Mechanics of running HWRF

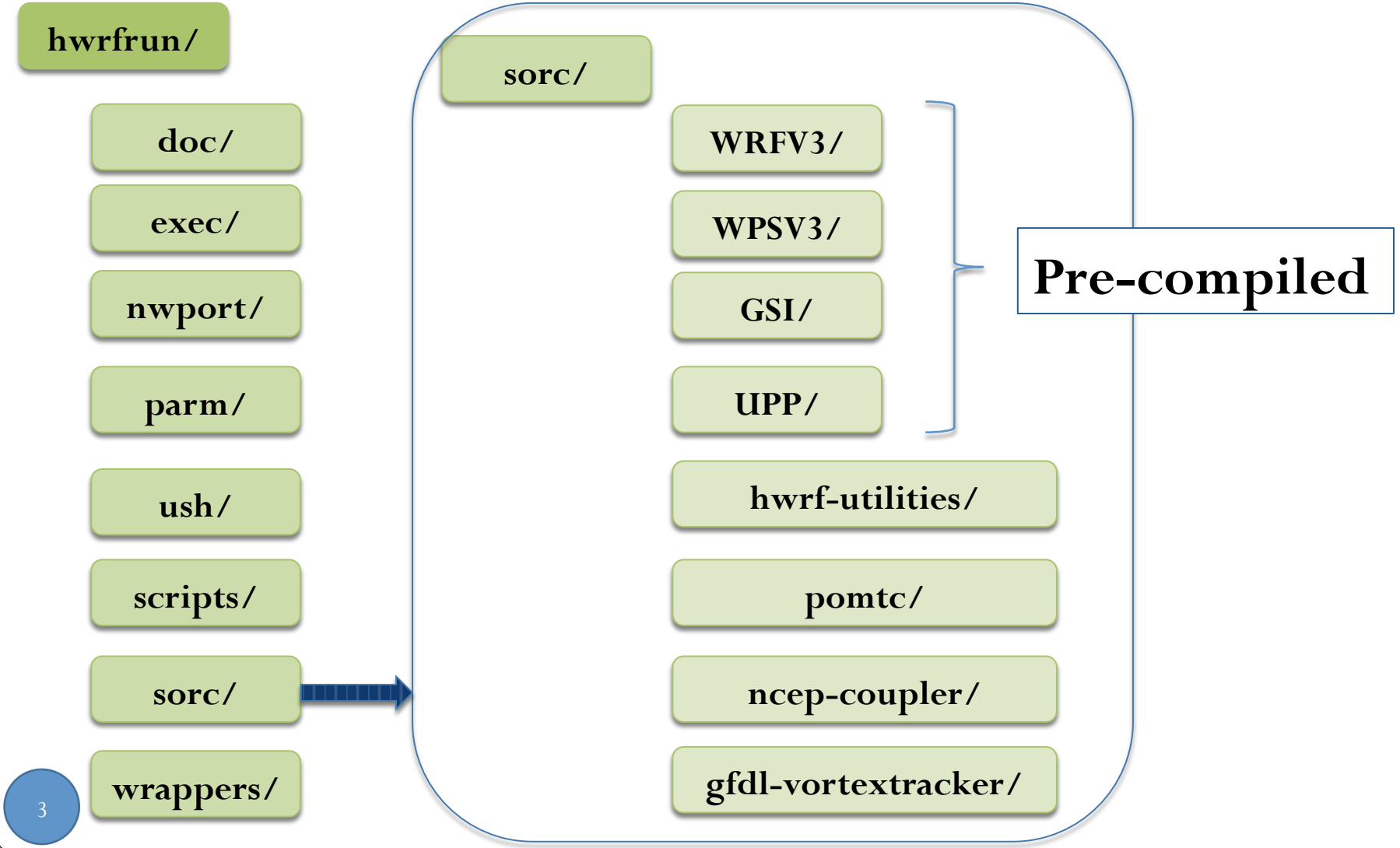
Mrinal Biswas

National Center for Atmospheric Research

For help with HWRF:
hwrf-help@ucar.edu

Directory Structure

Source code structure



Input data structure

fix/

```
bogus                hwrf-crtm-2.0.6      hwrf_eta_micro_lookup.dat  hwrf_storm_25               hwrf_track
hwrf_cpat.ice        hwrf-crtm-2.2.0      hwrf_fix_datestamp         hwrf_storm_cyn_axisy_47     hwrf_wps_geo
hwrf_cpat.moddef     hwrf-crtm-2.2.1      hwrf-pom                   hwrf_storm_cyn_axisy_50_ep  hwrf-wrf
hwrf_cpat.wind       hwrf_disclaimer.txt  hwrf_storm_20              hwrf_storm_radius           loop_curr
```

Fix files for GSI, Ocean, WPS, CRTM

gfs.2014101412/

```
gfs.t12z.1bamua.tm00.bufr_d  gfs.t12z.goesfv.tm00.bufr_d  gfs.t12z.sf06  gfs.t12z.sf24  gfs.t12z.sf72
gfs.t12z.1bhrs4.tm00.bufr_d  gfs.t12z.gpsro.tm00.bufr_d   gfs.t12z.sf102 gfs.t12z.sf30  gfs.t12z.sf78
gfs.t12z.1bmhs.tm00.bufr_d   gfs.t12z.mtiasi.tm00.bufr_d  gfs.t12z.sf108 gfs.t12z.sf36  gfs.t12z.sf84
gfs.t12z.airsev.tm00.bufr_d  gfs.t12z.pgrb2.0p25.f000     gfs.t12z.sf114 gfs.t12z.sf42  gfs.t12z.sf90
gfs.t12z.atms.tm00.bufr_d    gfs.t12z.prepbufr.nr         gfs.t12z.sf12  gfs.t12z.sf48  gfs.t12z.sf96
