ENHANCING COMMUNITY COLLABORATIONS THROUGH NWP SOFTWARE CONTAINERS

Michael J. Kavulich, Jr., John Halley Gotway, Jamie Wolff, Kate Fossell, Michelle Harrold, Tara Jensen, Tatiana Burek, John Exby

National Center for Atmospheric Research

Developmental Testbed Center

kavulich@ucar.edu

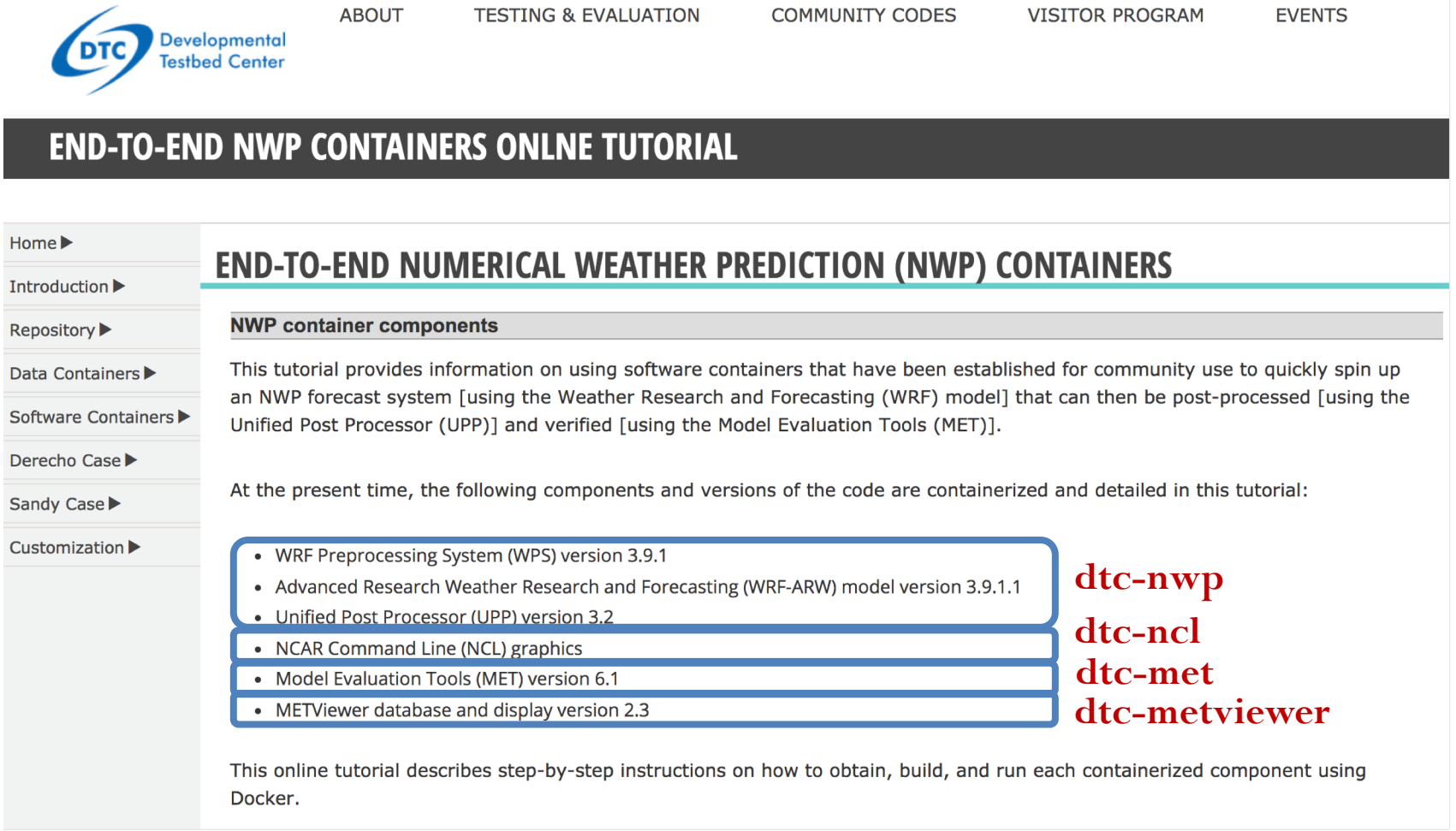
1st Annual Joint WRF/MPAS Users Workshop

