# High Performance Extreme Computing, Data Processing, and Visualization

Si Liu

Software Engineering Assembly 2014

Apr 7-11, 2014

# Team members:

TACC

Greg Abram
John Cazes
Greg Foss
Si Liu

Raytheon

Don Cook
Craig Stair

NCAR
NATIONAL CENTER FOR ATMOSPHERIC RESEARCH

Dave Gill
Jordan Powers

# Outline

- Objectives
- WRF nested runs
- Memory requirement
- IO efficiency and workflow
- Data analysis and visualization
- Conclusion and future work

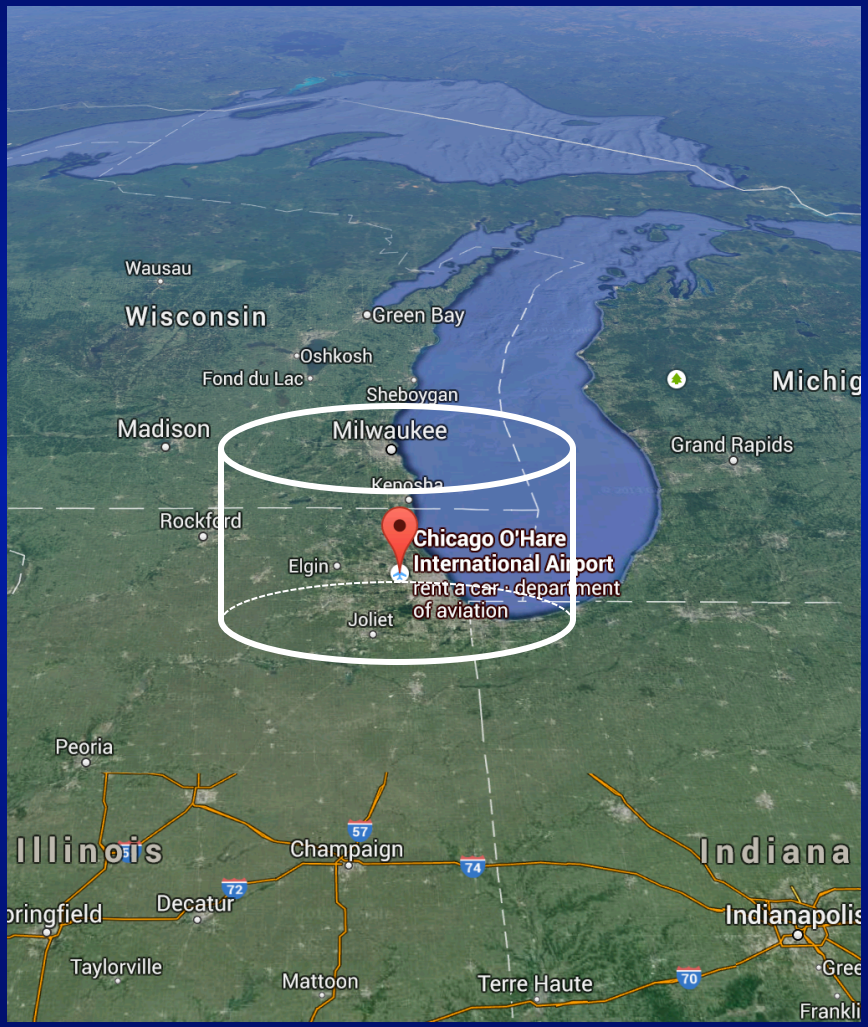
# Objectives

Raytheon R&D project

- Highly localized weather modeling
- Weather simulation for several hours
  - extremely high spatial resolution
  - extremely high temporal resolution
- High temporal resolution data collection for animated demonstration

# Domain of Interest

- Around O'Hare International Airport, Chicago, Illinois
  - Longitude: 87.9047W
  - Latitude: 41.9786N
- Cover the cylinder area
  - Diameter: 120 nautical miles
    About 222 km
  - Height: 70,000 feet
    About 21 km
- The expected resolution:
  - Horizontal: 167 meter or finer
  - Vertical: 300 feet or finer
    About 91 meters



# WRF

## Weather Research and Forecasting Model

- Open source community software developed and supported by NCAR and collaborative partners
- Parallel mesoscale weather model
- Used for both research and operational forecasts
- A large worldwide community of users (over 20,000 in over 130 countries)