gfs.t12z.cris.tm00.bufr_d    gfs.t12z.sanl                gfs.t12z.sf120 gfs.t12z.sf54  gfs.t12z.sfcanl
gfs.t12z.esamua.tm00.bufr_d  gfs.t12z.satwnd.tm00.bufr_d  gfs.t12z.sf126 gfs.t12z.sf60
gfs.t12z.esamub.tm00.bufr_d  gfs.t12z.sf00                gfs.t12z.sf18  gfs.t12z.sf66
```

GFS spectral, obs for parent and sfc analysis for ocean

gdas.2014101406/

```
gdas1.t06z.abias     gdas1.t06z.pgrb2.1p00.f003  gdas1.t06z.pgrb2.1p00.f009  gdas1.t06z.sf06
gdas1.t06z.abias_pc  gdas1.t06z.pgrb2.1p00.f006  gdas1.t06z.sf03             gdas1.t06z.sf09
```

3,6,9 GDAS output from previous cycle for nest init

enkf.2014101406/

```
sfg_2014101406_fhr06s_mem001  sfg_2014101406_fhr06s_mem021  sfg_2014101406_fhr06s_mem041  sfg_2014101406_fhr06s_mem061
sfg_2014101406_fhr06s_mem002  sfg_2014101406_fhr06s_mem022  sfg_2014101406_fhr06s_mem042  sfg_2014101406_fhr06s_mem062
sfg_2014101406_fhr06s_mem003  sfg_2014101406_fhr06s_mem023  sfg_2014101406_fhr06s_mem043  sfg_2014101406_fhr06s_mem063
sfg_2014101406_fhr06s_mem004  sfg_2014101406_fhr06s_mem024  sfg_2014101406_fhr06s_mem044  sfg_2014101406_fhr06s_mem064
```

GFS 80 member ensemble files for DA

tdr.08l.2014101412/

Tail Doppler Radar data not present for this case

loop/

```
hwrf_gfdl_loop_current_rmy5.dat.20141012  hwrf_gfdl_loop_current_wc_ring_rmy5.dat.20141012
```

