Troubleshooting HWRF

Sam Trahan

January 2016

Troubleshooting HWRF

- Where and why has my workflow stopped?
- Workarounds.
- Manual Execution
- Why has component X failed?

Stopped Workflow
rocotostat

- Rocoto:
  - More on this later today.
  - rocotostat - tells you status of your jobs
    - UNKNOWN = batch system is probably down
    - UNAVAILABLE = batch system is definitely down
    - DEAD = job failed too many times
    - RUNNING = last seen running. If it is not running, run rocotorun to check the status.
    - QUEUED = keep your fingers crossed
    - SUBMITTING = job was submitted. Run “rocotorun” again to update the status.
    - - = dependencies not met or a throttle was reached

Stopped Workflow
rocotocheck

- Rocoto:
  - More on this later today.
  - rocotocheck - details about one job
    - Look at bottom for why job was not submitted.
    - Jobs will not be submitted until dependencies are met.
    - Will not be submitted if throttles are reached.
      - Maximum cycles.
      - Maximum running jobs.
    - Will not be submitted if cycle is inactive (launcher not yet run or completion job was run).
    - Will not be submitted if you forget to run rocotorun!

[config] allow_fallbacks=yes
Turn on Automatic Fallbacks

- Many possible failures can be detected and worked around automatically.
  - [config]
  - allow_fallbacks=**yes**
- Enabled by default in operations.
- Disabled elsewhere to allow detection of failures.

Manual Modification of Settings COMhwrf/stormX.conf

- X = 1, 2, 3, ..., 8
- Configuration file generated by launcher.
  - Can edit settings and resubmit jobs!
- Other files in COM critical to workflow:
  - stormX.gsi_status:
    - run_gsi_d02 = YES or NO
    - run_gsi_d03 = YES or NO
    - Set both to NO to trick scripts into disabling GSI

Manual Modification of Settings COMhwrf/stormX.conf

- Other files in COM critical to workflow:
  - stormX.ocean_status
    - RUN_COUPLED = YES or NO
    - Set to NO to disable ocean
  - storm1.run_ensda
    - RUN_ENSDA=YES or NO
    - Set to NO to disable ENSDA

Run GSI without ENSDA

- Workflow can continue without it.
- To forcefully disable, edit COM/stormX.conf:
  - [config]
  - run_ensemble_da=**no**
- Rerun GSI

Run Forecast without GSI

- Standard workflow runs GFS analysis vortex relocation as a backup.
- Failure of GFS ENKF can cause failure of HWRF GSI jobs.
  - [config]
  - run_gsi=**no**
- Resubmit forecast and it will run with relocated GFS analysis vortex.

Rerunning the Forecast

- When rerunning the forecast ALWAYS rerun the unpost and the post-processing:
  - resubmit forecast
  - ... wait for forecast to run and then ...
  - resubmit unpost
  - ... wait for unpost to finish and then ...
  - resubmit post
  - resubmit products
- Can be done automatically in Rocoto by rewinding all of them.

Rerunning the Post

- To continue post from where it started:
  - resubmit post - will only repost files not yet posted
  - resubmit products - rerun alerts, regrid files not yet done
    - Will rerun tracker from beginning.
- To discard and rerun entire post:
  - resubmit unpost - clear out database info
  - ... wait for unpost to finish, and then ...
  - resubmit post - convert wrfout to native grib
  - resubmit products - convert to pressure grib

Manual Execution

- The HWRF system is a layered system.
- The underlying HWRF implementation is independent of the choice of workflow manager.
  - Or the choice of having NO workflow manager.
- If a workflow fails, you can switch mid-workflow to manual execution at one of several levels.

Wrappers

- Simple shell scripts that execute a scripts/ex*py
- Each script:
  - Load modules
  - Set a few key variables
  - Runs the scripts/exwhatever.py
- Users must submit to their batch system.
- Extensive documentation on DTC website.

Directly Run ex-scripts

- The scripts/ex*py are designed to be runnable directly.
- Would you rather:
  - Fix a bug, wait a half hour for a new job to queue, fix another bug, wait another half hour, ...
  - OR: Run the same ex-script multiple times in the same interactive debug job?

