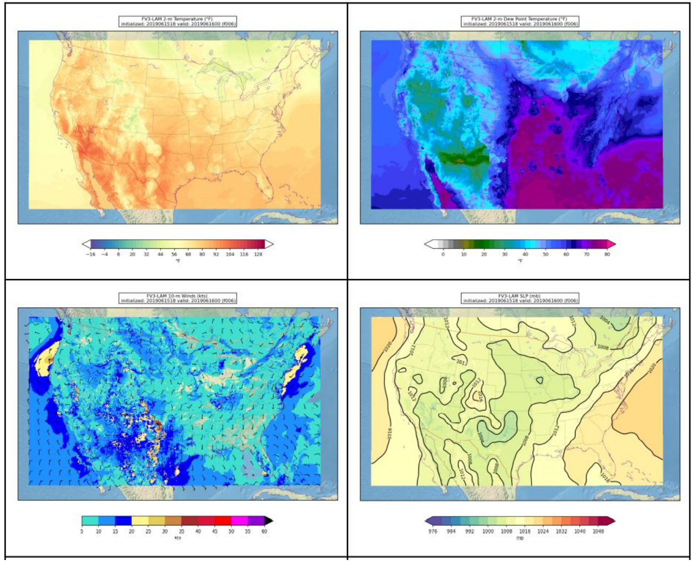
$ scp ubuntu@your-ip-address:"expt_dirs/GST_INDY_v16/20191518/postprd/*.png" .

Open with a browser or your favorite image viewer



Workflow 3: Change the physics suite & re-run the experiment over Indianapolis

#We can now take a look at the impact of physics upon the quality of the forecast. The UFS Weather Model used by the SRW Application can be configured to use a large assortment of physics suites. Let's try one that is particularly well suited to high resolution simulations where the convection no longer needs to be parameterized.



$ cd back to your ush directory

$ cd $HOME/ufs-srweather-app/regional_workflow/ush

#Now copy the config.sh.INDY.v1beta to config.sh

$ cp config.sh.INDY.v1beta config.sh

#Again, we will look at what this contains in a minute, but let's generate a new experiment and get it running

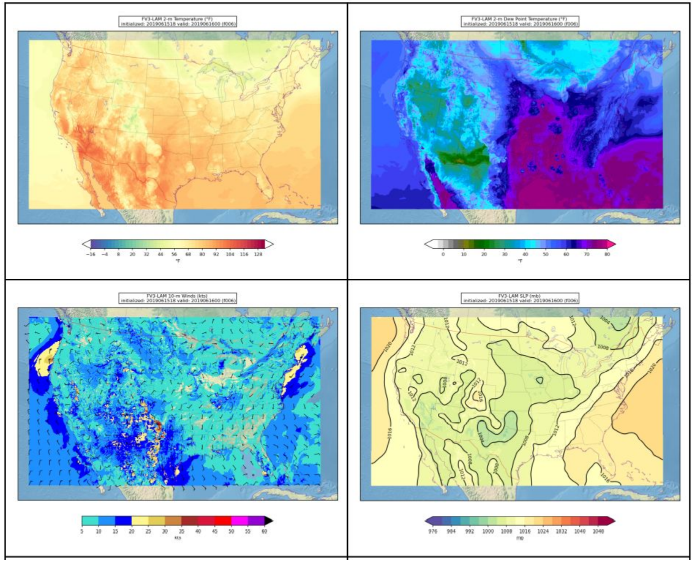
Workflow 3: Run the localized experiment with new physics suite

$ ./generate_FV3LAM_wflow.sh

This experiment is also set to add a slightly different entry to your crontab. You can check it out by listing the crontab contents.

$ crontab -l

Now we will look at the differences between the two configurations



Let's look at the configuration of what we are running

Open up the new config.sh with your favorite editor

This version will be slightly different than the first Indy experiment

```
#
# The machine on which to run, the account to which to charge computational
# resources, the base directory in which to create the experiment directory
# (if different from the default location), and the name of the experiment
# subdirectory.
#
MACHINE="AMI"
ACCOUNT="none"
COMPILER="gnu"
EXPT_SUBDIR="GST_INDY_v16"
EXPT_BASEDIR="/home/$USER/expt_dirs"

NNODES_MAKE_OROG="1"
NNODES_MAKE_ICS="1"
NNODES_MAKE_LBCS="1"
NNODES_RUN_POST="1"
NNODES_MAKE_SFC_CLIMO="1"
OMP_NUM_THREADS_RUN_FCST="1"
OMP_STACKSIZE_RUN_FCST="1024m"
PPN_MAKE_ICS="36"
PPN_MAKE_LBCS="36"
PPN_RUN_POST="2"
```