Loop current for ocean init

SYNDAT-PLUS/

```
syndat_tcvitals.2014
```

TC vitals

Configuring Experiment

HWRF Configuration

- Most configurable options are controlled by variables that live within the parm/ directory in conf files
- Four primary conf files control all options and required for each run and are called in the following order
  - hwrf_input.conf
  - hwrf.conf
  - hwrf_holdvars.conf
  - hwrf_basic.conf
- Users can create their own conf files (recommended for alternate configurations) and add them to the launcher

hwrf_input.conf

- Contains sections that specifies where to get input data. The hwrf.input classes read this to figure out where to pull data and where to put on disk.
- For this tutorial data is staged on disk

Location of files

```
[jet_hist_PROD2014]
inputroot2014=/lfs3/projects/hwrf-data/hwrf-input   ;; Input root location
inputroot=/lfs3/projects/hwrf-data/hwrf-input       ;; Input root location
…
```

Names of files

```
[gfs2014_naming]
gfs_sf    = gfs.t{aHH}z.sf{fahr:02d}  ;; GFS spect
gfs_sfcanl = gfs.t{aHH}z.sfcanl       ;; GFS surface a
…
```

!! NO NEED TO CHANGE ANYTHING FOR THE TUTORIAL!!

hwrf.conf

- Contains all the namelist-type parameters for all components

Fix file location

```
[dir]
geog_data={FIXhwrf}/hwrf_wps_geo/ ;; Geographic input data
FIXgsi={FIXhwrf}/hwrf-gsi/   ;; GSI input data for everything except CRTM
FIXcrtm={FIXhwrf}/hwrf-crtm-2.2.1/ ;; GSI CRTM input data
```

GSI namelist

```
[gsi_d02_nml]
HYBENS_REGIONAL=T ;; logical variable, if .true., then turn on hybrid
ensemble option
ENSEMBLE_SIZE_REGIONAL=80 ;; ensemble size
```

WRF namelist

```
[wrf_namelist]
physics.var_ric = 1.0
physics.coef_ric_l = 0.16
```


!! NO NEED TO CHANGE ANYTHING FOR THE TUTORIAL!!

hwrf.conf … Contd.

WPS Namelist

```
[wps_namelist]
share.wrf_core = 'NMM',
geogrid.map_proj =  'rotated_ll',
geogrid.geog_data_path = "{FIXhwrf}/hwrf_wps_geo/"…
```

WRF namelist

```
[moad_namelist]
physics.mp_physics = 5
physics.ra_lw_physics = 4
…
```

Location of executables

```
[exe]
wgrib={utilexec}/wgrib  ;; wgrib GRIB1 indexing and manipulation program
cnvgrib={utilexec}/cnvgrib  ;; cnvgrib GRIB1/2 conversion program
…
```

To see a full list of sections included in this file, type

```
grep "^\[" hwrf.conf
```


!! NO NEED TO CHANGE ANYTHING FOR THE TUTORIAL!!

hwrf_basic.conf

- The configuration file responsible for setting directory paths to which the later conf files refer
- Assumes another file has set CDSCRUB, CDSAVE, syndat and CDNOSCRUB variables in the [dir] section
- Configures the workflow-related variables in [config] section

[config]	workflow-related variables
[prelaunch]	configures overrides for default settings
[sanity]	configures sanity checks
[dir]	directory paths
[archive]	archiving locations and methods

!! NO NEED TO CHANGE ANYTHING FOR THE TUTORIAL!!