Boulder, Colorado, USA

12 June 2018

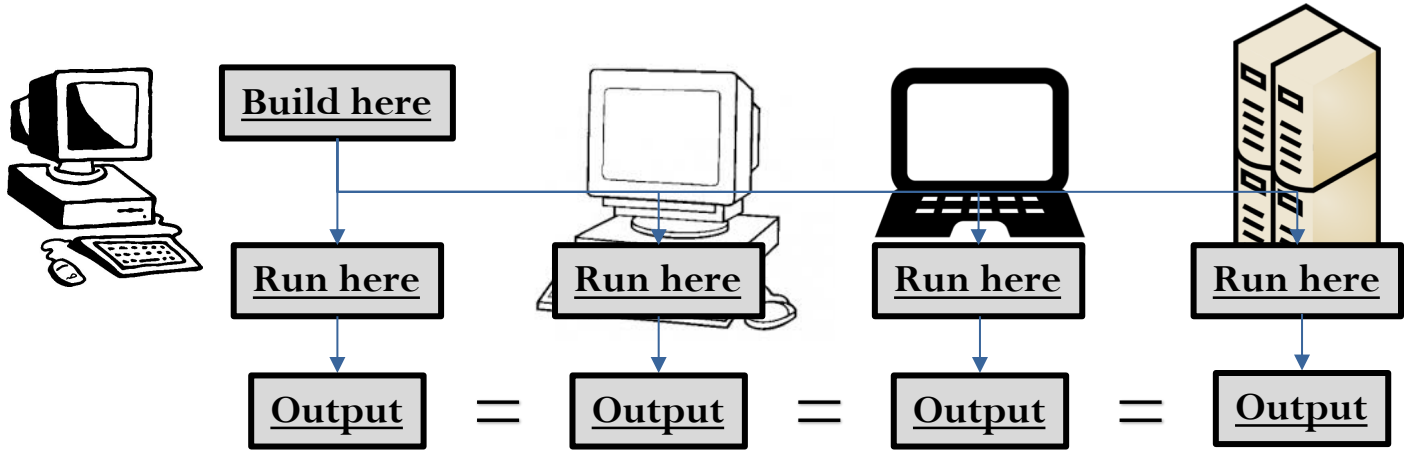
Container NWP Tutorial

https://dtcenter.org/met/container-nwp-tutorial/



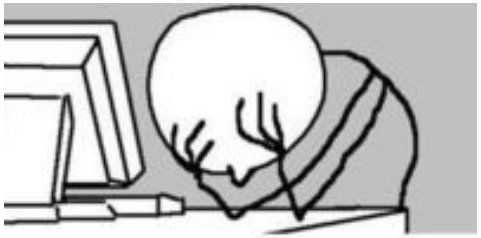
What is a container?

- A container is a self-contained "box" that allows you to essentially build software once and run it anywhere
- Similar to a "virtual machine", but much more lightweight and portable



Why use containers?

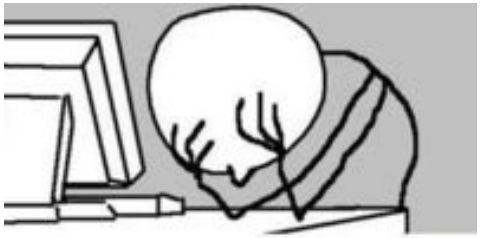
- Software systems require substantial set-up to get all the necessary code, including external libraries, compiled on a specific platform → ***Containers help solve this problem!***
  - Efficient, lightweight, secure, and self-contained (including operating system, libraries, code, and executables) systems
  - Everything required is packaged into isolated components, ready for deployment directly to users
  - Software runs the same, regardless of where it is deployed
  - Eliminates possible frustrations with up-front system setup
  - Facilitates cloud computing, classroom use, etc.



Someone trying to compile WRF

Why use containers?

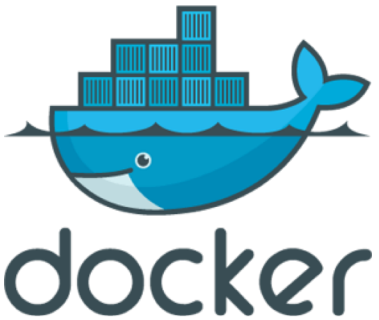
- The ability to build NWP software once and guarantee that it will be portable to other machines has many obvious use cases
  - Running NWP in a classroom or tutorial setting
  - NWP in the cloud
  - Development → Testing
  - Send code to collaborators with guaranteed reproducibility
  - Community support



Someone trying to *help someone else* compile WRF

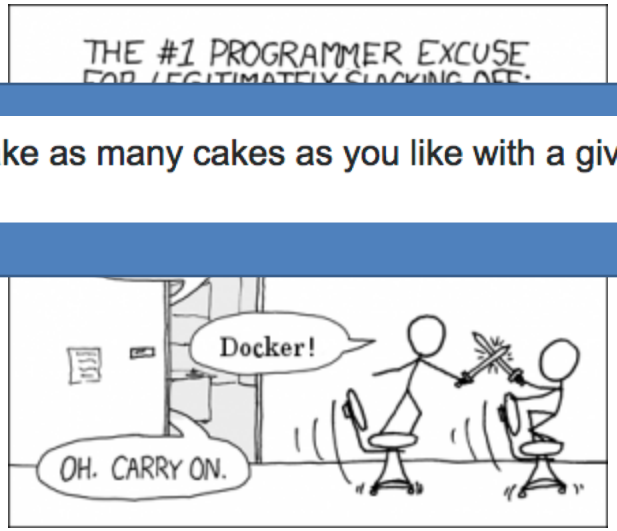
What is a Docker container?