Some of the key differences

```
CCPP_PHYS_SUITE="FV3_GFS_v16"
POST_OUTPUT_DOMAIN_NAME="INDY"

EXTRN_MDL_NAME_ICS="HRRR"
EXTRN_MDL_NAME_LBCS="RAP"
USE_USER_STAGED_EXTRN_FILES="TRUE"
```

```
#
USE_USER_STAGED_EXTRN_FILES="TRUE"
EXTRN_MDL_SOURCE_BASEDIR_ICS="/contrib/GST/model_data/HRRR"
EXTRN_MDL_FILES_ICS=( "hrrr.wrfprsf00.grib2" )
EXTRN_MDL_SOURCE_BASEDIR_LBCS="/contrib/GST/model_data/RAP"
EXTRN_MDL_FILES_LBCS=( "rap.wrfprsf06.grib2" )
```

Plot the results!

When your experiment is complete, the subdirectory 2019061518/postprd will again contain a series of grib2 files created by the UPP.

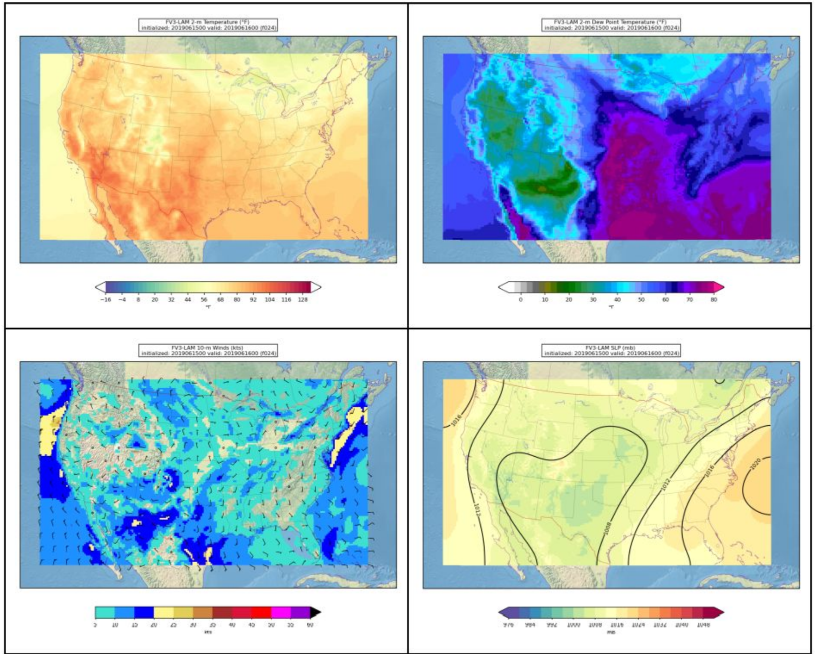
#From your experiment directory run the following

$ export EXPTDIR=$PWD

If you are using the same terminal, you can skip the next step
$ export HOMErrfs=~/ufs-srweather-app/regional_workflow

$ HOMErrfs/ush/make_plots.sh

When the script completes there will be a series of PNG files in the postprd directory as well.