hwrf_basic.conf

```
[config]
forecast_length=12 ;; Length of the forecast in hours
run_gsi=yes          ;; GSI and FGAT initialization
run_ocean=yes        ;; POM coupling
...
use_spectral=yes
...

[dir]
WORKhwrf={CDSCRUB}/{RUNhwrf}/{vit[YMDH]}/{vit[stormid3]} -- Main working
directory
/glade/scratch/${USER}/pytmp/hwrfrun/2014101412/08L
HOMEhwrf={CDSAVE}/{EXPT} -- Main HWRF installation top directory
/glade/scratch/${USER}/HWRF_v3.7a/hwrfrun
com={CDSCRUB}/{RUNhwrf}/com/{vit[YMDH]}/{vit[stormid3]} -- COM directory for
communication between cycles: COMIN env
/glade/scratch/${USER}/pytmp/hwrfrun/com/2014101412/08L
```



!!DO NOT CHANGE ANYTHING IN [dir]!!

hwrf_v3.7release.conf

- A fifth conf file is provided for the community release
- Set the exact configuration you need
  - Run GSI?
  - Use Spectral files?
  - Change physics options
  - Location of input data staged on disk
  - And many more

```
[config]
input_catalog=comm_hist → Use data staged on disk
archive=none
publicrelease=yes
run_ensemble_da=no
scrub=no
```

hwrf_v3.7release.conf .. Contd.

```
[dir]
inputroot=PATH/TO/INPUT/DATA
## Syndat directory for finding which cycles to run
syndat={inputroot}/SYNDAT-PLUS
## Output root is the desired output location for HWRF runs
outputroot=PATH/TO/DESIRED/OUTPUT
## Non-scrubbed directory for track files, etc.
CDNOSCRUB={outputroot}/noscrub
## Scrubbed directory for large work files.
CDSCRUB={outputroot}/pytmp
## Save directory.  Must be the parent directory of the HWRF install
CDSAVE=/PATH/TO/HWRF/PARENT

[comm_hist]
inputroot=/PATH/TO/INPUT/DATA/
gfs={inputroot}/gfs.{aYMDH}/
gdas1={inputroot}/gdas1.{aYMDH}/
…
[exe]
gsi={HOMEhwrf}/sorc/GSI/run/gsi.exe
post={HOMEhwrf}/sorc/UPP/bin/unipost.exe
```

TCVitals

Output Data

Input Data

Paths to Executables

Edit paths for input/output, follow instructions

system.conf

- This is system specific configuration file for the machine you are running the jobs. In this tutorial we will use the one for NCAR's Yellowstone.

In ${SCRATCH}/hwrfrun/parm

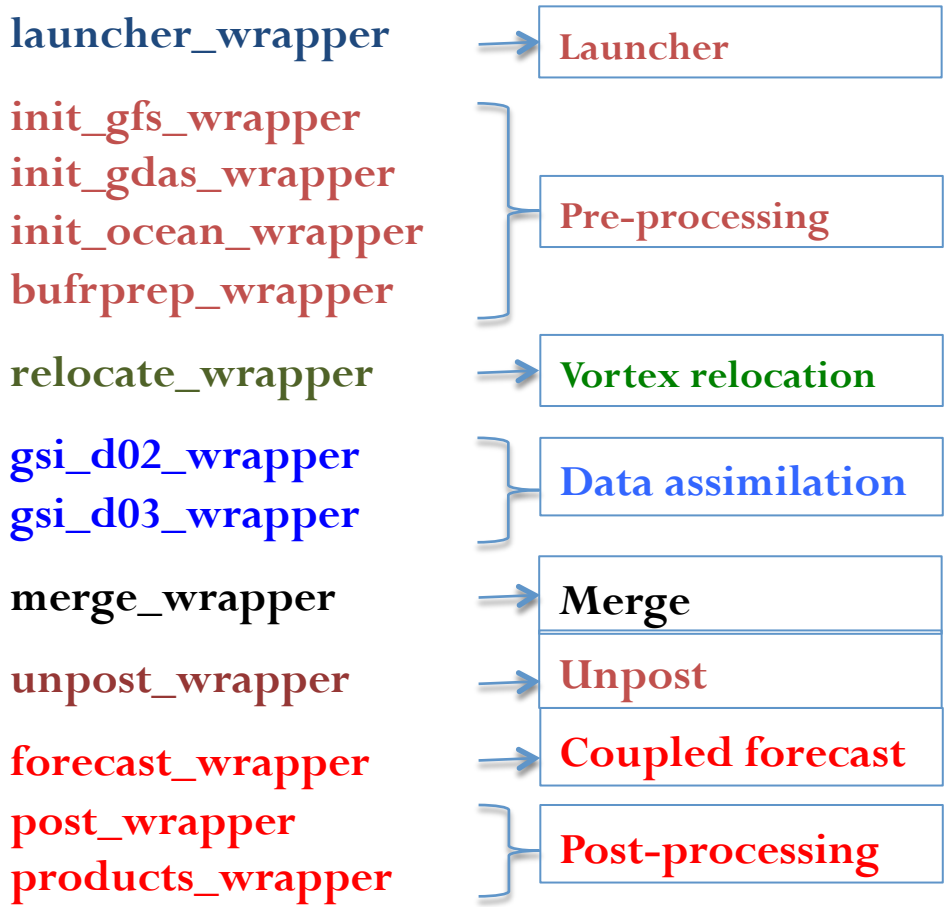
Link the system.conf.yellowstone to system.conf