- Docker is one of the leading software containerization platforms
  - Home page: **https://www.docker.com**
  - Documentation: **https://docs.docker.com**
- A Docker container
  - is open source
  - is an easy way to build a development environment
  - can hold applications "inside the container"
  - is portable across Linux, Mac, and Windows machines
  - is much smaller than a virtual machine
  - sets up a user-defined partition between the host machine and "container land"
  - allows "root" inside the container, but does not alter permissions on the host machine
  - ***requires root access to install Docker***
  - ***runs on a single node***



Images and Containers

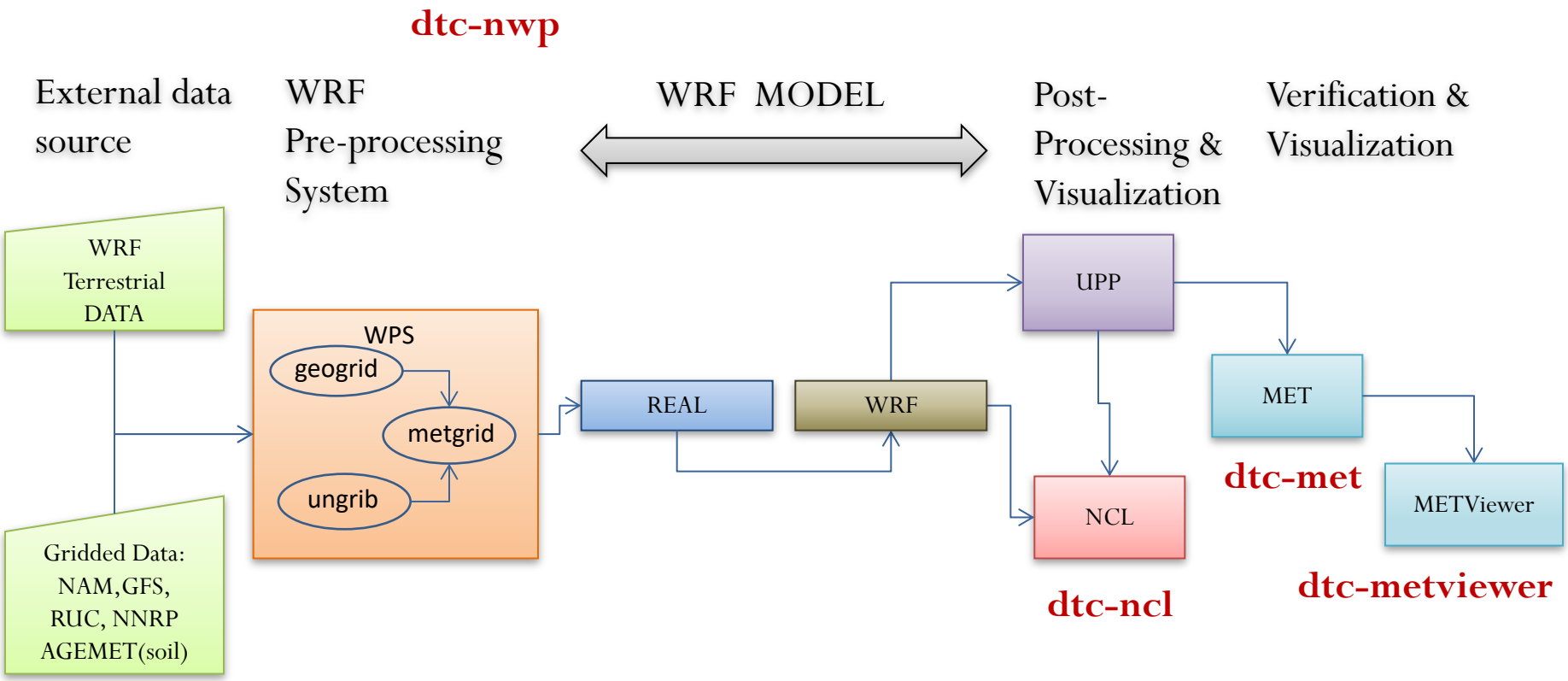
- Images can be…
  - Built from scratch with a *Dockerfile (slower, but offers customization!)*
    - ***docker build –t dtc-met .***
  - Saved to a tar file, which can then be loaded for faster deployment
    - ***docker load –i dtc-met.tar.gz***