Directly Run ex-scripts
exhwrf_launch

- Most complicated job to run:

```
cd /path/to/HWRF/scripts
export HOMEhwrf=/path/to/HWRF
export USHhwrf=/path/to/HWRF/ush
./exhwrf_launch.py 2015092912 11L HISTORY \
    /path/to/HWRF/parm \
    config.EXPT=HWRF \
    ... more configuration options ...
```

- Sets up working directory, makes conf and holdvars

Directly Run ex-scripts
most other jobs

- Later jobs:

```
bash
. /path/to/com/2015092912/11L/storm1.holdvars.txt
export PYTHONPATH=$USHhwrf
export TOTAL_TASKS=24 # number of MPI ranks
$EXhwrf/exhwrf_post.py
```

- The holdvars file exists just so we can do this.
- Some jobs require more variables to specify which piece of work to do (ie.: GDAS vs GFS init)
  - See the documentation, or mimic the rocoto/tasks/*.ent

Interactive Python

- If you have already run the launcher:

```
bash
. /path/to/com/2015092912/11L/storm1.holdvars.txt
export PYTHONPATH=$USHhwrf
export TOTAL_TASKS=24 # number of MPI ranks
python
>>> import hwrf_expt, produtil.setup
>>> produtil.setup.setup()
>>> hwrf_expt.init_module()
>>> hwrf_expt.nonsatpost.run()
```

- Allows fine-grained control over debugging.

Why has [X] failed?

- Steps for debugging:
  1. Check main jlogfile for high-level messages.
  2. Check job's log for detailed messages
     - Usually the error is near the end.
     - Usually not at the end. At the end, there is a listing of directory contets, and other useful diagnostics.
     - Look for the last few stack tracebacks.
  3. Check detailed logs from individual programs.
     - WRF: rsl.out.0000
     - prep_hybrid: prep.log
     - post: vpost.log
     - etc.

Why has [X] failed?
WRF

- WRF:
  - Crashes within <6 hrs are usually initialization issues.
    - Check ocean.
    - Check GSI.
    - Check for bad data in wrfinput, wrfanl, wrfbdy
  - Crashes after ~6hrs are model issues or bad boundary data.
  - Error messages in I/O servers about bad tags mean 32-bit integer wraparound. Increase number of I/O servers per group.

Why has [X] failed?
WRF

- WRF:
  - Error like “nest moved during coupling timestep”
    - Coupling timestep = 540 seconds.
    - Domain 2 motion must be on a coupling timestep.
    - Domain 3 does not have that restriction
  - Sudden exit with no warning?
    - Memory or disk quota issue.

Why has [X] failed?
GSI

- GSI:
  - Logs: $WORKhwrf/gsi_d0?/stdout
  - Usually fails due to missing data or memory issues.
    - Check to see all enkf, gdas, wrf ensda, bufr, etc are present.
  - Has gargantuan memory requirements.
    - Empty stdout means memory issues.
    - Sudden exit of job? Memory issues. (OOM killer)
    - Jet will report memory usage at bottom of job.
  - Reduce number of cores per node if memory issues occur.

Why has [X] failed?
initialization

- prep_hybrid or ungrib
  - Failures nearly always mean corrupted input files
  - OR you are using a low-res GFS/CFS/GEFS and need to change the prep_hybrid namelist
- geogrid
  - Write errors, or read errors in real, reading geogrid output?
  - Could be quota issues, or...
  - Large or high-resolution domains (9:3:1) can reach limits of NetCDF 3.x and PNetCDF file formats.
  - Switch to NetCDF 4 or binary if needed.

Why has [X] failed?
products

- Tracker: read main tracker log. Logging is extensive and detailed.
  - Failure reason is usually giving up waiting for input.
- Regridder:
  - Usually fails waiting for post, due to failed forecast.
  - OR due to disk quota or wallclock limit
  - For other failures, read the regridder logs and look for stack tracebacks.
- If you hit disk quota, don't try to rerun the post and products. Files are probably corrupted.
- Instead, resubmit unpost, then post and products.