!! NO NEED TO CHANGE ANYTHING FOR THE TUTORIAL!!

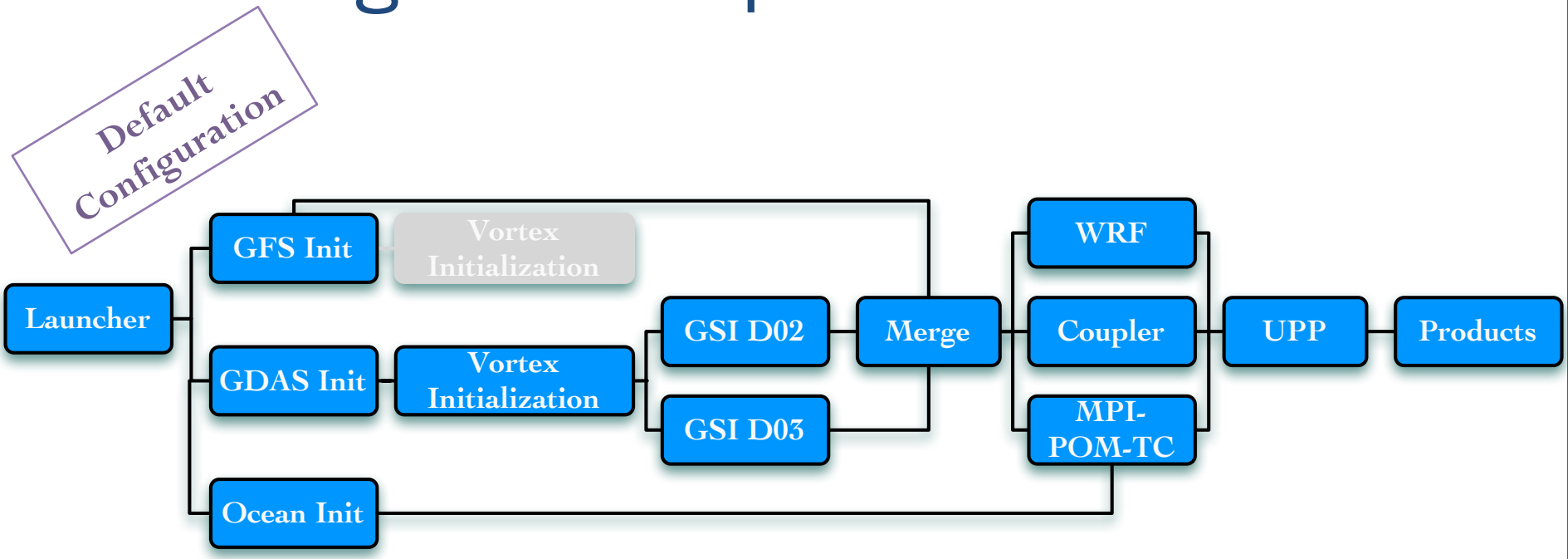
Wrappers for running HWRF

Wrappers

- Wrappers are shell scripts that runs a single component of the HWRF system



Configuration Options: Workflow



hwrf_basic.conf

```
run_gsi=yes
run_ocean=yes
run_relocation=yes
use_spectral=yes
```