# WRF Nested Runs

- A fine-grid run based on the parent coarse-grid run
- Cover only a portion of the parent domain
- Lateral boundaries driven from the parent domain
- Why nested runs:
  - High resolution model run over the large domain is prohibitively expensive (memory, storage, computing)
  - High resolution for a very small domain with mismatched time and spatial lateral boundary conditions
  - Other reasons

# Two-way nested runs

Run parent and child (nested) domains in a single simulation with information exchange between them

- Parent simulation: 500m horizontal
- Child (nested) simulation:167m horizontal
- Still expensive (at least doubles the problem size we finally solve in every single run)

# One-way Nested Runs

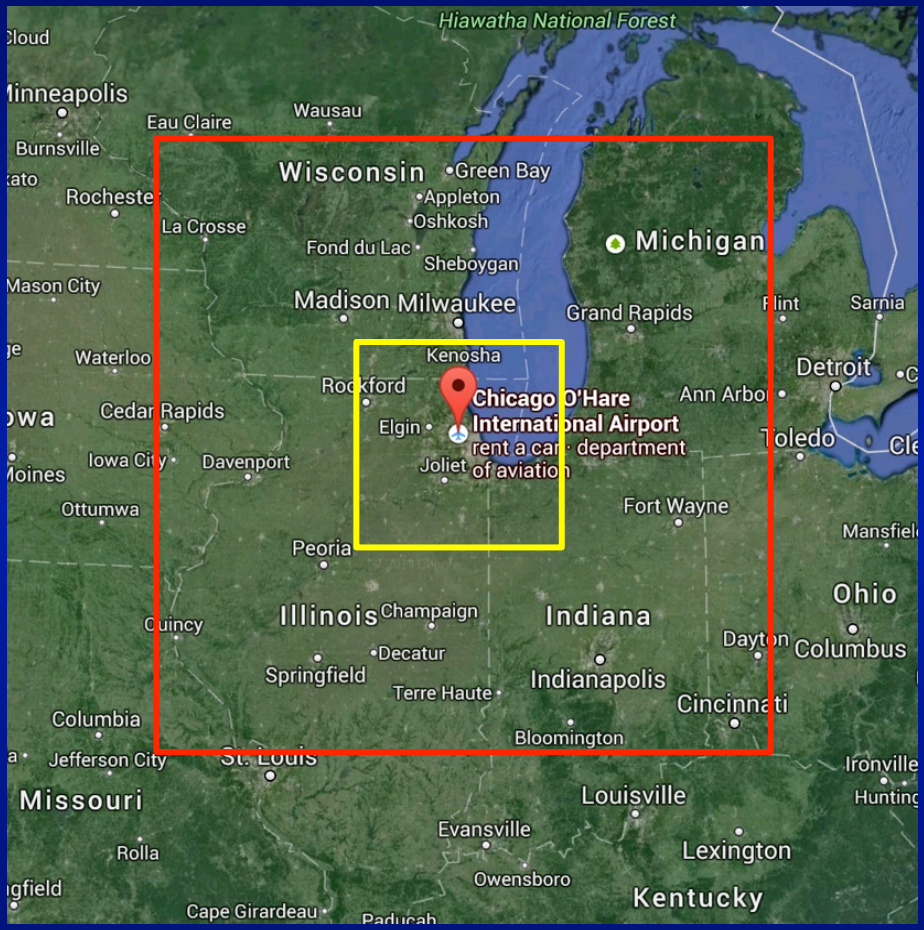
- Achieved with ndown.exe program
- A finer-grid run is made as a subsequent run after the coarser-grid run
  - Make a complete coarse-grid run (500m horizontal) and collect output data
  - Create the initial and lateral boundary conditions for the fine-grid run with the WRF ndown.exe program
  - Run the fine-grid simulation (167m horizontal) with the input files generated in the previous step

# Vertical refinement

- Achieved through ndown.exe
- Original plan:
  100 vertical levels (parent domain)
  → around 200/300/500 vertical levels (child domain)
- Vertical refinement is yet limited in WRFV-3.4.* and 3.5.*.
- Current implementation:
  234 vertical levels (parent domain)
  → 234 vertical levels (child domain)

