7th Workshop on Distributed Supercomputing – Durango, CO

Weather Forecasting on Distributed Memory Supercomputers

Leslie Hart
NOAA/OAR
Forecast Systems Laboratory
High Performance Computing Systems
Boulder, CO

March 4-6, 2003

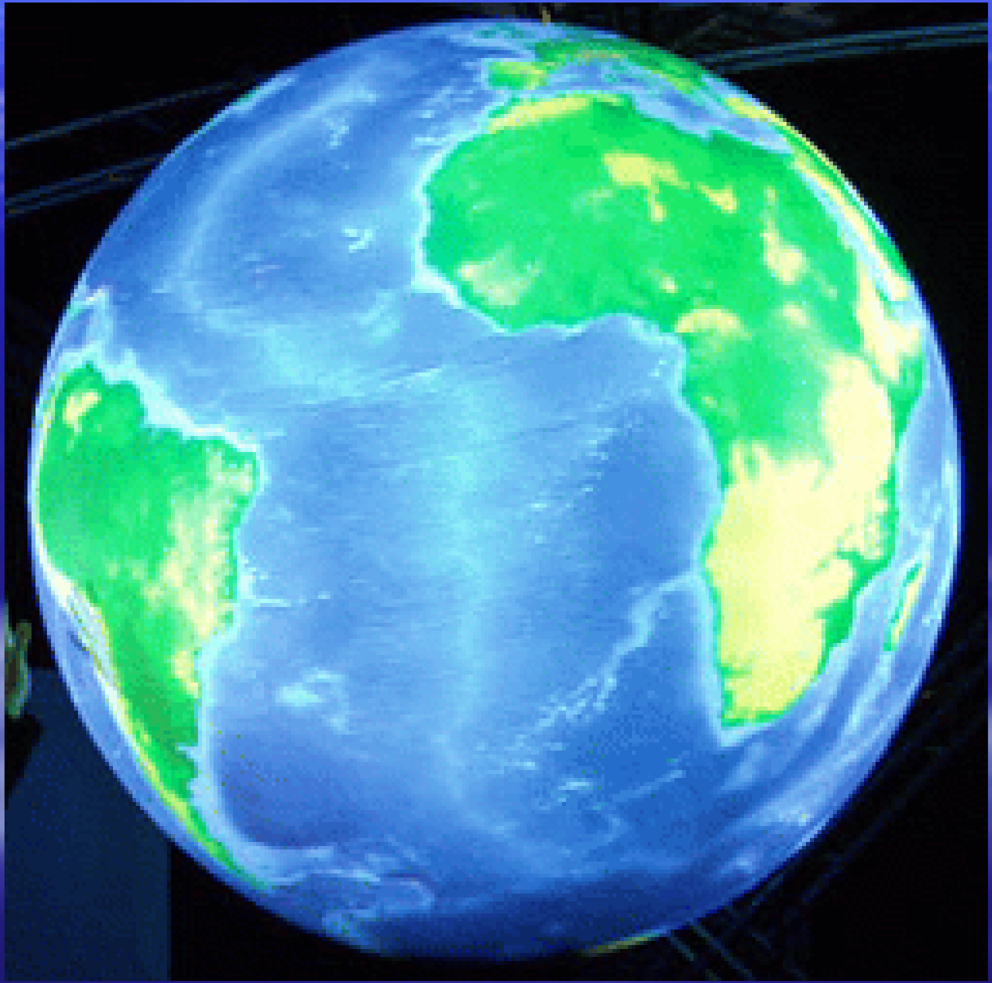
Environmental Prediction

- Climate
  - Decades – Centuries
- Seasonal-Interannual
  - El Nino and similar effects
- Short-term weather forecasting
  - Hours – 2 weeks
  - Global
  - CONUS
  - Local
  - Hurricane
- Ocean
- Solar
- Development

