The HWRF Development Process

Ligia Bernardet

Developmental Testbed Center

In collaboration with the HWRF Developers Committee

Vijay Tallapragada, EMC

Sam Trahan, EMC

Christina Holt, DTC

Acknowledgments to DTC staff (M. Biswas, L. Carson, H. Shao, D. Stark, M. Hu, K. Fossell), EMC HWRF team, and HWRF users and developers

Calendar: Operational Implementation

Activities	Approximate dates
Development of upgrades	Ongoing
Final development of proposed upgrades	September - December
Test of individual proposed upgrades	December - March
Final test of combined proposed upgrades	March
Pre-implementation test at NCO	April
HWRF operational implementation (AL & EP)	May
HWRF public release	August

WRF public release

GSI public release

HWRF distributed development

- Examples of HWRF activities currently going on
  - DTC: changes to compilation/configuration to support public
  - EMC: ensemble capability in python scripts
  - ESRL/OU/EMC: regional HWRF ensemble (EnKF) in DA
  - URI: alternate ocean initialization based on NCODA
  - UCLA: new eddy-mixing formulation in PBL scheme
  - DTC: updates to WRF from community (sync with v3.6.1)
  - EMC: improvements to vortex initialization
  - CIMMS: upgrades to UPP synthetic satellite images
  - etc.

Q: How do we move forward together with distributed developments?!?
A: With effective communication and a robust HWRF code management!

Communications

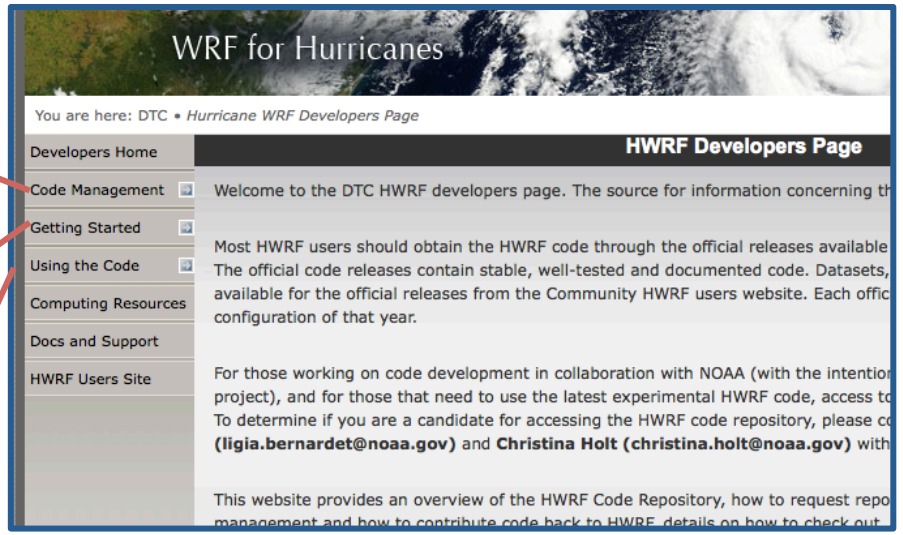
- HWRF Developers Committee
  - Membership: 2 from DTC, 2 from EMC
  - All developers welcome to meetings (Monday noon ET)
  - Forum for discussion, plans, and updates for development
  - Including testing, evaluation, and technical aspects
- Mailing list for exchanging information about development
  - hwrf_developers@rap.ucar.edu
  - All those with HWRF repository access are members
- Additional meetings scheduled as needed
  - Example: developers of HWRF regional ensemble with EnKF are meeting weekly now because of fast development phase

HWRF Developers Website

Code Management	Overview
Getting Started	Code Development Process
Using the Code	Roles and Responsibilities
Computing Resources	Testing

Getting Started	Obtaining Repository Access
Using the Code	Repository Structure
Computing Resources	Code Structure

Using the Code	Checking Out the Code
Computing Resources	Development Branches
Docs and Support	Build & Install
HWRF Users Site	Running HWRF



