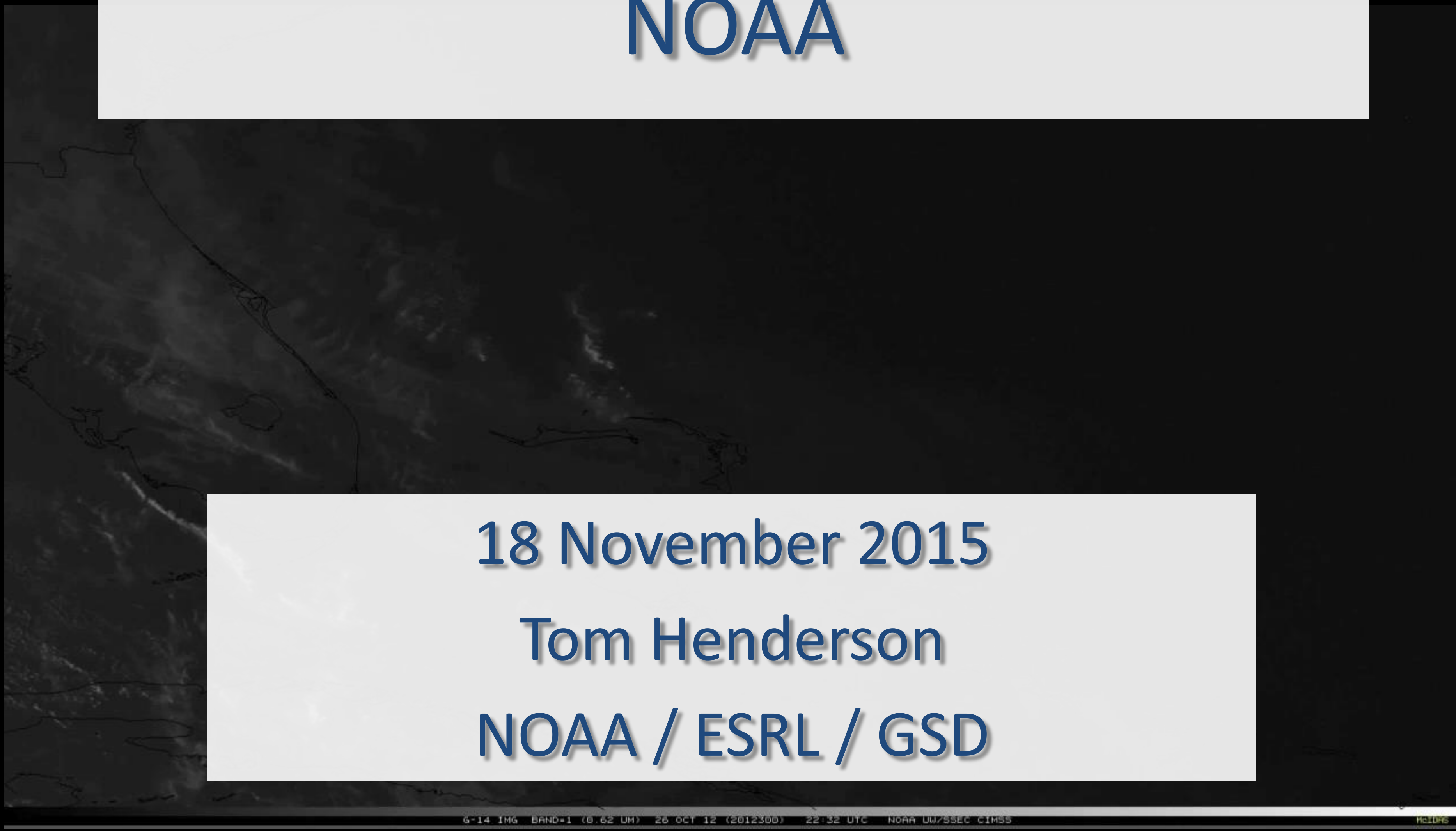
Advancing Weather Prediction at NOAA

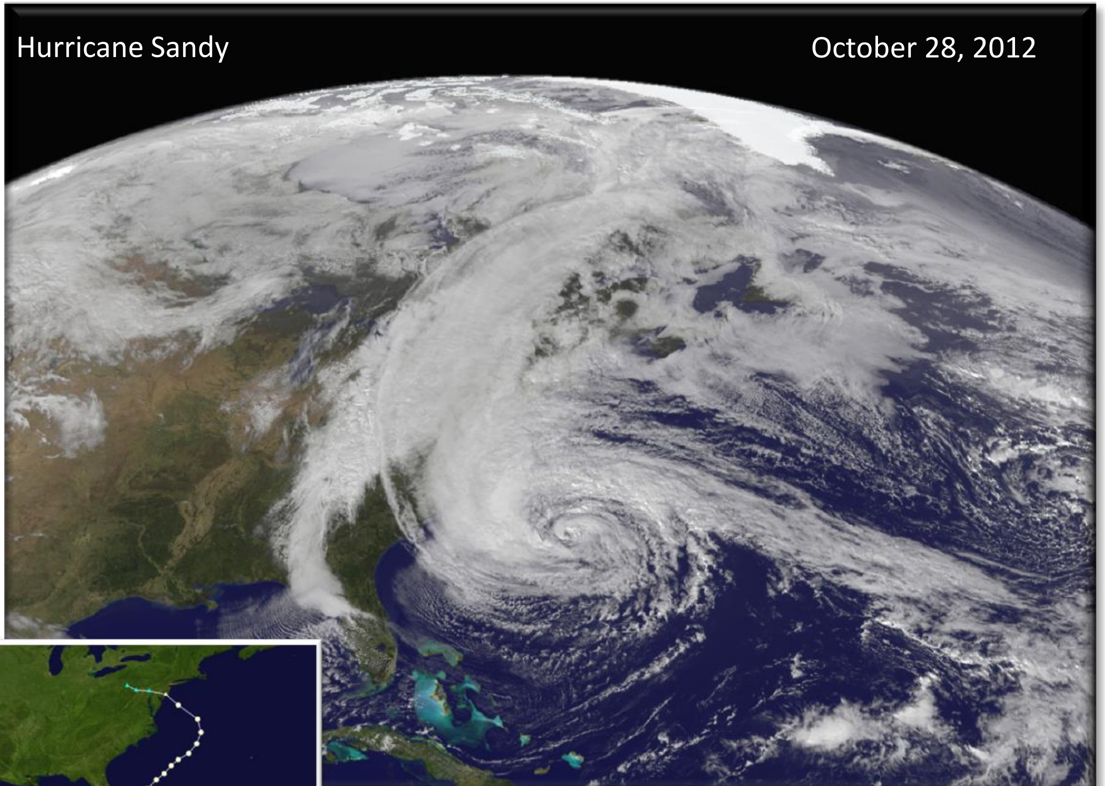
18 November 2015

Tom Henderson

NOAA / ESRL / GSD



The U. S. Needs Better Global Numerical Weather Prediction