# Nested Domains Sketch Map

- Outer domain
  - 1345 x 1345 grid cells (500 m)
  - 234 vertical level (300 feet)
- Inner domain
  - 1345x1345 grid cells (167 m)
  - 234 vertical levels (300 feet )
- Nested ratio
  - Horizontal 3:1
  - Vertical 1:1



# WRF Workflow

- Obtain the Global Forecast System (GFS) model data
- Run geogrid.exe, ungrib.exe, and metgrid.exe in WRF Preprocessing Systems (WPS)
- Run real.exe to generate the initial and lateral boundary condition files for the coarse-grid run
- Make a coarse-grid run (only a few output files are necessary)
- Re-run geogrid.exe and metgrid.exe for both parent and nested domains
- Re-run real.exe for both parent and nested domains
- Execute ndown.exe to generate the fine-grid initial and lateral boundary conditions
- Make the fine-grid run and produce output files frequently as required

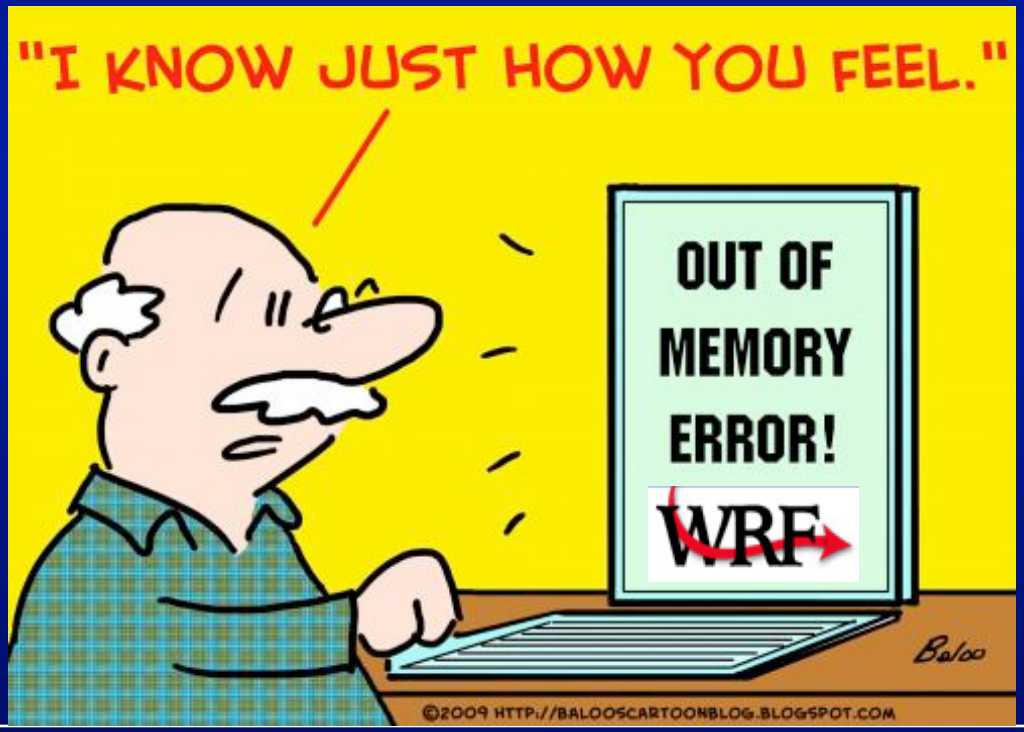
# TACC Stampede System

- Dell Linux Cluster
- 6,400+ Dell PowerEdge server nodes
  - 2 Intel Xeon E5 (Sandy Bridge) processors
  - 1 Intel Xeon Phi Coprocessor (MIC Architecture)
- The aggregate peak performance
  - Xeon E5 processors: 2+ PF; Xeon Phi processors: 7+ PF
- Login nodes, large-memory nodes, graphics nodes
- Global parallel Lustre file system