Extensive resources for developers: http://www.dtcenter.org/HurrWRF/developers

The centralized HWRF repository

- With one command the HWRF repo can be obtained

  svn co https://svn-dtc-hwrf.cgd.ucar.edu/trunk HWRF
- One more command for GSI and another command for HYCOM
- What is included
  - End-to-end python scripts
  - Tools for automation using the Rocoto Workflow Manager
  - Source for components
    1. WRF: atmospheric model
    2. WPS: global model pre-processor
    3. HWRF-utilities: libraries, utilities, and vortex initialization
    4. GSI: data assimilation
    5. MPIPOM-TC: ocean model
    6. HYCOM (optional)
    7. Coupler
    8. UPP: postprocessor
    9. GFDL Vortex Tracker

Origin of components

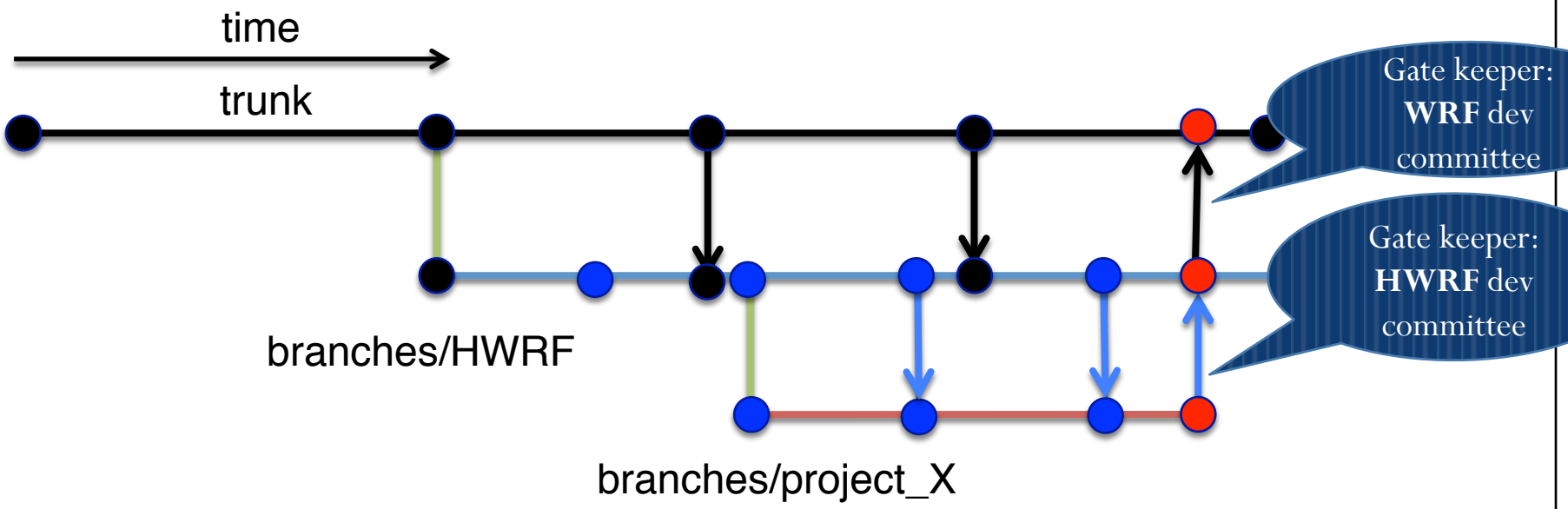
Component	SVN code repository
WRF	`https://svn-wrf-model.cgd.ucar.edu`
WPS	`https://svn-wrf-wps.cgd.ucar.edu/`
HWRF-utils	`https://svn-dtc-hwrf-utilities.cgd.ucar.edu`
Coupler	`https://svn-dtc-ncep-coupler.cgd.ucar.edu`
MPIPOM-TC	`https://svn-dtc-pomtc.cgd.ucar.edu`
HYCOM	`https://svn-dtc-hycom.cgd.ucar.edu`
UPP	`https://svn-dtc-unifiedpostproc.cgd.ucar.edu`
Tracker	`https://svn-dtc-gfdl-vortextracker.cgd.ucar.edu`
GSI	`https://svnemc.ncep.noaa.gov/projects/gsi`
	`https://gsi.fsl.noaa.gov/svn/comgsi`