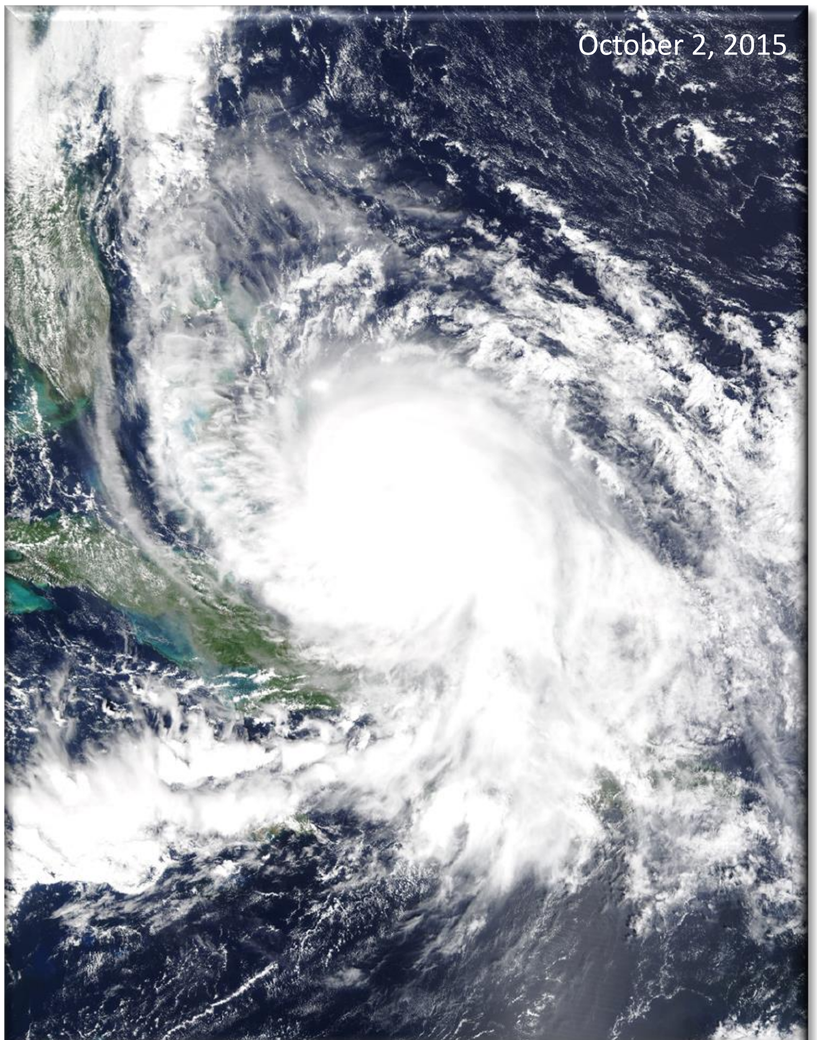
“A European forecast that closely predicted Hurricane Sandy's onslaught days ahead of U.S. and other models is raising complaints in the meteorological community.”

"The U.S. does not lead the world; we are not