Wrappers: global_vars.sh

- Each wrapper sources the global_vars.sh file, which sets a few variables required by each component

```
##### Definition of the Storm ######
export START_TIME=2014101412    # Initial start date
export SID=08L                  # Storm ID
export CASE=HISTORY             # HISTORY OR FORECAST

##### Location of HWRF installation #####
export HOMEhwrf=/PATH/TO/HWRF/INSTALLATION

export EXPT=`echo ${HOMEhwrf} | rev | cut -d/ -f1 | rev`

##### File containing the case-specific variables defined in launcher
#####
export startfile=${HOMEhwrf}/wrappers/$EXPT-${START_TIME}-$SID.start
```

Wrappers: launcher_wrapper

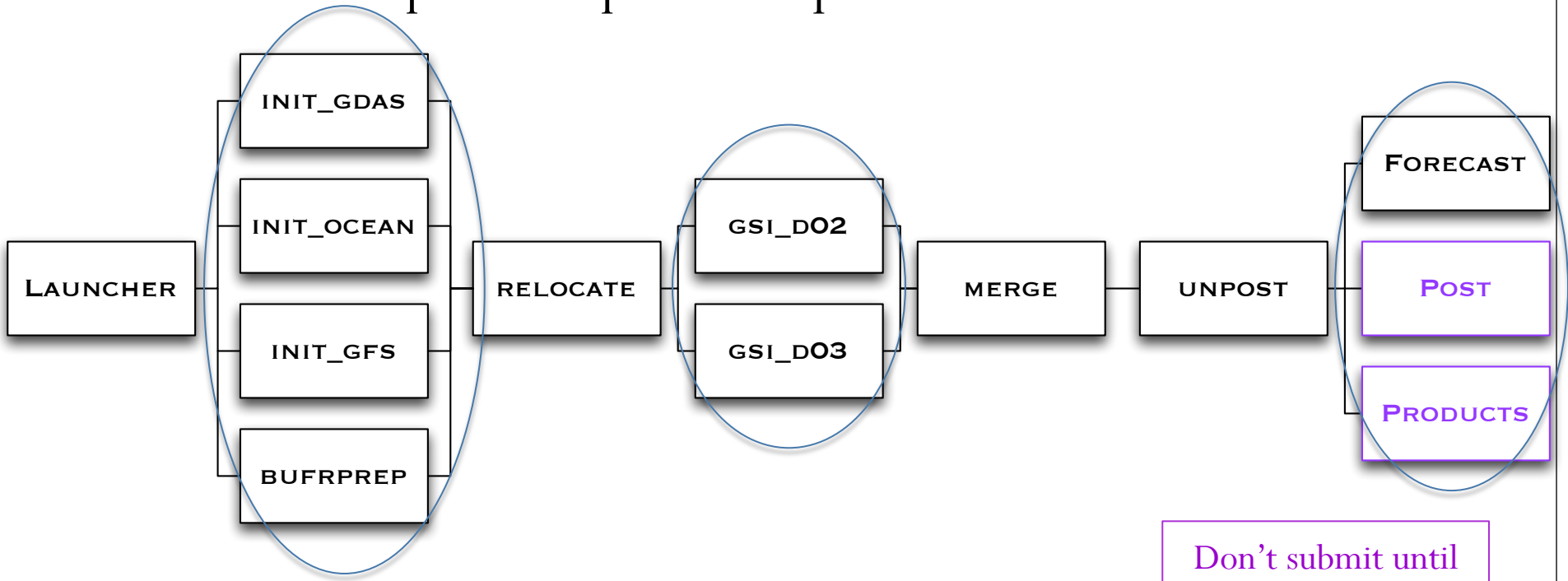
- launcher_wrapper runs the exhwrf_launch.py script, which is responsible for gathering configuration options, creating output directory structure, and writing storm1.conf.
- Loads few env variables from global_vars.sh

```
${HOMEhwrf}/scripts/exhwrf_launch.py "$YMDH" "$STID" "$CASE_ROOT"
"$HOMEhwrf/parm" \
"config.EXPT=${EXPT}" "config.startfile=${startfile}" \
"config.HOMEhwrf=$HOMEhwrf" "config.case_root=$CASE_ROOT" \
"$HOMEhwrf/parm/hwrf_v3.7release.conf" \
"$@"
```