Source code for components comes from their own repositories

In HWRF repo, these are *externals*, that is, *links* to other repos

Important because code is not being duplicated, helps avoids divergence

Example for WRF development



- WRF repository is hosted at NCAR, gets contributions for non-HWRF groups
- Branches/HWRF is used for centralizing all HWRF development
  - DTC updates it periodically from trunk (black arrows)
- Branches/projects are used by members of a project for development
  - Developers update them periodically from branches/HWRF
- When development is ready/tested, it gets committed to trunk

Important! Avoids divergence

Access for developers

- Account on SVN repositories (takes 2 weeks)
  - EMC arranges access to EMC GSI repository
  - DTC arranges access to all other repositories
- HFIP PIs can apply for accounts/projects on NOAA's Jet
  - Follow instructions at https://rdhpcs-s.noaa.gov/acctmgmt
  - Let Robert Gall (robert.gall@noaa.gov) know you're applying
  - Contact Nysheema Lett (Nysheema.Lett@noaa.gov) for a NOAA email address if you don't have one
  - Jet Questions go to Jet Help Queue (rdhpcs.jet.help@noaa.gov)
  - For help determining needed resources, email Christina or Ligia

Helpful Jet documentation: https://sites.google.com/a/noaa.gov/oar-jetdocs/

Great resource for svn: http://svnbook.red-bean.com/

What else is needed to run HWRF?

- Fix files (topography, microphysics tables etc.)
  - Available from DTC
- Input datasets (GFS, GDAS, GFS ensemble, obs etc.)
  - Available in NOAA HPSS but a challenge in other platforms
- Two running options:
  - Simple, step by step: use *wrappers* to submit python scripts
    - Instructions are available in HWRF Users' Guide v3.6a
  - Automated: use *Rocoto Workflow Manager*
    - Documentation available here: http://rdhpcs.noaa.gov/rocoto/
    - Details for using with HWRF: `HWRF/README.rocoto`
    - **Training will be provided by DTC in a few weeks**