54 the image is the recipe, the container is the cake ;-) you can make as many cakes as you like with a given recipe – Julien Sep 7 '17 at 8:25

  - The result of docker run
    - ***docker run -it dtc-met /bin/bash***
  - The result of docker create
    - ***docker create --name wps_geog dtc-nwp-wps_geog***

End-to-End NWP Workflow



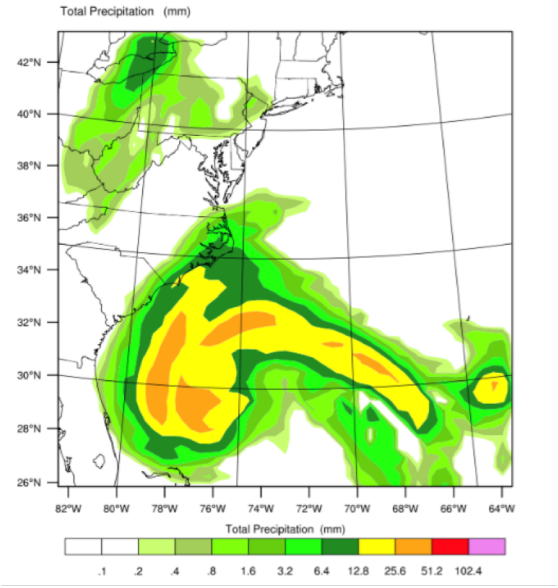
Scripts and Configuration

- Shell scripts required to run each NWP component
- Necessary namelist and configuration files
  - Vtable.GFS
  - namelist.wps and namelist.input
  - MET configuration files
- Case-specific data
  - GFS files for ICs/LBCs
  - Observation data for gridded (Stage II) and point (NDAS prepbufr) verification
  - Sample METViewer plot xml

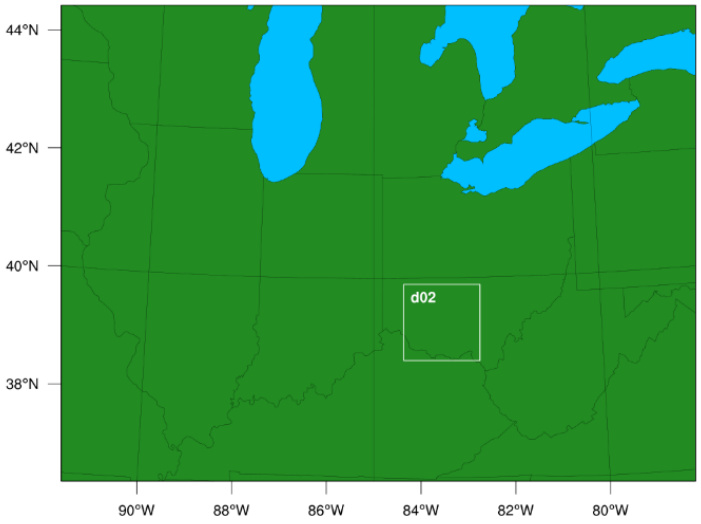
Sample Cases

- DTC containers package everything that is needed to build and run the model and produce verification, including code and data
  - Built with GNU compilers; can be run serially or with distributed memory
- Two cases with full datasets are provided in current inventory
  - Hurricane Sandy (Initialized on 27 Oct. 2012)
    - 40-km domain centered over East Coast (6-h forecast)
  - Derecho event over the Eastern CONUS (Initialized on 29 June 2012)
    - 12-km parent domain with 3-km nest over southern Ohio (24-h forecast)

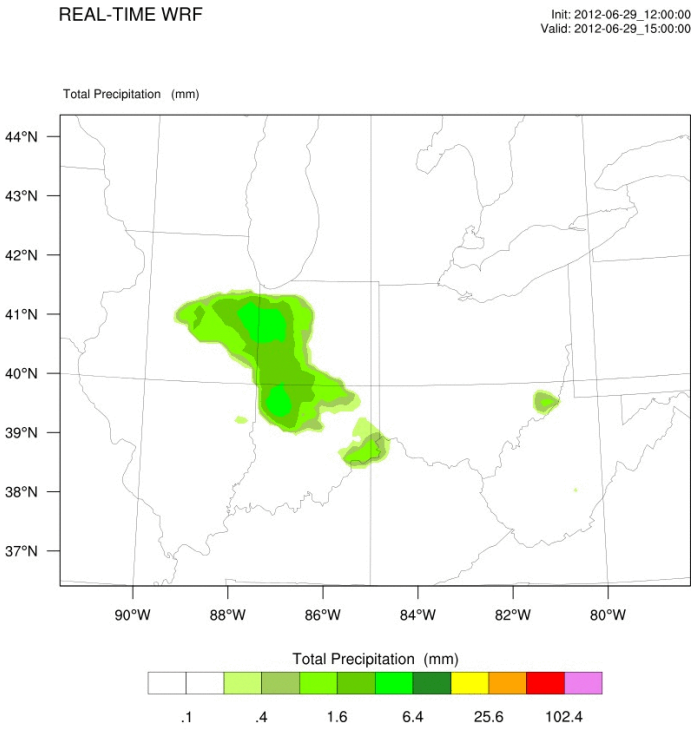
Total precip for Hurricane Sandy model forecast