No. 1 in weather forecasting, I'm very sorry to say that," says AccuWeather's Mike Smith…”

Source: USA Today, October 30, 2012

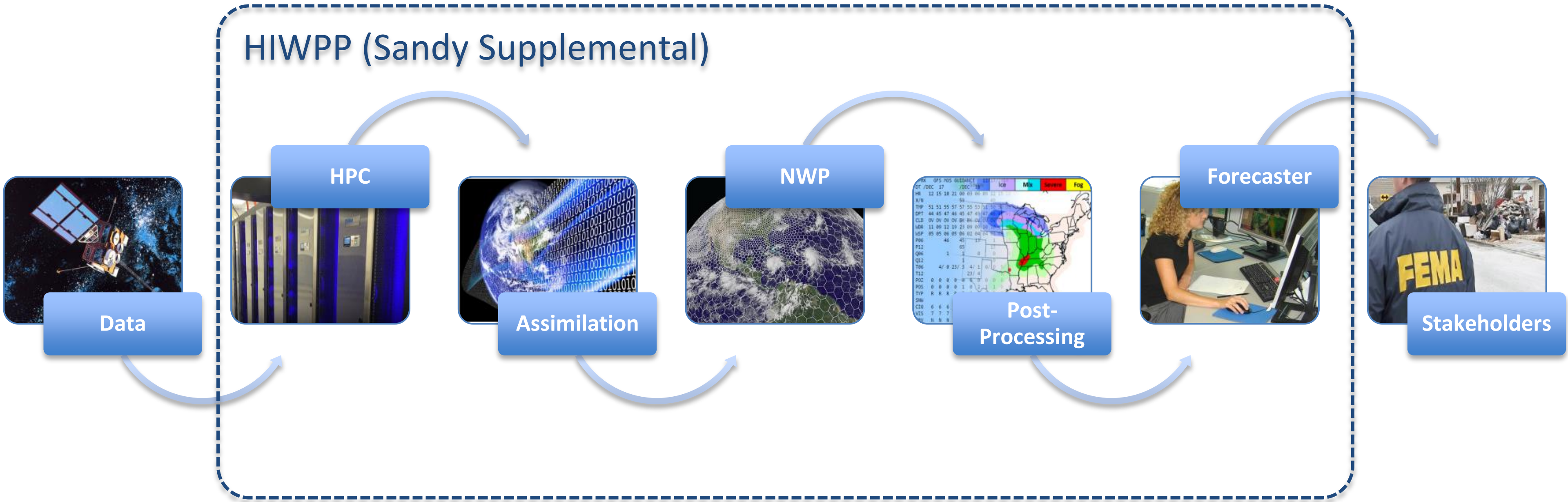
Congressional Response:
High Impact Weather Prediction Program (HIWPP)