Weather Forecasting

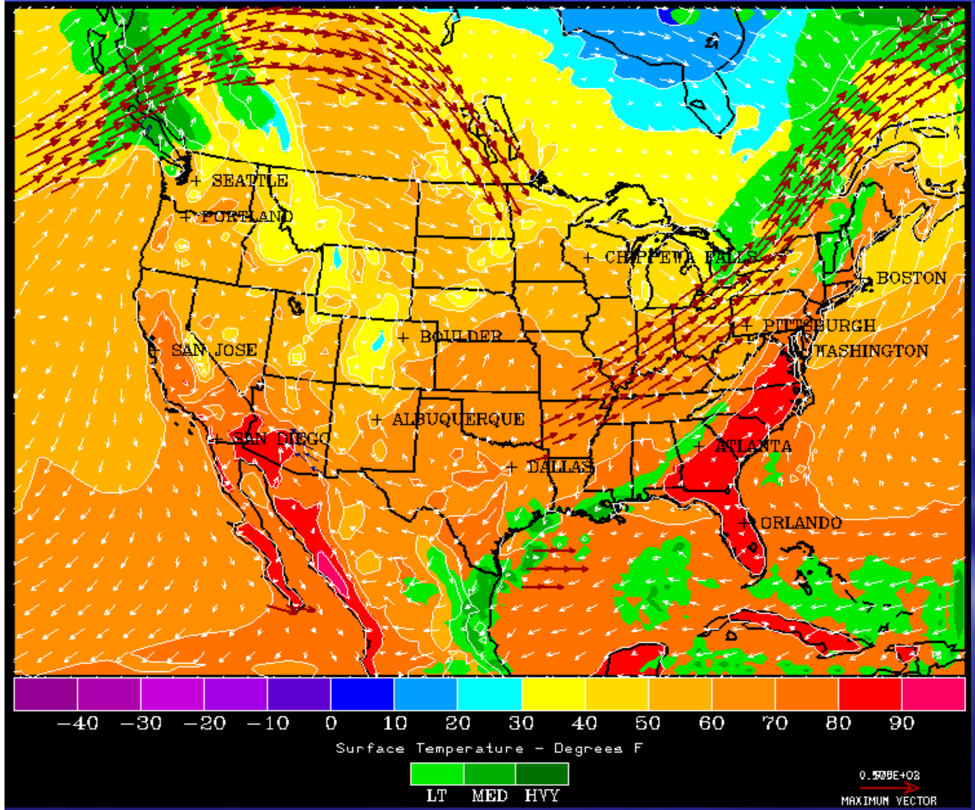
- Observing Systems
  - Baloon data (twice per day, thousands of sites)
  - Automated Surface Stations (ASOS)
  - ACARS (Aircraft instrumentations)
  - GPS-IPW
  - Satellite Data
  - Little information over the oceans

Weather Forecasting Global



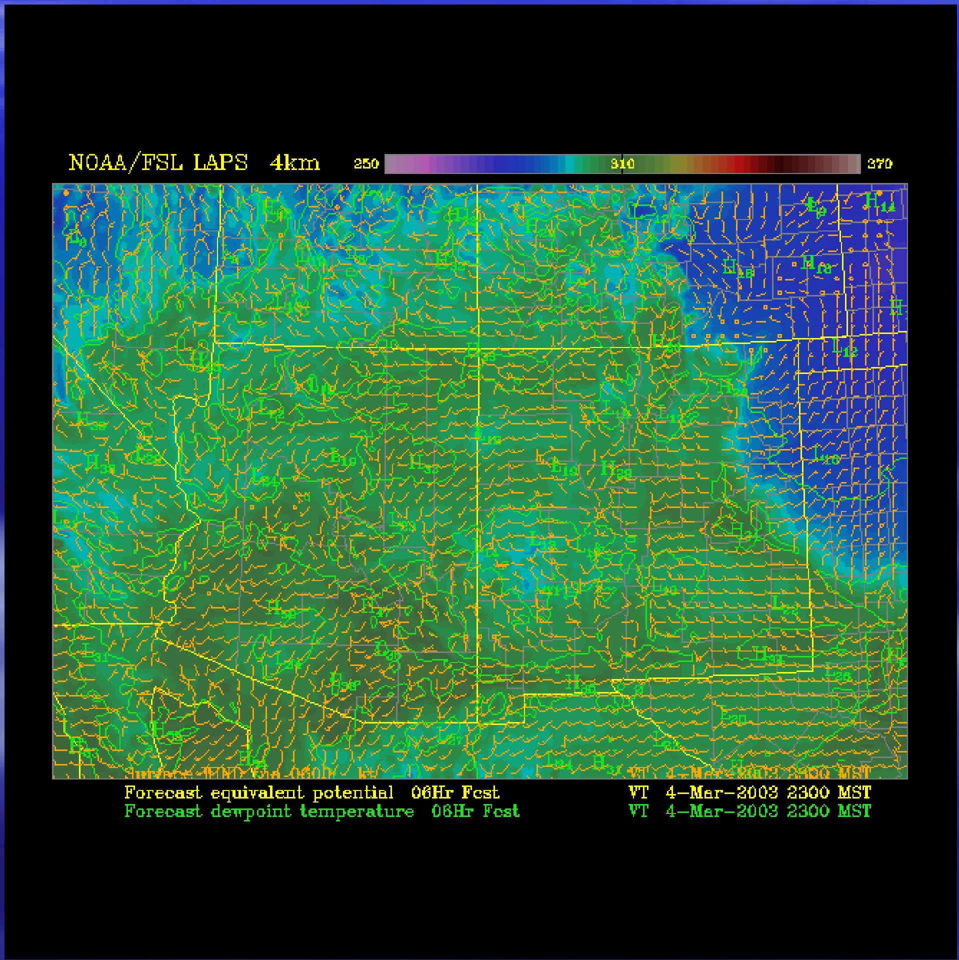
- Composed of FFT and Legendre Transforms.
  - Usually done by transposing data and performing transforms on node.
- Usually a week to two week forecasts
- (Very) Similar in structure to Climate models
- Used as boundary conditions for CONUS models