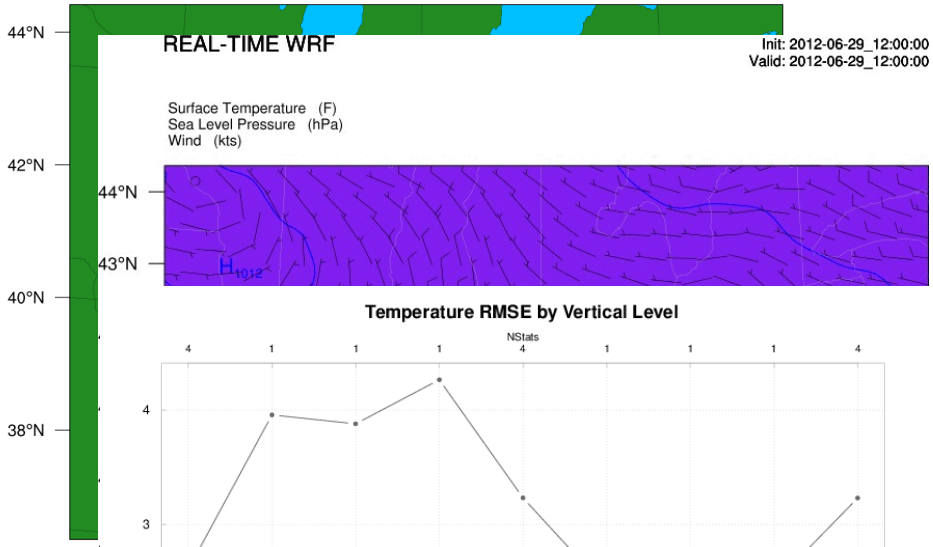
Derecho case domain configuration

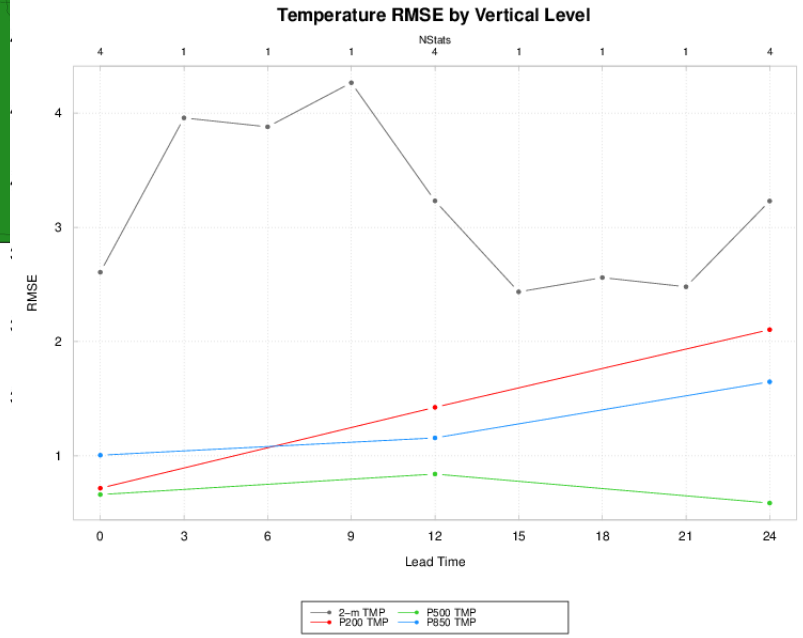


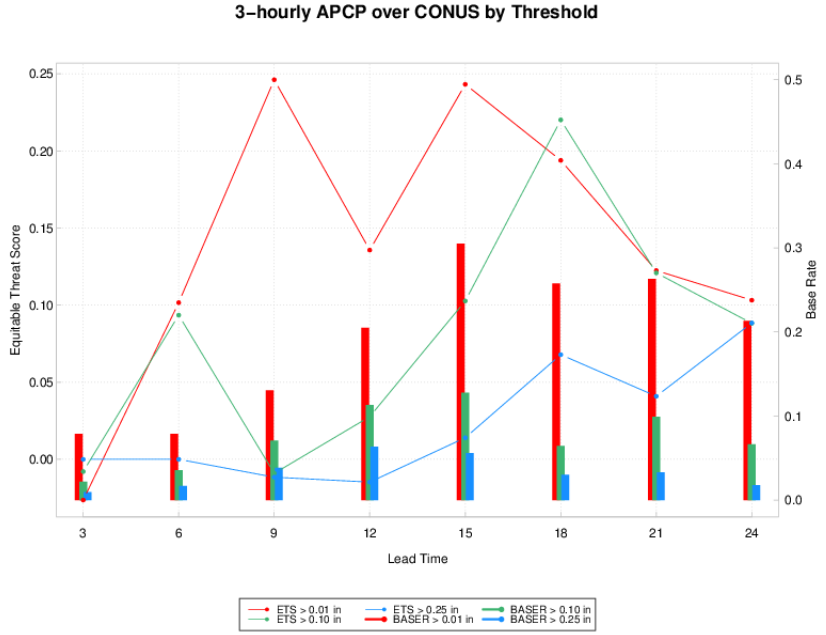
Derecho Case



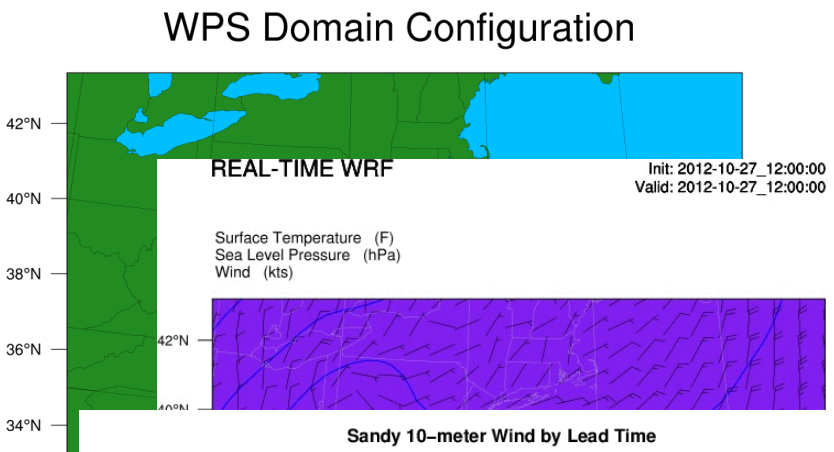
WPS Domain Configuration

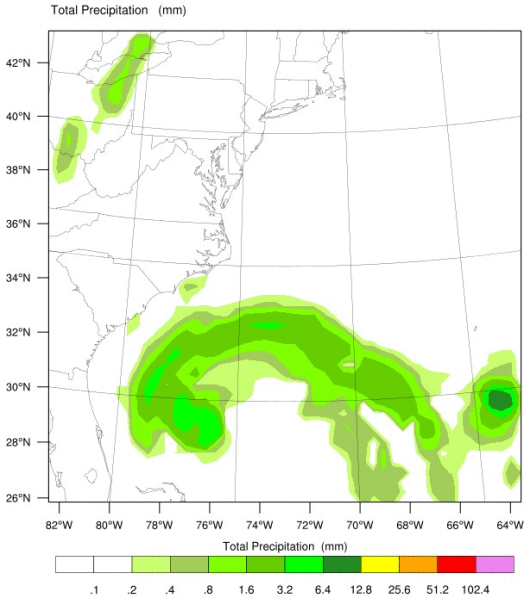


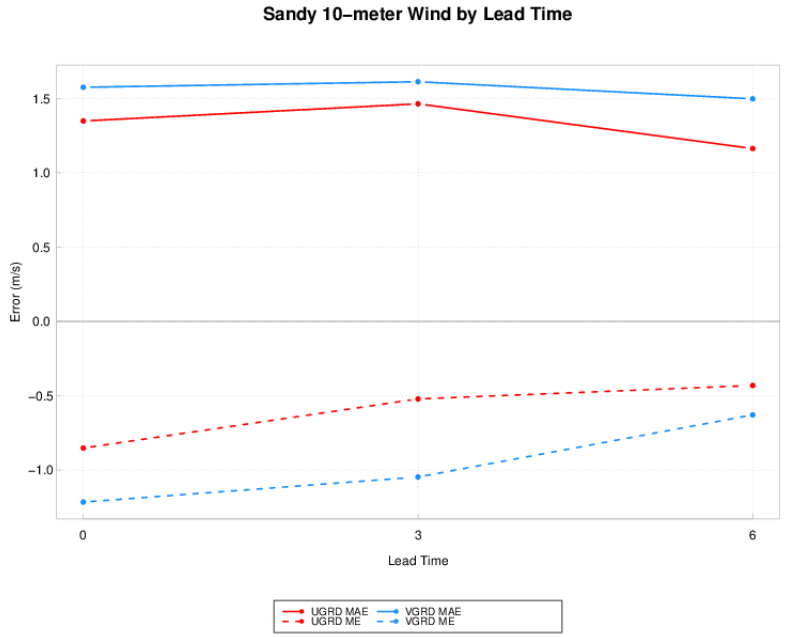


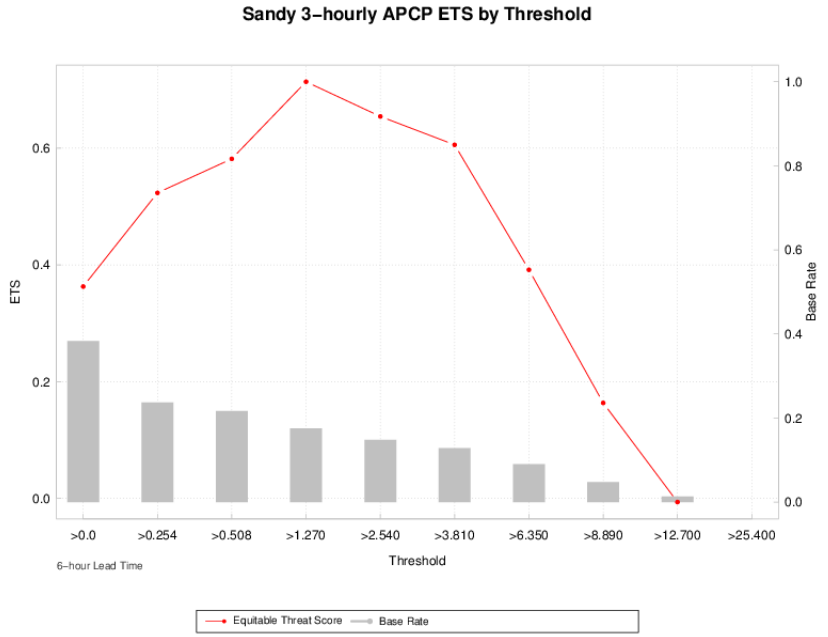


Sandy Case


WPS Domain Configuration
REAL-TIME WRF
Init: 2012-10-27_12:00:00
Valid: 2012-10-27_12:00:00
Surface Temperature (F)
Sea Level Pressure (hPa)
Wind (kts)
42°N
40°N
38°N
36°N
34°N
32°N
30°N
28°N
26°N


REAL-TIME WRF
Init: 2012-10-27_12:00:00
Valid: 2012-10-27_13:00:00
Total Precipitation (mm)
42°N
40°N
38°N
36°N
34°N
32°N
30°N
28°N
26°N
82°W
80°W
78°W
76°W
74°W
72°W
70°W
68°W
66°W
64°W
Total Precipitation (mm)
.1
.2
.4
.8
1.6
3.2
6.4
12.8
25.6
51.2
102.4


Sandy 10–meter Wind by Lead Time