Three Years Later… Hurricane Joaquin



- US forecast track improved, but still not quite as good as European’s
- NOAA’s Hurricane Weather Research & Forecast Model intensity forecasts were ‘second to none’
- US research models had 20” precipitation forecasts in South Carolina 36 hours in advance (verified)
- The National Hurricane Center *correctly* never issued any hurricane watches or warnings for the mainland

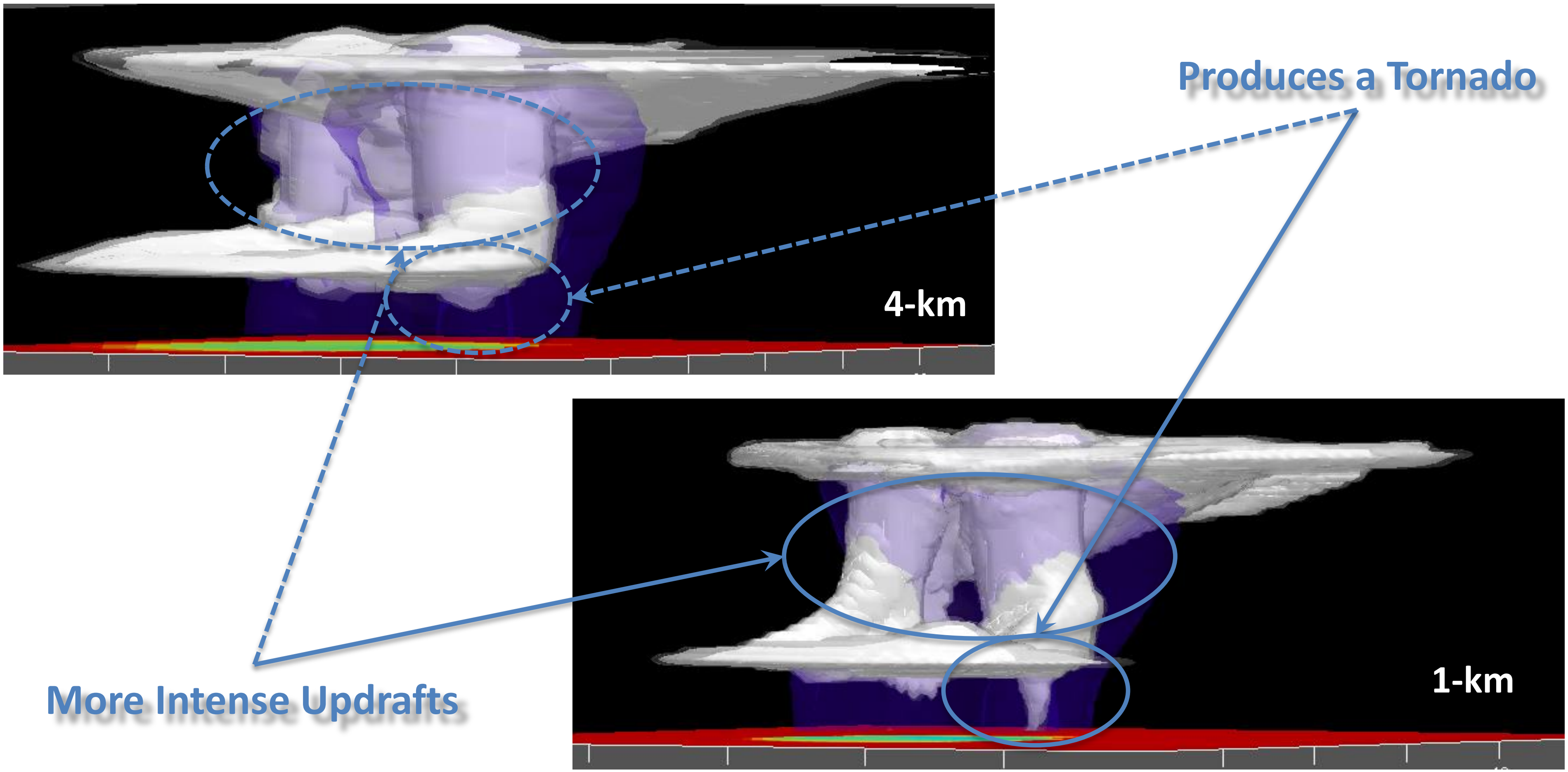
The Forecast Process



HPC critical for Assimilation and Numerical Weather Prediction (NWP)