TACC

# WRF OOM Crisis

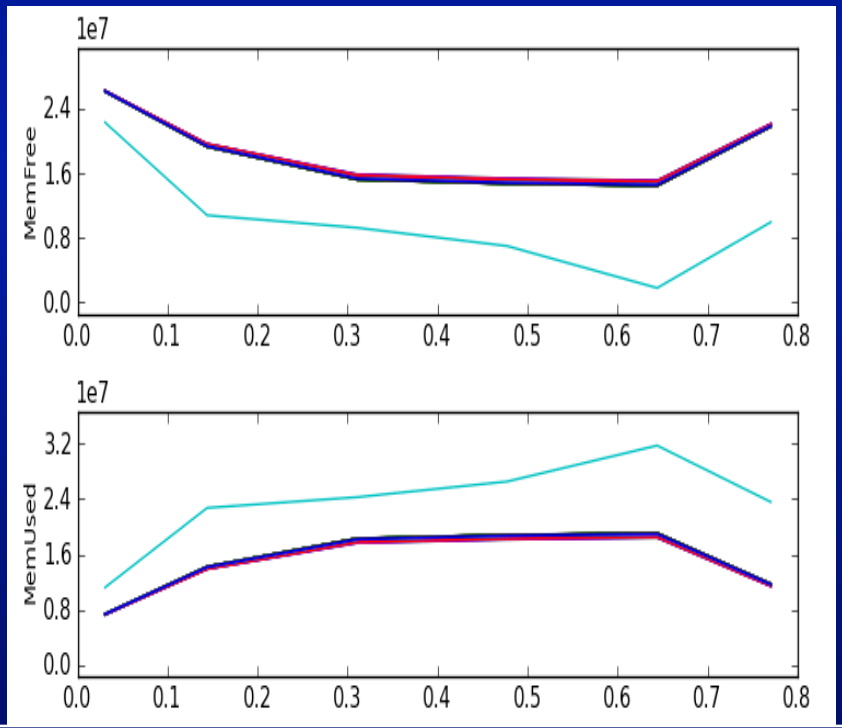
- OOM is a normal problem to such high-resolution simulations. WRF is no exception.
- WRF has not supported vertical partitions yet.



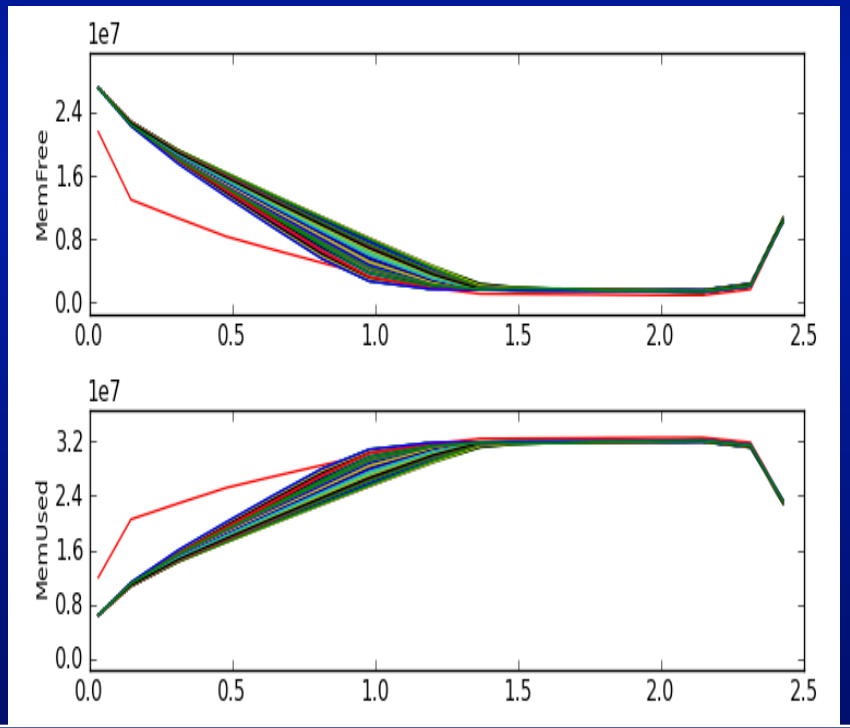
# Monitor WRF Memory Usage

- Each task needs a huge amount of memory.
- Task zero may require more memories than others.
- TACC Stats: http://tacc-stats.tacc.utexas.edu/