Error (m/s)
1.5
1.0
0.5
0.0
−0.5
−1.0
0
3
6
Lead Time
UGRD MAE
UGRD ME
VGRD MAE
VGRD ME


Sandy 3–hourly APCP ETS by Threshold
ETS
Base Rate
0.6
0.4
0.2
0.0
1.0
0.8
0.6
0.4
0.2
0.0
>0.0
>0.254
>0.508
>1.270
>2.540
>3.810
>6.350
>8.890
>12.700
>25.400
Threshold
6–hour Lead Time
Equitable Threat Score
Base Rate

Docker disadvantages, Singularity solutions?

- Docker has some potentially significant disadvantages to consider
  - Requires root access to install and update Docker software
  - Problems running on more than one node
- A new container software "Singularity" sets out to eliminate these issues
  - Designed with HPC applications in mind
  - Has easy support for cross-node MPI applications
  - Does not require special permissions to install
  - Docker containers can be converted for use with Singularity

Future Work

- Expanded use for in-person tutorials
- Containerize Gridpoint Statistical Interpolation (GSI) data assimilation code
- Containerize UPP separately from WPS and WRF
- Seek projects for using containers in cloud computing
- Explore alternatives to Docker
  - Root access requirement is limiting
  - Issues mapping directories in Windows
- Future tutorials:
  - This Friday at 10:30
  - January 2019 AMS annual meeting

DTC Container Links

- **Container-DTC-NWP**
  - https://github.com/NCAR/container-dtc-nwp/
- **MET**
  - https://github.com/NCAR/container-dtc-met/
  - https://dtcenter.org/met/users/downloads/docker_container.php
    (Instructions for installing and running pre-built container)
- **METViewer (Containers for MySQL and METViewer)**
  - https://github.com/NCAR/container-dtc-metviewer/
- **End-to-end NWP container online tutorial**
  - https://dtcenter.org/met/container-nwp-tutorial/