- Gathers all the options for a particular user configuration into the single file **storm1.conf** in the ${COMhwrf} directory.
- If you change any configuration, **run launcher_wrapper again**

Wrappers

- Wrappers must be submitted in sequence
- Some wrappers may be submitted simultaneously, while others require completion of previous task before submission



Don't submit until forecast job is running

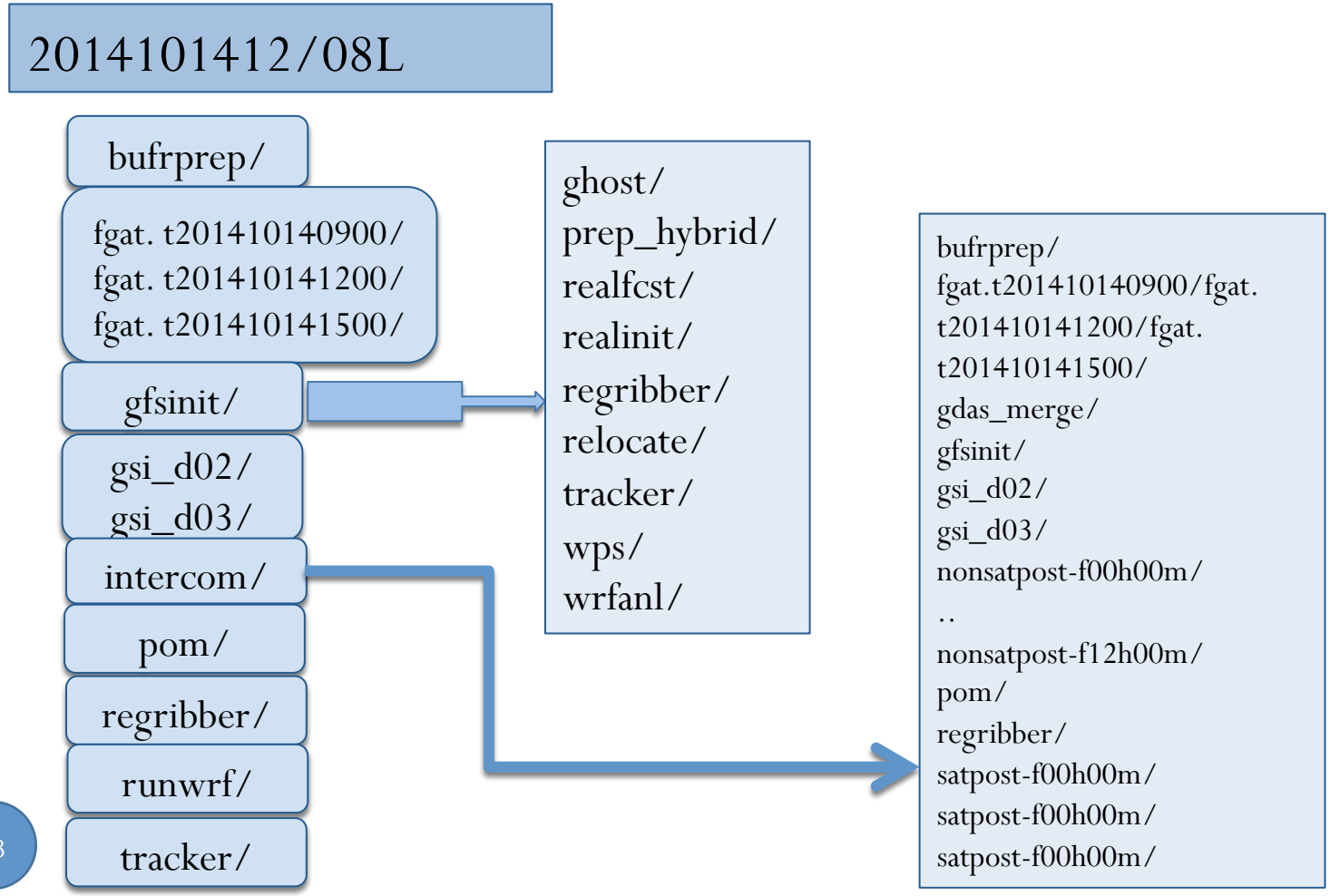
Cycled run

- This tutorial is designed to run a cold start
- Users willing to run cycled should follow the same procedure and sequence to run the wrappers as long as the previous 6 h outputs are present in the designated directories
- The scripts will determine the initial intensity and decide to use the previous 6 hr HWRF vortex or GFS vortex.

Output Directory Structure

Output directory

/glade/scratch/${USER}/pytmp/hwrfrun



Output directory

/glade/scratch/${USER}/pytmp/hwrfrun

com/2014101412/08L

- 3-hrly 9 h wrfout
- GRIB2 (pressure, sat vars)
- GRIB1 (tracker)
- Tracker outputs
- GSI diagnostics

Visualization

- Plotting Track and Intensity using gfdlvortex tracker plotting capability
- A 126 h run is available on disk which contains the HWRF operational run
  - Expect differences, because the tutorial exercise is a cold start, whereas the operational run is cycled.
- Grads scripts are provided to plot UPP outputs.