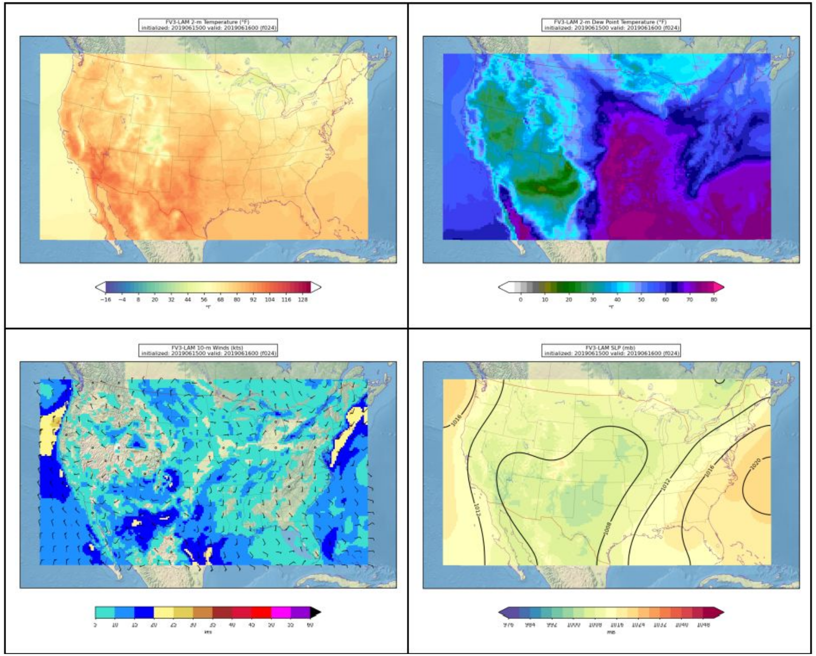
Download v1Beta plots

Create an INDY-v1beta subdirectory in your plots directory

$ cd plots/INDY-v1beta

$ scp ubuntu@your-ip-address:”expt_dirs/GST_INDY_v1beta /20191518/postprd/*.png” .

Open with a browser or your favorite image viewer



Create plots of differences

#When you have two experiments on identical grids, the SRW provides a script that allows you to plot the differences between the forecasts. The plot_allvars_diff.py script is in ush/Python. The script takes 8 arguments, which are described within it.

$ python ~/ufs-srweather-app/regional_workflow /ush/Python/plot_allvars_diff.py 2019061518 1 6 1 ~/expt_dirs/GST_INDY_v16 ~/expt_dirs/GST_INDY_v1beta /contrib/GST/NaturalEarth INDY

```
#
# Instructions:          Make sure all the necessary modules can be imported.
#                        Seven command line arguments are needed:
#                        1. Cycle date/time in YYYYMMDDHH format
#                        2. Starting forecast hour
#                        3. Ending forecast hour
#                        4. Forecast hour increment
#                        5. EXPT_DIR_1: Experiment 1 directory
#                           -Postprocessed data should be found in the directory:
#                             EXPT_DIR_1/YYYYMMDDHH/postprd/
#                        6. EXPT_DIR_2: Experiment 2 directory
#                           -Postprocessed data should be found in the directory:
#                             EXPT_DIR_2/YYYYMMDDHH/postprd/
#                        7. CARTOPY_DIR:  Base directory of cartopy shapefiles
#                           -Shapefiles cannot be directly downloaded to NOAA
#                             machines from the internet, so shapefiles need to
#                             be downloaded if geopolitical boundaries are
#                             desired on the maps.
#                           -File structure should be:
#                             CARTOPY_DIR/shapefiles/natural_earth/cultural/*.shp
#                           -More information regarding files needed to setup
#                             display maps in Cartopy, see SRW App Users' Guide
#                        8. POST_OUTPUT_DOMAIN_NAME:  Name of native domain
#                           used in forecasts and in constructing the names
```

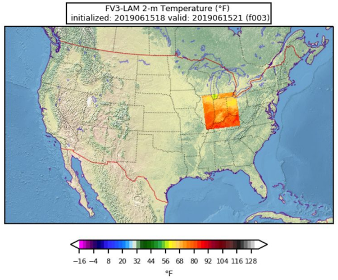
Download difference plots

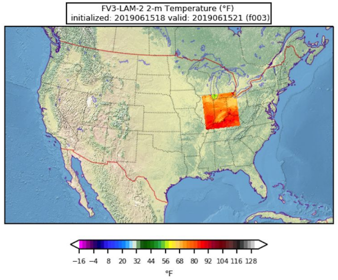
Create a diffs subdirectory in your plots directory

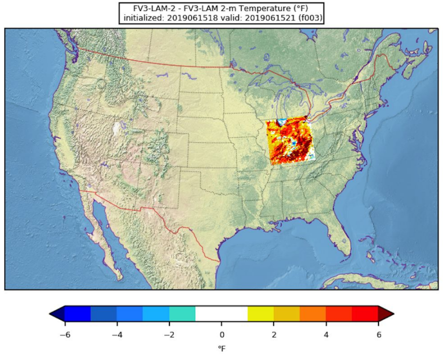
$ cd plots/diffs

$ scp ubuntu@your-ip-address:"expt_dirs/GST_INDY_v16/20191518/postprd/*diffs*.png" .

Open with a browser or your favorite image viewer







Try it out on your own!

We are available to take questions through zoom and slack

Getting Help

Epic.noaa.gov

Upcoming we will be adding an SRW landing page to our website. Click “Get Code” on the homepage to be directed to releases, supporting documentation, forums, videos, and FAQs.