

Developmental Testbed Center Updates

Kathryn Newman^{1,2} Evan Kalina^{1,3} Louisa Nance^{1,2}

DTC team: Mrinal Biswas^{1,2}, Laurie Carson^{1,2}, James Frimel^{1,4}, Linlin Pan^{1,3}

¹DTC ²NCAR ³CIRES at NOAA/GSL ⁴CIRA at NOAA/GSD

With contributions from: Bin Liu^{1,2} and Zhan Zhang¹

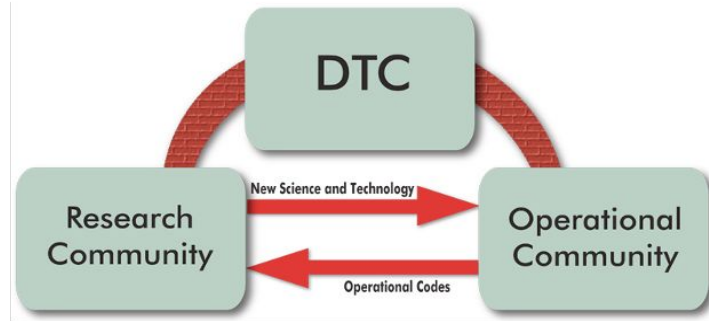
¹NOAA/EMC ²IMSG

Overview of the DTC

DTC purpose:

Facilitate the interaction and transition of NWP technology between research & operations

The DTC is a collaborative facility between NCAR & NOAA/ESRL/GSL



Strong partnerships with operational partners & model developers is critical

O2R: Support operational NWP systems to the **community**

R2O:

Partner with developers to get innovations into **centralized code**

Perform diagnostics and **T&E on promising NWP innovations** for possible operational implementation

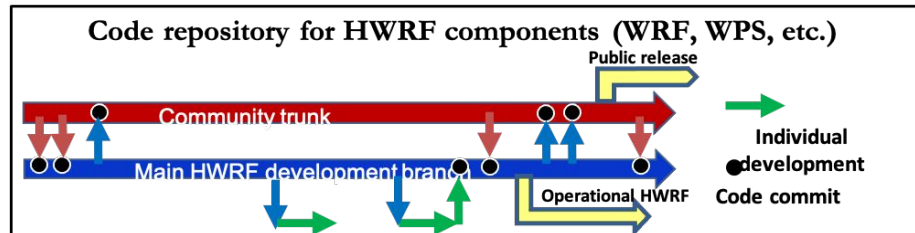
Interaction between **R&O:** Workshops, **visitor program**, newsletter



HWRF code management

Centralized HWRF repository

- SVN & Git repositories house all HWRF components
- Automated build for entire system, end-to-end python scripts, tools for automation, source for components
- Support unified scripts that run all HWRF components
- Conduct regression, consistency checks to ensure code integrity
 - Keep truck up to date, H220 (H221) merge
- Streamline access/checkout for components that move from VLab to GitHub
 - ProdGSI move from VLab to Github
- Sync main HWRF development branches with community trunks



User and Developer Support

- Helpdesks serving HWRF (user/developer) and stand-alone GFDL Vortex Tracker migrated to community forums
 - DTC continues to monitor and answer questions –community contributions encouraged!

<https://dtcenter.org/forum/hwrf-support/developers>

<https://dtcenter.org/forum/hwrf-support/users>

<https://dtcenter.org/forum/gfdl-vortex-tracker-support>

- New email (hwrf-access@ucar.edu) for collecting repository requests
- —DTC chairs bi-weekly HWRF developer committee calls
 - Mondays @ 12 pm ET
 - Developer mailing list

www.dtcenter.org/HurrWRF/users

www.dtcenter.org/HurrWRF/developers



User and Developer Support

- Streamlining the transition of new developments (developer support)
 - Access to the unified HWRF code repository with experimental codes
 - Support for inter-developers collaboration
 - Training, assistance with developments, specialized helpdesk
 - Oversight of code integration to avoid divergence
- Successful port of HWRF system to Orion HPC!
 - Platform with easier access – enabling HWRF development
 - Current input task/restricted data limitations
- Developments merged into HWRF centralized repository
 - Rob Fovell (U. Albany)/Joe Olson (NOAA/GSL): *Updated MYNN scheme for HWRF*
 - Mike Iacono/John Henderson (AER): *Cloud overlap with varying decorrelation length (RRTMG)*
 - Ryan Torn/Xiaohui Zhao (U. Albany): *Implement SPPT in HWRF ensemble*