NWP Resolution Matters:

Simulation of a Tornado-Producing Super-Cell Storm

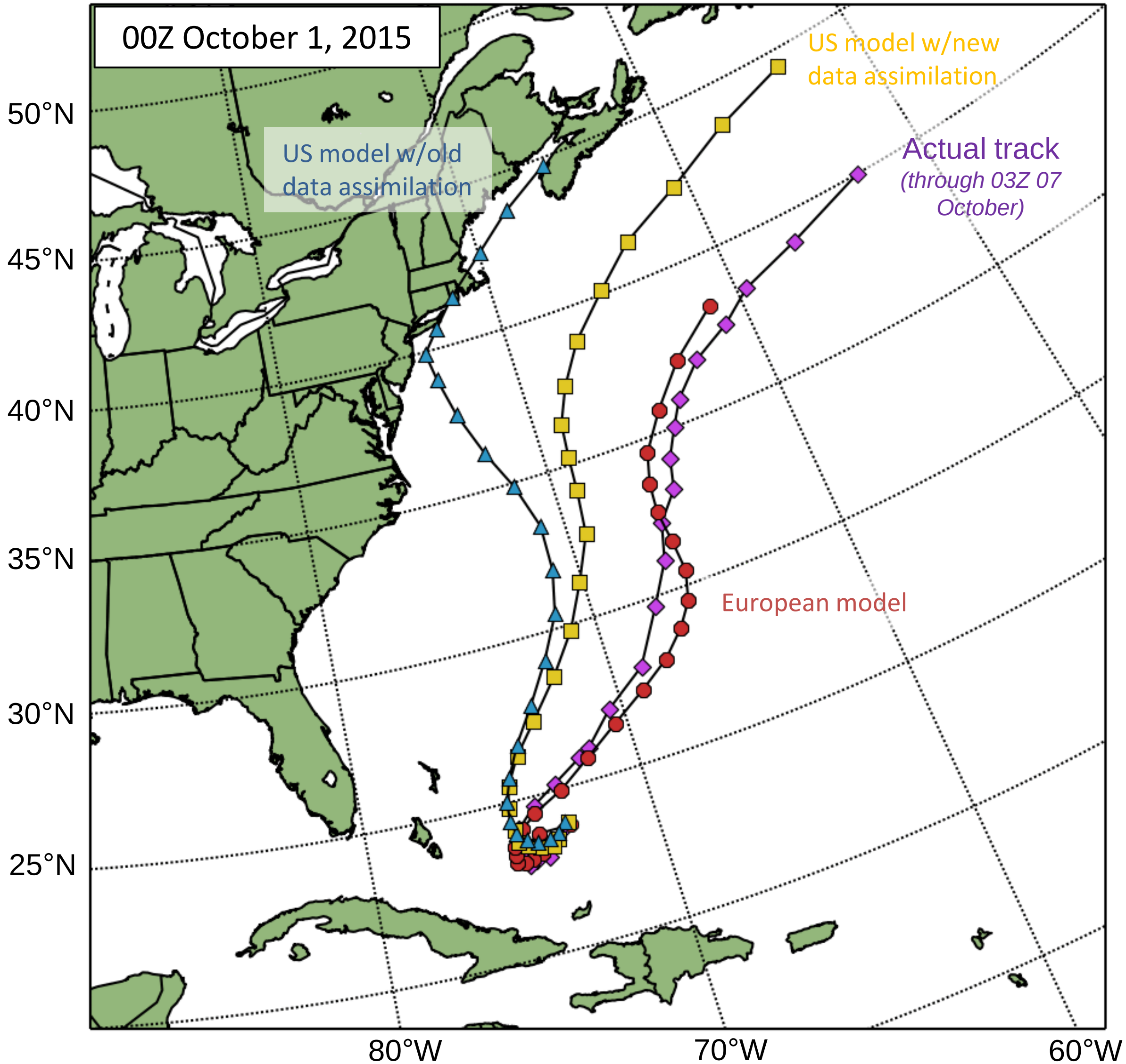


Resolution of current operational GFS: ~13km

Simulations with GFDL's variable-resolution FV^3, non-hydrostatic (aka cloud-permitting) model. Courtesy of Lin and Harris (2015 manuscript)

