Enhancing community collaborations through NWP software containers

Michelle Harrold[1,2], Jamie Wolff[1,2], John Halley Gotway[1,2], Kathryn Fossell[1,2], and John Exby[1]

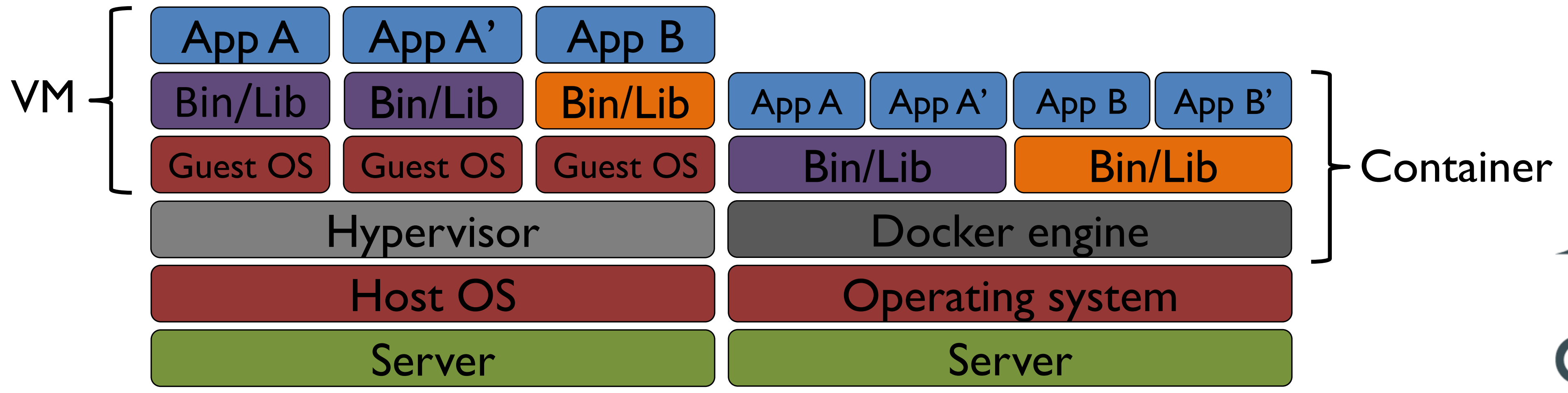
Developmental Testbed Center

1) National Center for Atmospheric Research

2) Developmental Testbed Center

What are containers?

- Software systems require substantial set-up to get all the necessary code, including external libraries, compiled on a specific platform. Recently, the concept of containers has been gaining popularity because they allow for software systems to be bundled (including operating system, libraries, code, and executables) and provided directly to users, eliminating possible frustrations with up-front system setup
- Using containers allows for efficient, lightweight, secure, and self-contained systems
  - Everything required to make a piece of software run is packaged into isolated containers, ready for development, shipment, and deployment
  - Guarantees that software will always run the same, regardless of where it is deployed
- Containers vs. virtual machines (VMs): VMs bundle a full operating system, whereas containers only contain necessary libraries and dependencies