Container NWP Tutorial

https://dtcenter.org/met/container-nwp-tutorial/

ABOUT TESTING & EVALUATION COMMUNITY CODES VISITOR PROGRAM EVENTS

END-TO-END NWP CONTAINERS ONLNE TUTORIAL

Home ▶
Introduction ▶
Repository ▶
Data Containers ▶
Software Containers ▶
Derecho Case ▶
Sandy Case ▶
Customization ▶

END-TO-END NUMERICAL WEATHER PREDICTION (NWP) CONTAINERS

NWP container components

This tutorial provides information on using software containers that have been established for community use to quickly spin up an NWP forecast system [using the Weather Research and Forecasting (WRF) model] that can then be post-processed [using the Unified Post Processor (UPP)] and verified [using the Model Evaluation Tools (MET)].

At the present time, the following components and versions of the code are containerized and detailed in this tutorial:

- WRF Preprocessing System (WPS) version 3.9.1
- Advanced Research Weather Research and Forecasting (WRF-ARW) model version 3.9.1.1
- Unified Post Processor (UPP) version 3.2
- NCAR Command Line (NCL) graphics
- Model Evaluation Tools (MET) version 6.1
- METViewer database and display version 2.3

Thank you

This online tutorial describes step-by-step instructions on how to obtain, build, and run each containerized component using Docker.

Why use containers for NWP?

Advantages:

- ✓ Reduces spin-up time to build necessary code components
- ✓ Highly portable
- ✓ Use in cloud computing
- ✓ Easily sharable with other collaborators
- ✓ Easy to replicate procedures and results

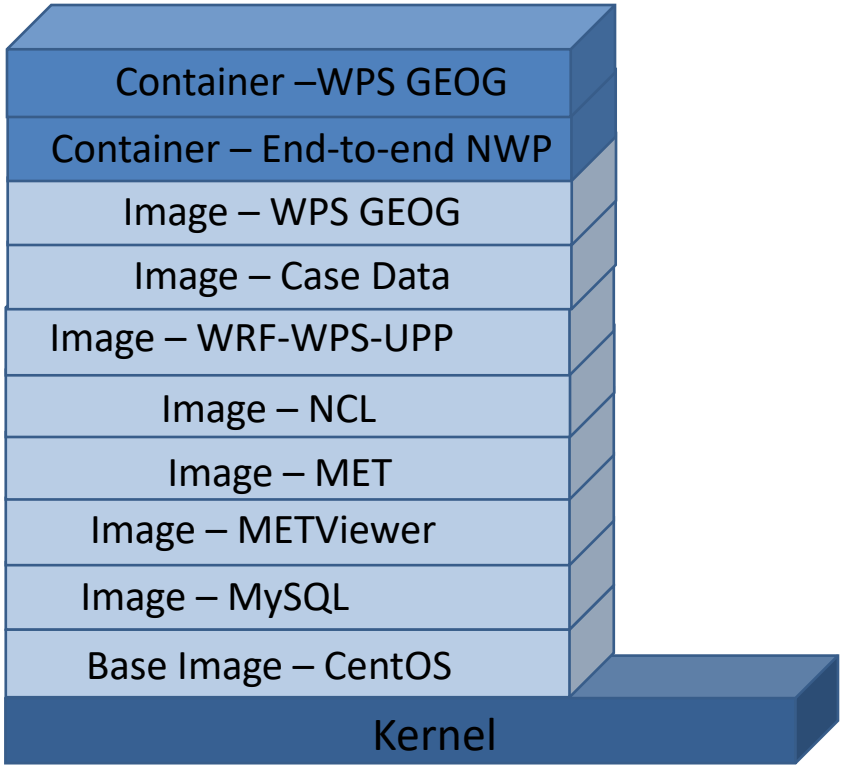
Who can benefit from containers?

- ✓ Graduate and undergrad students
- ✓ University faculty
- ✓ Researchers
- ✓ Tutorial participants

Ultimately, containers will substantially reduce the spin-up time with setting up software systems, which promotes greater efficiency in producing model and statistical output!

What does this look like for the end-to-end DTC container?

Contains necessary build and run commands to run end-to-end NWP workflow




```
[nitro:~/container-dtc-nwp/components] jwolff% docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
[nitro:~] jwolff% docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
4b96a3a0d0b7        dtc-nwp-derecho     "true"              2 minutes ago       Created                                 derecho
856064629ea5        dtc-nwp-wps_geog    "true"              3 minutes ago       Created                                 wps_geog
dtc-nwp-derecho     latest              8f7ff6e1e5ca        About an hour ago   773MB
dtc-nwp-wps_geog    latest              869b67961d1f        About an hour ago   16.5GB
debian              jessie              25fc9eb3417f        4 days ago          123MB
centos              latest              d123f4e55e12        4 days ago          197MB
```