- Converting raw WRF outputs, using UPP to get the fields in pressure levels and then visualize using Grads scripts is provided in the Idealized run.

Thank you

- You can
  - Ask question during the tutorial
  - Consult our HWRF Users Guide

http://www.dtcenter.org/HurrWRF/users/docs/users_guide/HWRF_v3.7a_UG.pdf

  - Consult HWRF Scientific Documentation

http://www.dtcenter.org/HurrWRF/users/docs/scientific_documents/HWRF_v3.7a_SD.pdf

  - Contact me later: biswas@ucar.edu
  - Reach our help-desk: hwrf-help@ucar.edu

Backup slides

Command line arguments

```
{HOMEhwrf}/scripts/exhwrf_launch.py {DATE} {STID} HISTORY
    /path/to/parm config.EXPT={EXPT}
    config.startfile={startfile} config.SUBEXPT={anyname}
    config.run_gsi=no
```

- {DATE}
  - YYYYMMDDHH-YYYYMMDDHH for a range of cycles
  - YYYYMMDDHH for a single cycle
  - YYYYMMDDHH YYYYMMDDHH for two specific cycles
- *{STID}* is the storm ID, i.e. 18L for Sandy
- *{EXPT}* is the name of parent directory of `rocoto/`
- Can set any conf parameter in this line without editing a conf file
  - e.g. add option: `config.run_gsi=no`

New configure file

- Pass the directory/name of one extra configure file

```
{HOMEhwrf}/scripts/exhwrf_launch.py {DATE} {STID} HISTORY
    /path/to/parm config.EXPT={EXPT}
    config.startfile={startfile} config.SUBEXPT={anyname}
    hwrf_biswas.conf
```

hwrf_biswas.conf

```
[config]
disk_project=dtc-hurr
archive=none
scrub_com=no

[relocate]
scrub=no

[gsi_d02]
scrub=no
…
```