**ndown.exe**

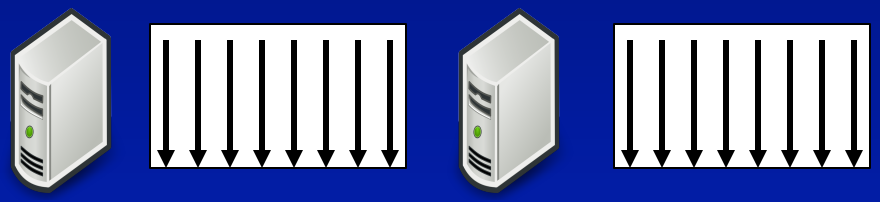



**wrf.exe**

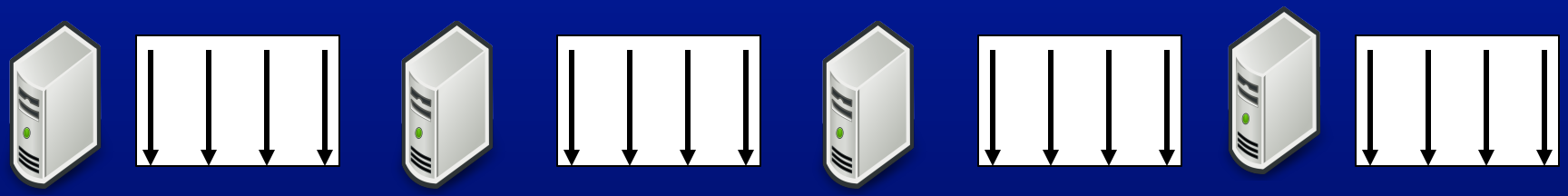


# How to Meet the Memory Requirement (1)

- Original settings

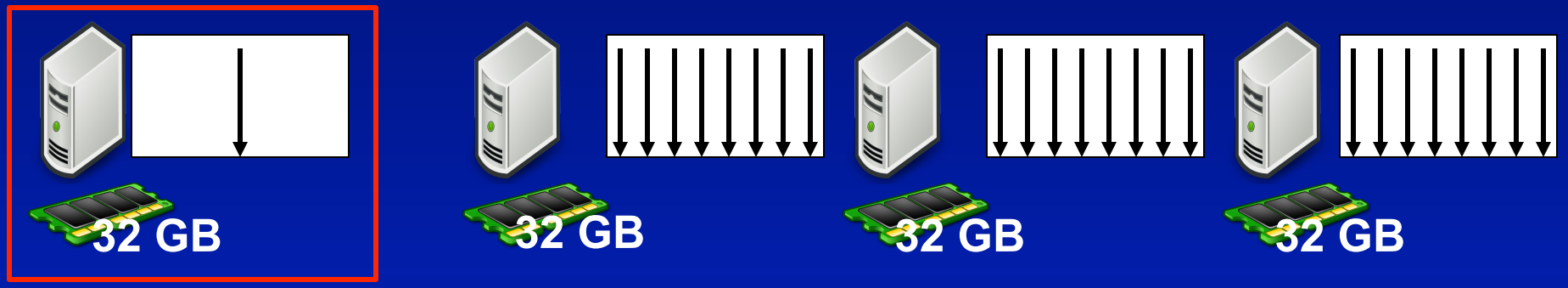


- Fewer MPI tasks per node

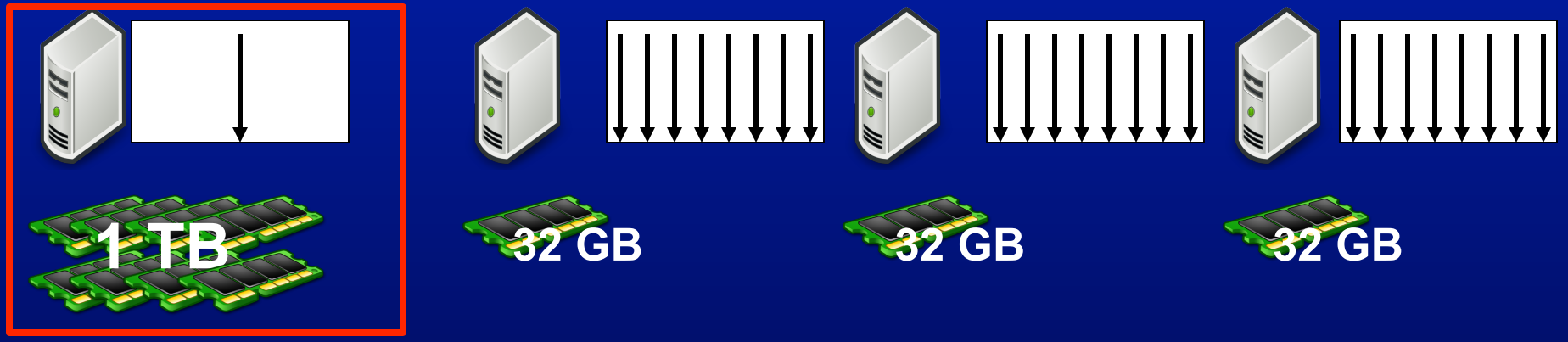


# How to Meet the Memory Requirement (2)

- SLURM reconfiguration is necessary.
- One dedicated node for Task 0




- One dedicated large-memory node for Task 0




TACC

# IO Workload

- Each output file is huge:
  - More than 400 million grid points
  - About 200 variables
  - 11+ GB per file
- Record the output every 3 model seconds:
  - Generate 20 files per model minutes, 1200 files per model hour