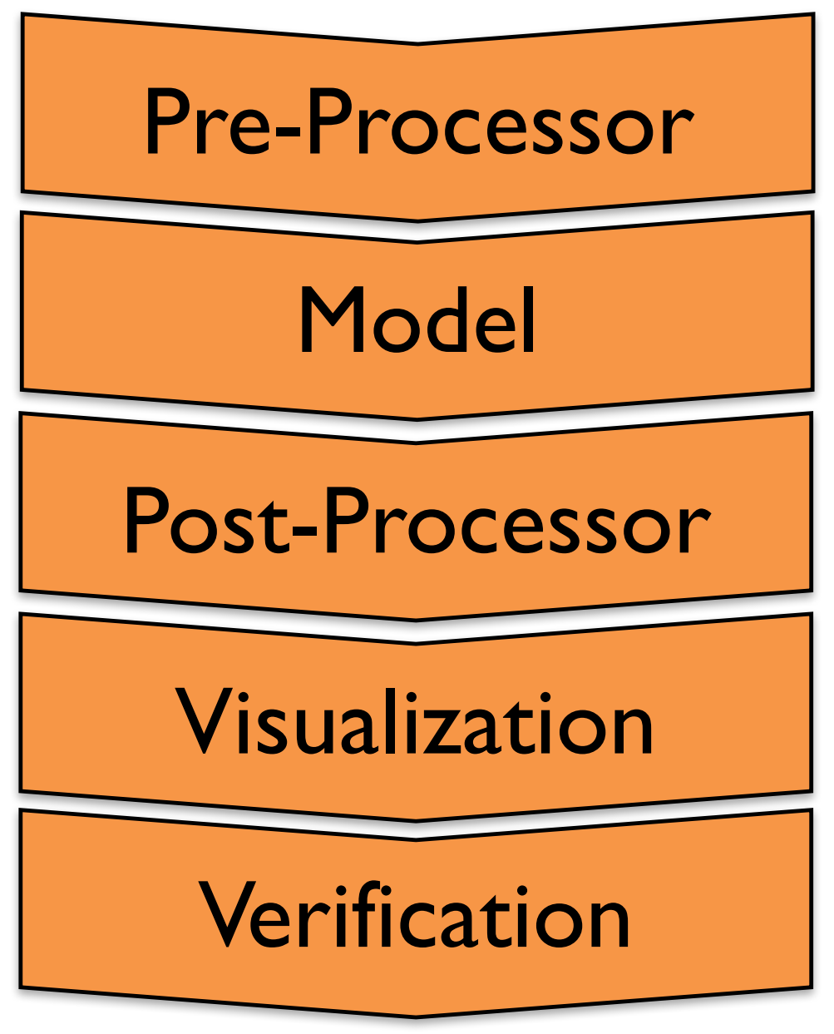


- Docker is one of the leading software packages that deploys applications inside software containers

Why use containers in NWP?

- A frequent stumbling block when first running an end-to-end modeling system is properly setting up and compiling all of the necessary code components, including the forecast model, pre- and post-processors, visualization, and verification → Containers help solve this problem!
- Who is interested in containers?
  - Graduate and undergraduate students
  - University faculty
  - Researchers
  - Tutorial participants
- Why NWP in containers?
  - Highly portable
  - Use in cloud computing
  - Easily sharable with other collaborators
  - Easy to replicate procedures and results (even if not working on the same platform!)
- Ultimately, containers will substantially reduce the spin-up time with setting up and compiling software systems and promote greater efficiency getting to the end goal of producing model output and statistics more quickly