New Object-Oriented Python scripts

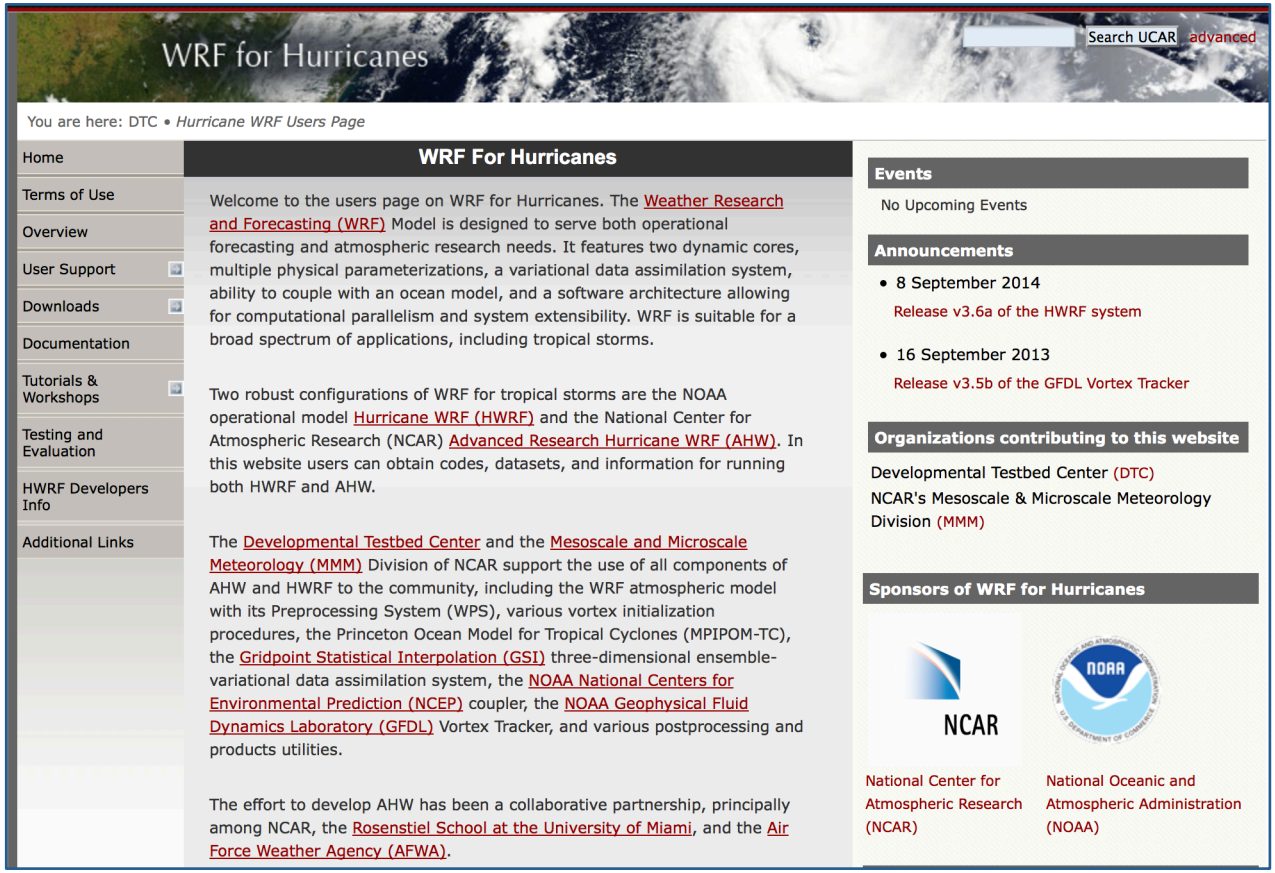
- Recently developed by EMC and DTC
  - At least 3x less lines than previous ksh scripts
  - Modular, small blocks make it easier to reuse code
  - No hardcodes, all configuration is abstracted
- Partially implemented in 2014 operational HWRF
- End-to-end now available in HWRF repo and public release

It is not necessary to know Python to run HWRF.
For introducing changes to HWRF workflow, familiarity with Python and HWRF is required.

Documentation in public wiki:
https://wiki.ucar.edu/display/DTCHWRF/DTC+HWRF+Scripts+Home

www.dtcenter.org/HurrWRF/users

HWRF Public Release



Current release: HWRF v3.6a (2014 operational)