- Serial IO and Parallel IO
  - Wasting a lot of computing resources
  - Slow down the whole system (so many IO requests)

# IO Techniques

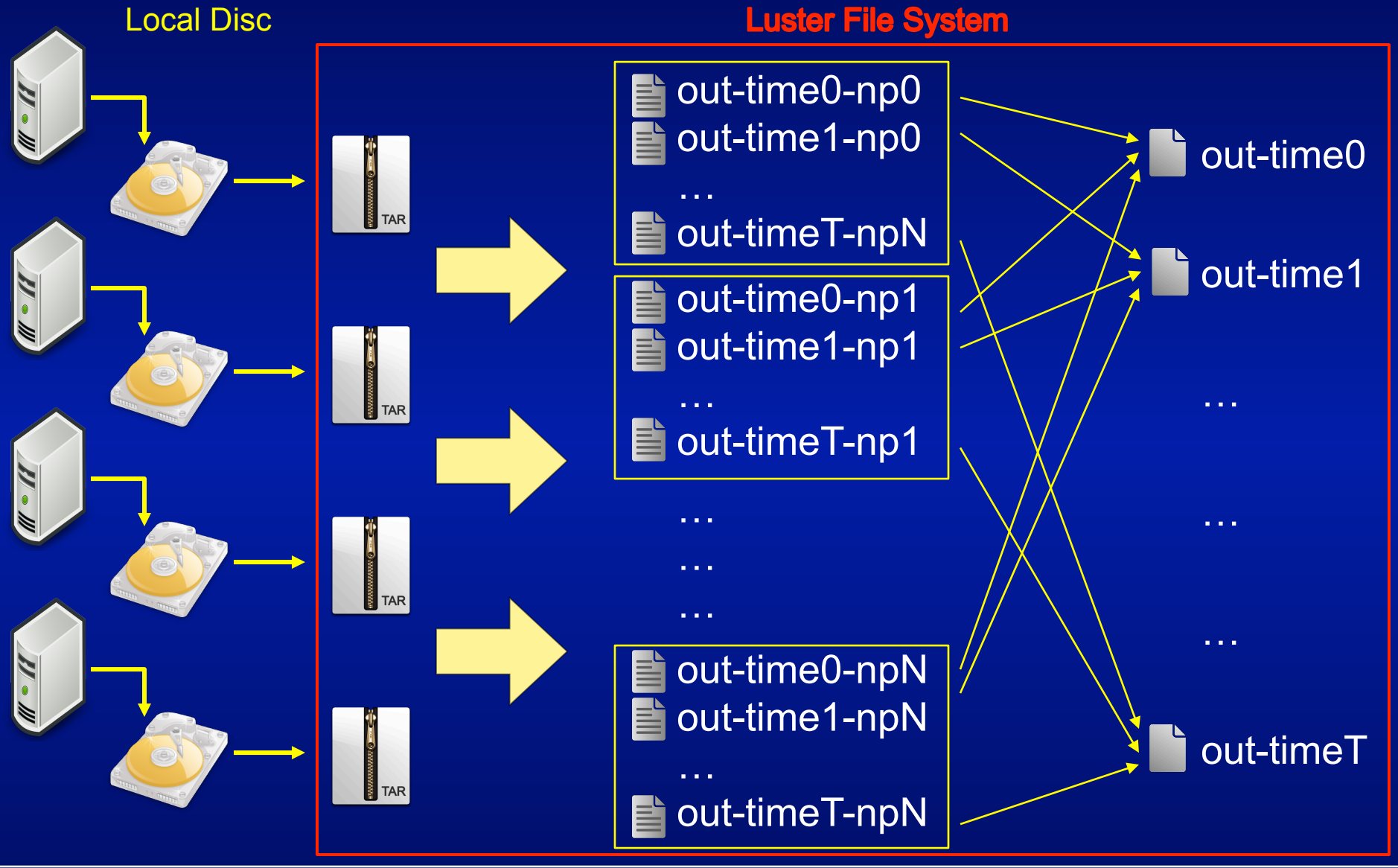
- Restrict output variables
  - Modified Registry.EM_COMMON (WRF recompiled)
  - Cut the output file size by 30-50%
- WRF checkpoint and restart mechanism
  - Make job complete within the wallclock limit
  - Reduce the risk of job failure and data loss
  - Validate the output data after every single run
- Use local disk to temporarily keep the output
  - Local /tmp space (68-70GB available disk space)
- WRF output files and WRF restart files partition
  - About 10 minutes → about 0.5s per output step