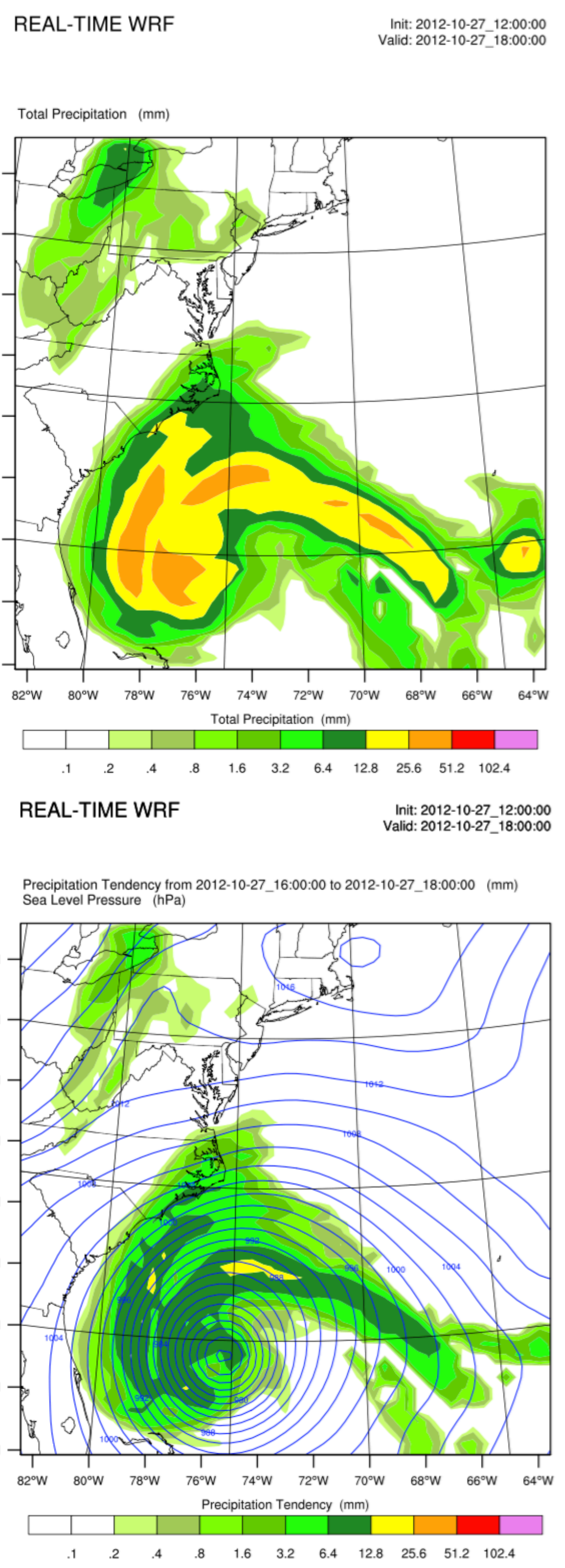
Typical end-to-end workflow



Acknowledgments: The DTC is funded by the National Oceanic and Atmospheric Administration, the Air Force, and the National Center for Atmospheric Research (NCAR). NCAR is sponsored by the National Science Foundation (NSF).

Running an end-to-end NWP system in a container

- The Weather Research and Forecasting (WRF) Model, WRF Preprocessing System (WPS), and the NCAR Command Language (NCL) have been implemented in Docker containers (J. Exby, K. Fossell, and J. Hacker, NCAR)
  - **https://github.com/NCAR/container-wrf/**
- Leveraging the work above, the DTC implemented the Unified Post-Processor (UPP) and the Model Evaluation Tools (MET) into containers to complete the end-to-end system
  - **https://github.com/NCAR/container-dtc-nwp/**
- To better learn MET, a standalone container for running the online MET tutorial is also available
  - **https://github.com/NCAR/container-dtc-met/**
- The containers provided by the DTC package everything that is needed to build and run the model and produce verification, including code and data
  - Uses CentOS and gfortran
  - Can be run on Mac, Linux or Windows machines
  - Can be run serially or with distributed memory
- Two cases with full datasets are provided
  - Hurricane Sandy (Initialized on 27 Oct. 2012)
    - ✓ 40-km domain centered over East Coast (6-h simulation)
  - Derecho event over the Eastern CONUS (Initialized on 29 June 2012)
    - ✓ 12-km parent domain with 3-km nest over Eastern CONUS (24-h simulation)
- What are in the DTC containers?
  - README files for building Docker images and running code components
  - Run scripts for WPS, WRF, UPP, and MET
  - Necessary parameter files
    - ✓ Vtable.GFS
    - ✓ namelist.wps and namelist.input
    - ✓ MET configuration files
  - Case-specific data:
    - ✓ GFS initial conditions
    - ✓ Observation data for point and gridded verification



Above: Total precipitation at the end for Hurricane Sandy model simulation
Below: Precipitation tendency and SLP at the last output time for Hurricane Sandy model simulation

Future Activities

- Containerize METViewer, a database and display system for MET
- Add additional cases and functionality
- Enhance NCL plotting capabilities
- Anticipated short course offered at AMS Annual Meeting in Austin, TX in 2018

Additional questions?

Contact information:

John Exby – exby@ucar.edu
Kate Fossell – fossell@ucar.edu
John Halley Gotway – johnhg@ucar.edu
Michelle Harrold – harrold@ucar.edu
Jamie Wolff – jwolff@ucar.edu