Testing and Evaluation

MYNN Planetary Boundary Layer replacement

- Testing to replace GFS-EDMF PBL with MYNN PBL
 - Development by Rob Fovell (U. Albany) via DTC visitor program in collaboration with Joe Olson (NOAA/GSL)
- Merged latest MYNN code from WRF community, integrated Rob's development
 - Development committed to HWRF trunk

Determined scheme not ready for operational implementation

Cloud Overlap with varying decorrelation length (RRTMG) update

- Testing of upgraded cloud overlap scheme
 - Development by Mike Iacono and John Henderson (AER) via DTC visitor program
- Iterative testing of different configurations to determine optimal performance
- Final configuration resulted in **~5% improvement in the 3-5 day hurricane track forecast.**

Cloud Overlap changes implemented in 2020 operational HWRF

HAFS Repository Management

The official/authoritative HAFS repository:

- <https://github.com/NOAA-EMC/HAFS>
- Supports the main development activities and operational implementations.
- Mainly hosts the **develop** and **master** branches, plus some implementation branches/tags.

The community/organizational HAFS forks:

- e.g., <https://github.com/hafs-community/HAFS>
- Mainly provides community support and promotes organizational level collaborations.
- Host HAFS related developments for submodule repositories/forks.

The personal HAFS forks:

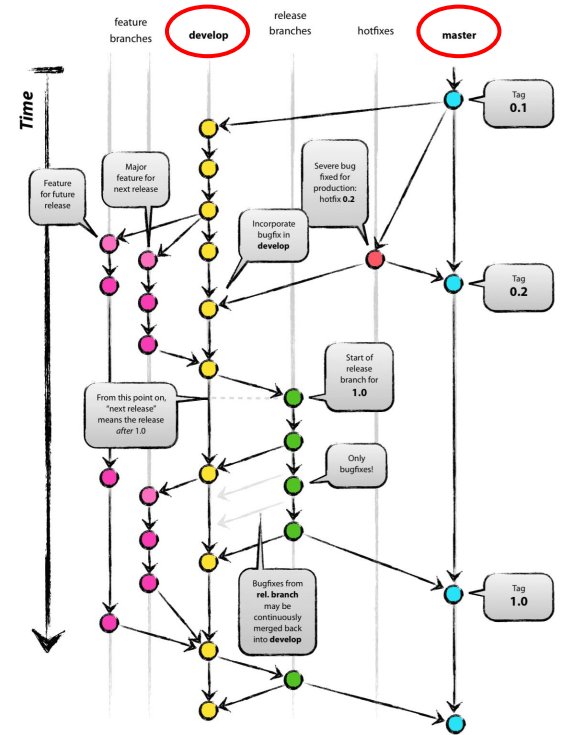
- Developer's forks for individual feature (or capability) development.
- New developments/features can be integrated back into the authoritative repository or the trusted forks through GitHub Pull Requests.

Guidance Documents

- [NCO Implementation Standards](#)
- [NCEP EE2 Guidance](#)
- [HAFS GitFlow Rational](#)