# Merge split WRF output

- Regroup split WRF output files
  - Task-based → Time-step-based
  - Several “tar/untar” work to reduce the Metadata Server workload of the Lustre file system
- Merge split WRF output
  - Advanced Regional Prediction System (ARPS)
  - Thousands of sequential jobs
  - Large-memory node
  - TACC Parametric Job Launcher Utility

    utility for submitting multiple serial applications simultaneously

# IO Workflow


Local Disc
Luster File System
TAR
TAR
TAR
TAR
out-time0-np0
out-time1-np0
…
out-timeT-npN
out-time0-np1
out-time1-np1
…
out-timeT-np1
…
…
…
out-time0-npN
out-time1-npN
…
out-timeT-npN
out-time0
out-time1
…
…
…
out-timeT

# IO Cost Summary

| | Traditional IO workflow Method | Our IO workflow |
|---|---|---|
| Time per step | 10 minutes | 0.4-0.5s on average (1024 cores) |
| Total Time | more than 8 days | About 8-10 minutes |
| Extra time for data processing | 0 | 8-10 hours depending on the computing resources |
| Target data files | 1201 wrf-out files, 11 GB each 13TB in total for one-hour | 1201 wrf-out files, 11GB each 13TB in total for one-hour run |
| Extra space required | 0 | 256 tar ball files, 36GB each 10TB in total |

# Data Analysis and Visualization

- WRF output uses pressure values to identify altitude whereas visualization software requires coordinate values in the height axis
- Translate WRF output files to VTK files (Python)
  - converting  pressure values into Z coordinates
  - irregular grid
- Resulting VTK files are read into ParaView
- Target variables are modeled as a surface
- For a generalized aviation reference, an aviation map provided by Raytheon is included for background in the animation

# Achievements and Conclusions

- We model a specific time frame and region to provide meteorological data with high spatial and temporal resolution.
- The resolution in time and space is well beyond what has ever been attempted with previous WRF simulations.
- The memory management techniques are portable to other programs and other platforms.
- The new-designed IO workflow is more convenient and efficient.

# Future work

- Compare with other experimental data for validation
- A follow-on study will be performed over the other domains of interests
- Even-higher resolution simulations if necessary
- Understand the WRF design and exceed the limit of WRF settings
- Investigate optimized IO workflow with T3PIO library
- Improving WRF performance with Xeon Phi

# Acknowledgement

Special thanks to

- Ming Chen (NCAR)
- Bill Barth, Tommy Minyard, Todd Evans (TACC)

TACC

Si Liu

siliu@tacc.utexas.edu

(512)-471-0958

For more information:

www.tacc.utexas.edu


TACC