Weather Forecasting CONUS



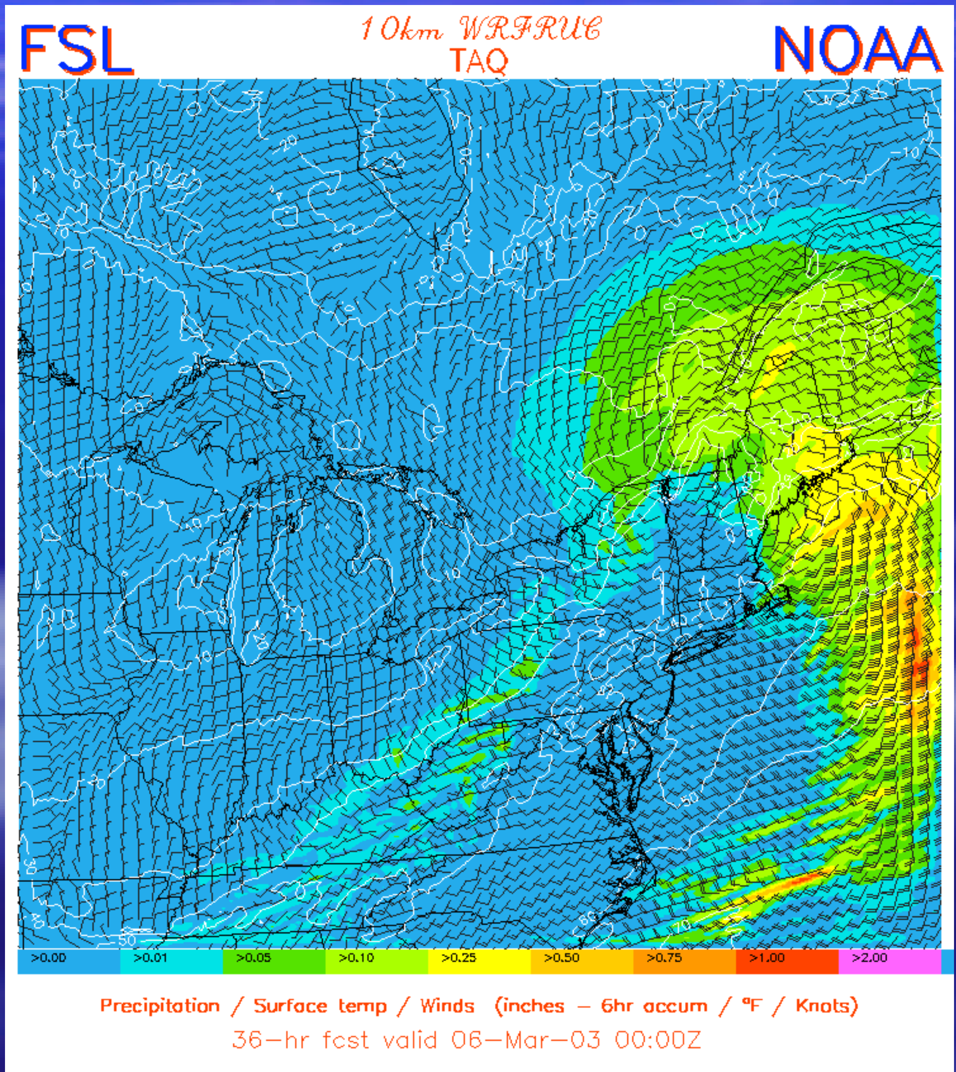
- Usually finite difference approximation models
  - Parallelized with domain decomposition
- Forecast duration from hours to less than two weeks
- Currently 12-20KM
  - CONUS is around 6000x4000KM
  - 2-4GBytes output per run @20KM
- Runtime 30 minutes to 2 hours
- Computational complexity goes up by a factor of 10 as resolution doubles.
  - ***Assuming no new physics***