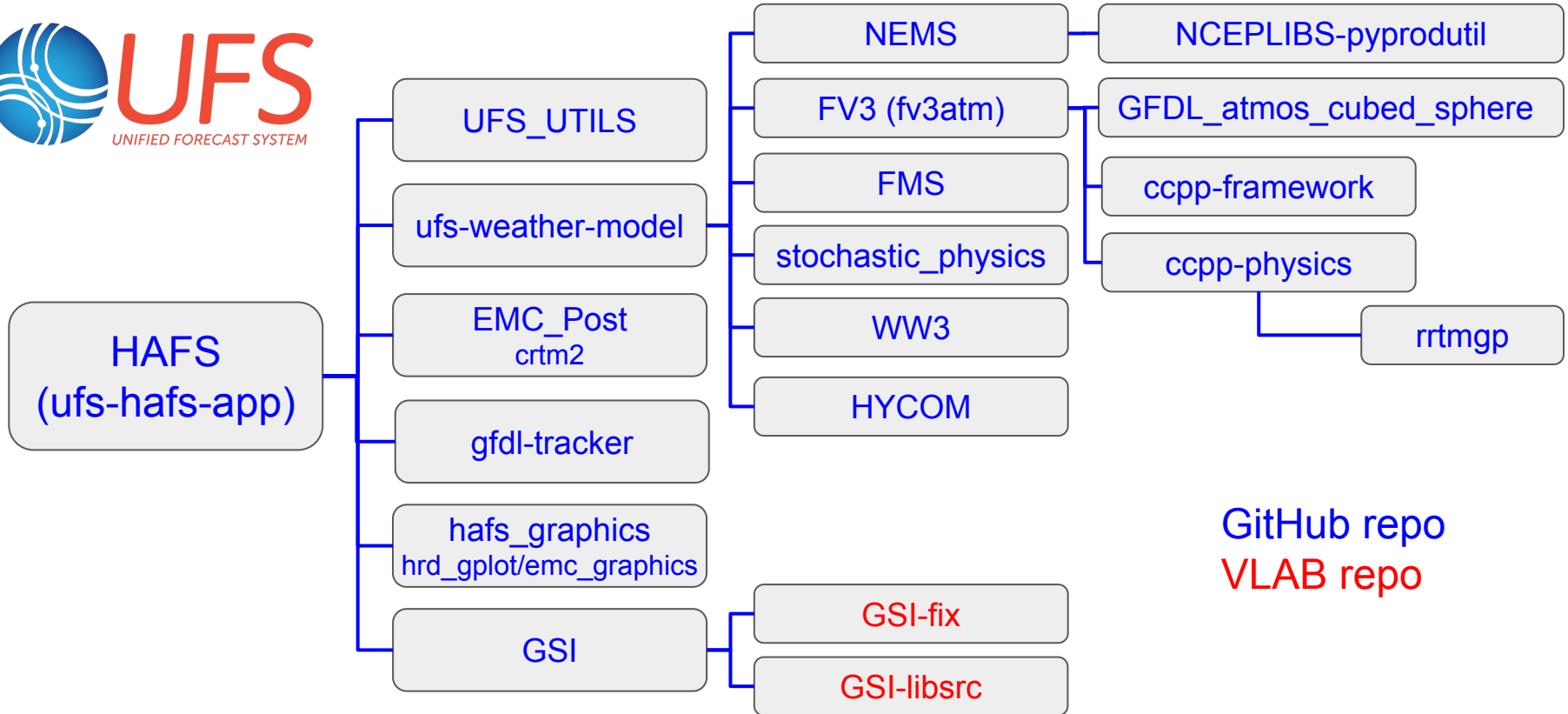
Branch Convention:

- **develop**
- **master**
- **feature/[name]**
- **support/[name]**
- **product/hafs.vx.x.x**



GitFlow branching strategy (from Vincent Driessen's blog)

HAFS Subcomponents/Submodules



GitHub repo
VLAB repo

A Quick Start for HAFS Users

A. Clone and checkout

```
git clone --recursive https://github.com/NOAA-EMC/HAFS.git
```

B. Build and install

```
cd HAFS/sorc  
./build_all.sh  
./install_all.sh  
./link_fix.sh
```

C. Configure and run HAFS

```
cd HAFS/parm  
cp system.conf.jet system.conf  
cd HAFS/rocoto  
vi cronjob_hafs.sh  
./cronjob_hafs.sh
```

Repeat running this driver periodically or add it as a cron task to advance the workflow.

Cheers! You are now running HAFS!



```
#!/bin/sh  
cd /ifs3/projects/hwrfv3/${USER}/HAFS/rocoto  
./run_hafs.py -f -s sites/xjet.ent 2018 06L HISTORY # Florence
```

Note: a detailed HAFS developer guide can be found [here](#).

HAFS governance and regression testing

Additional governance considerations for hafs-community repository

- Code review committee
- Regression testing – run simple tests on multiple platforms.
 - Potential for DTC to aid in this process
- Frequency of sync between hafs-community and authoritative repository
 - Required frequency of developers to sync their folks with authoritative repository?
 - Would help mitigate divergence issues!