Better Data Assimilation = Better Forecasts

Hurricane Joaquin



Source: Corey Guastini
EMC's Model Evaluation Group

Enabler: High Performance Computing

New supercomputing funded by the Sandy Supplemental Disaster Relief Act:

- Enabling Sandy Supplemental R&D
- 30% Increased Capacity

Research

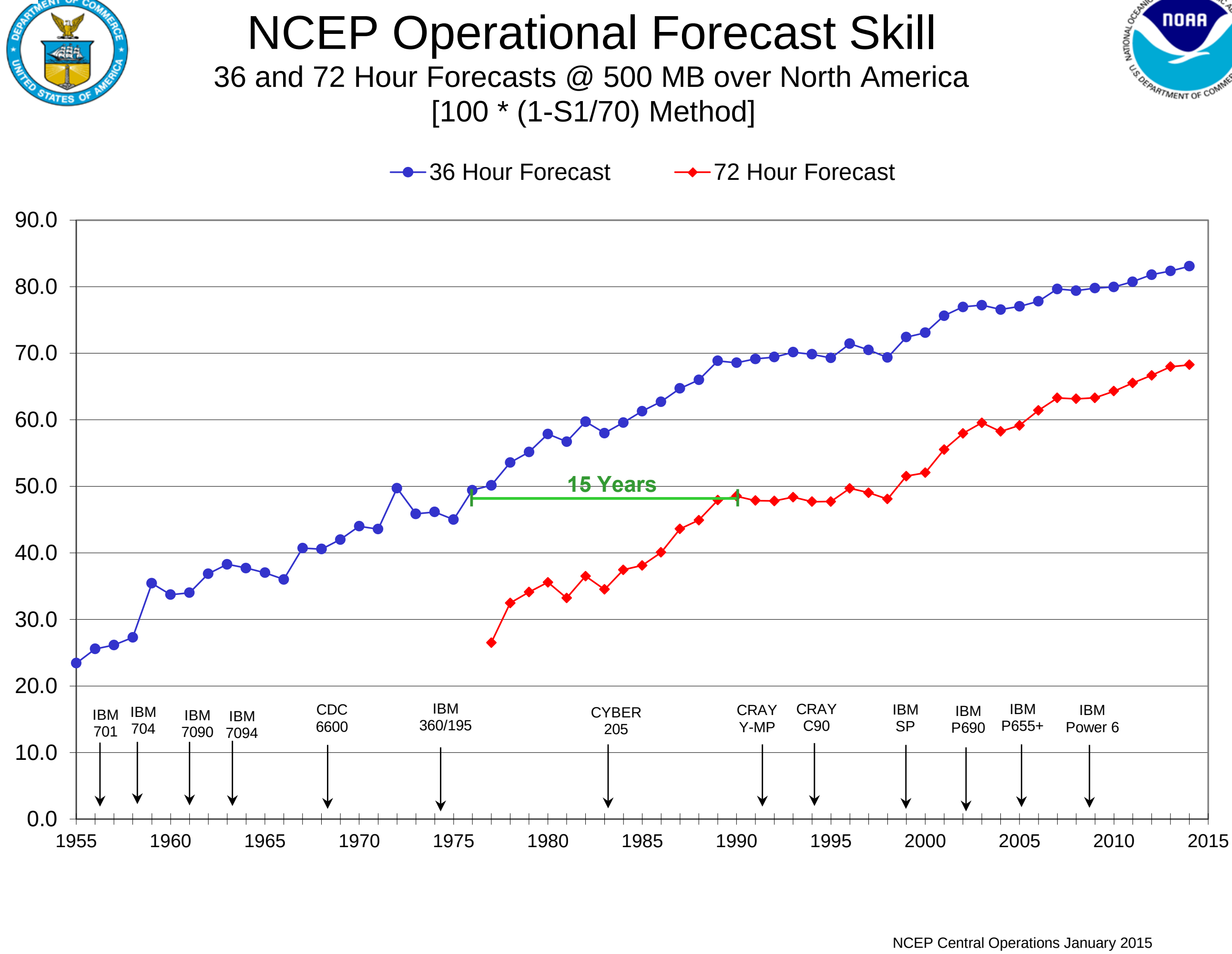
- Weather & Climate Supercomputing System
- 2 petaflop increase