Weather Forecasting Local



- Usually finite difference approximation models
  - Parallelized with domain decomposition
- Forecast duration hours to days
- Current resolution 2-10KM
- Usually more detailed physics
  - Runtime can be up to 6 hours
- Applications
  - Local severe weather
  - Fire weather support
  - Hydrology

Weather Forecasting Common Features



- Data ingest
  - Small observations arriving in < Kbyte packages
  - Multiple types of data
  - Large input data such as radar and satellite
- Postprocessing
  - Produce graphics
  - Produce WMO-standard data formats
- Pre- and postprocessing results in hundreds of jobs per forecast run

Weather Forecasting
Planned Science Improvements

- Cloud resolving physics
- More detailed solar radiation
- More sophisticated precipitation characterizations
- Four-dimensional variational assimilation (4DVAR)
  - Initial conditions
  - Time varying observations
  - Run adjoint of model backwards and model forwards to minimize error in observations
  - Two orders of magnatude greater than model itself

Example Computational Requirements

- 20KM CONUS requires 36 Xeon CPUs to complete in less than an hour
  - => 10KM will require > 360 Xeon CPUs
  - => 5KM will require > 3600 Xeon CPUs
- 22KM CONUS requires 88 Power3 CPUs to complete in less than an hour (more detailed physics/radiation)

Software Development

- Typically done with MPI
  - Some models have OpenMP as an alternative parallelization
  - Most communications is nearest neighbor in nature for FDA, transpose for Spectral
- Codes are typically regular
- I/O
  - Checkpointing is often on a per node basis
  - Data saves
    - Collapse to single node
    - Write multiple files

FSL Developed Software

- Scalable Modeling System (SMS)
  - Ease of use
    - Minimize code changes
    - Support debugging
  - High performance
    - Don't break serial optimizations
  - Correctness
    - Same result as serial code for any number of processes
  - Portability

NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
NOAA
U.S. DEPARTMENT OF COMMERCE

Sample SMS Code

```
        integer IM, i
        parameter(IM = 15)
CSMS$DECLARE_DECOMP(my_dh, <(IM/3) + 4>)

CSMS$DISTRIBUTE(my_dh, <IM>) BEGIN
        real x(IM), y(IM), xsum
CSMS$DISTRIBUTE END

C Begin executable code
CSMS$CREATE_DECOMP (my_dh, <IM>, <2>)

        open (10, file = 'x_in.dat', form='unformatted')
        read (10) x
        close (10)

CSMS$PARALLEL(my_dh, <i>) BEGIN
        do 100 i = 3, 13
          y(i) = x(i)  - x(i-1) - x(i+1) - x(i-2) - x(i+2)
  100   continue
CSMS$EXCHANGE(y)
        do 200 i = 3, 13
           x(i) = y(i) + y(i -1) + y(i+1) + y(i -2) + y(i+2)
  200   continue
        xsum = 0.0
        do 300 i = 1, 15
           xsum = xsum + x(i)
  300   continue
CSMS$REDUCE(xsum, SUM)
CSMS$PARALLEL END
        print *,'xsum = ',xsum
```

SMS Code Example

```
        integer IM, i
        parameter(IM = 15)
CSMS$DECLARE_DECOMP(my_dh, <(IM/3) + 4>)

CSMS$DISTRIBUTE(my_dh, <IM>) BEGIN
        real x(IM), y(IM), xsum
CSMS$DISTRIBUTE END

C Begin executable code
CSMS$CREATE_DECOMP (my_dh, <IM>, <2>)
        open (10, file = 'x_in.dat', form='unformatted')
        read (10) x
        close (10)
CSMS$PARALLEL(my_dh, <i>) BEGIN
        do 100 i = 3, 13
          y(i) = x(i) - x(i-1) - x(i+1) - x(i-2) - x(i+2)
  100   continue
CSMS$EXCHANGE(y)
CSMS$CHECK_HALO(y, 'before loop 200')
        do 200 i = 3, 13
           x(i) = y(i) + y(i -1) + y(i+1) + y(i -2) + y(i+2)
  200   continue
CSMS$COMPARE_VAR(x, 'after loop 200')
        xsum = 0.0
        do 300 i = 1, 15
           xsum = xsum + x(i)
  300   continue
CSMS$REDUCE(xsum, SUM)
CSMS$PARALLEL END
        print *,'xsum = ',xsum
```

Summary of Needs

- Computational requirements will increase dramatically over the next few years
- For operational requirements clusters need to be highly stable in all ways
- Weather forecasting needs deterministic runtimes and deterministic scheduling