Yearly releases, code downloads, datasets, documentation, helpdesk

800 registered users

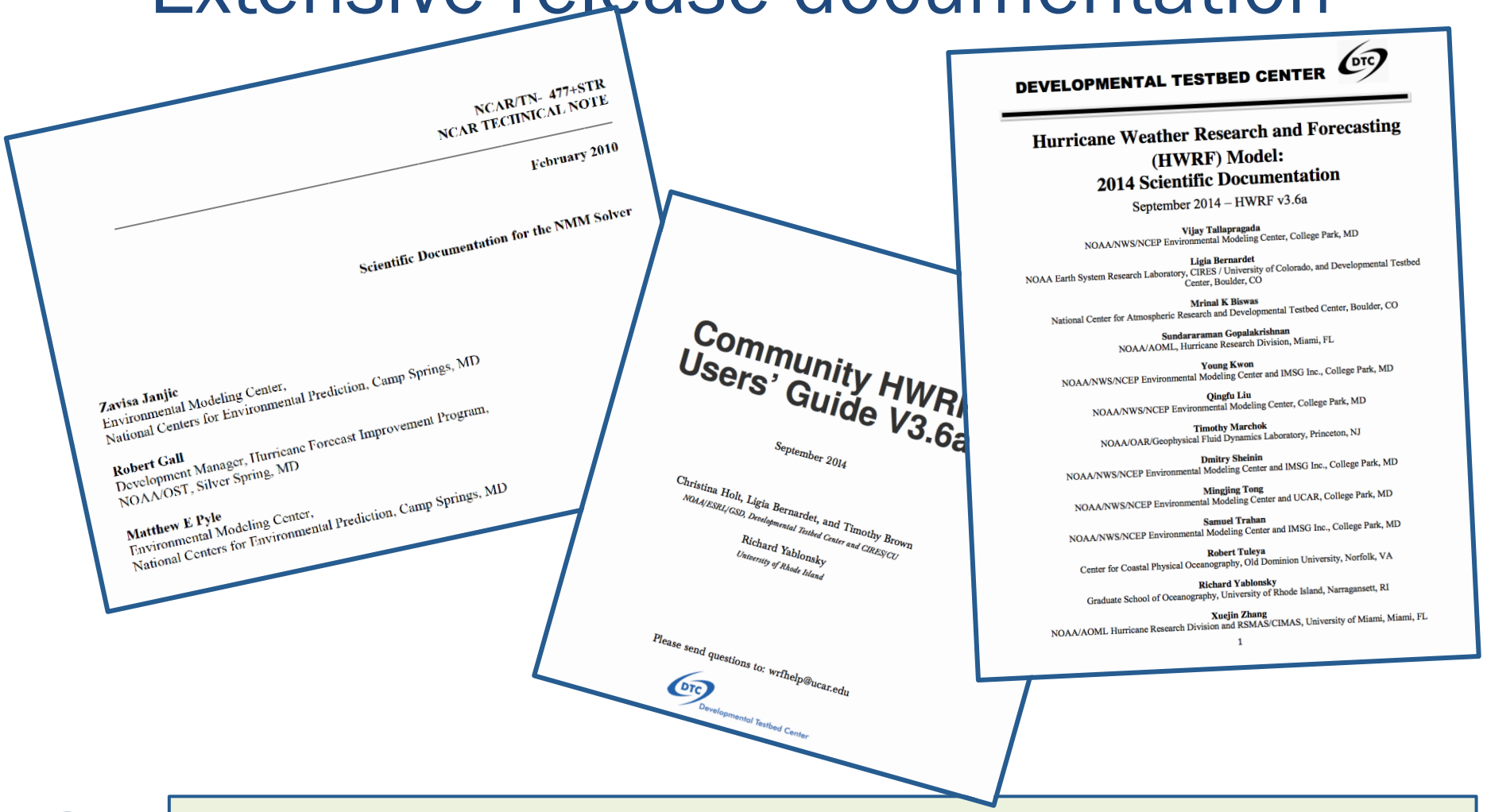
Stable, tested code

Operational and research capabilities (idealized simulation, alternate physics)

Ideal for users, not developers

2014 Tutorials: College Park, MD and May in Taiwan (36 participants from 10 countries

Extensive release documentation



Users Guide: http://www.dtcenter.org/HurrWRF/users/docs/users_guide/HWRF_v3.6a_usersguide.pdf
HWRF Sci Doc: http://www.dtcenter.org/HurrWRF/users/docs/scientific_documents/HWRFv3.6a_ScientificDoc.pdf
NMM Solver Sci Doc: http://www.dtcenter.org/HurrWRF/users/docs/scientific_documents/NMM_scientific_2-2-10_final.pdf

Summary

- DTC facilitates access to HWRF code for users and developers
- Lots of resources, websites, and documentation
- It is very important that developers follow code management so new code becomes available for operational testing
- We are here to help! Please contact us if you would like more information about the development process