Operations

- Accelerator technology
- Exploratory project

Fine Grain

Forecast Skill Improvement



Can Rate of Improvement be Maintained?

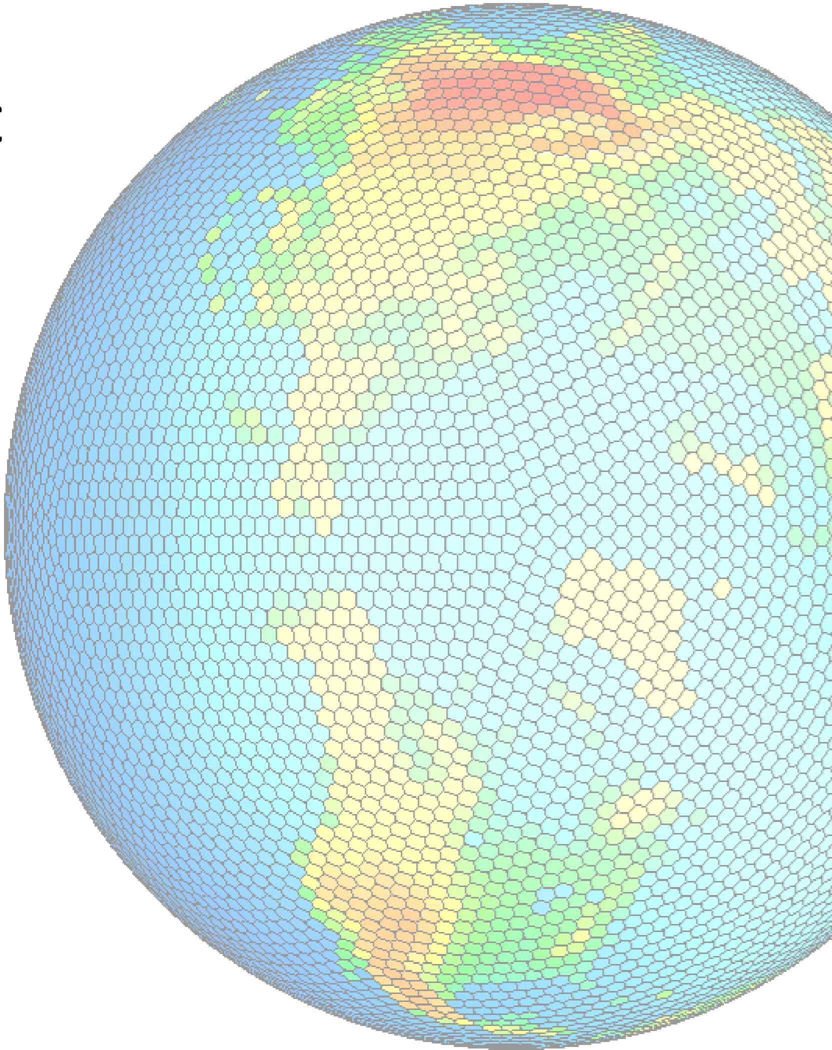
- Exponentially increasing computer power has enabled linear forecast improvement
- How to overcome CPU clock speed stall?
  - More cores -> must exploit fine-grained parallelism
  - MIC, GPU?
- Two components of modern NWP models
  - **“Dynamics”**: large scale flows
  - **“Physics”**: sub grid-scale parameterizations
- Key requirement: single source code

NWP Dynamics on GPU and MIC

- Non-Hydrostatic Icosahedral Model (NIM)
  - Single-source => directives
    - GPU OpenACC, F2C-ACC
    - CPU OpenMP
    - MIC OpenMP
    - MPI SMS

<u>Fine-Grained Parallelism</u>

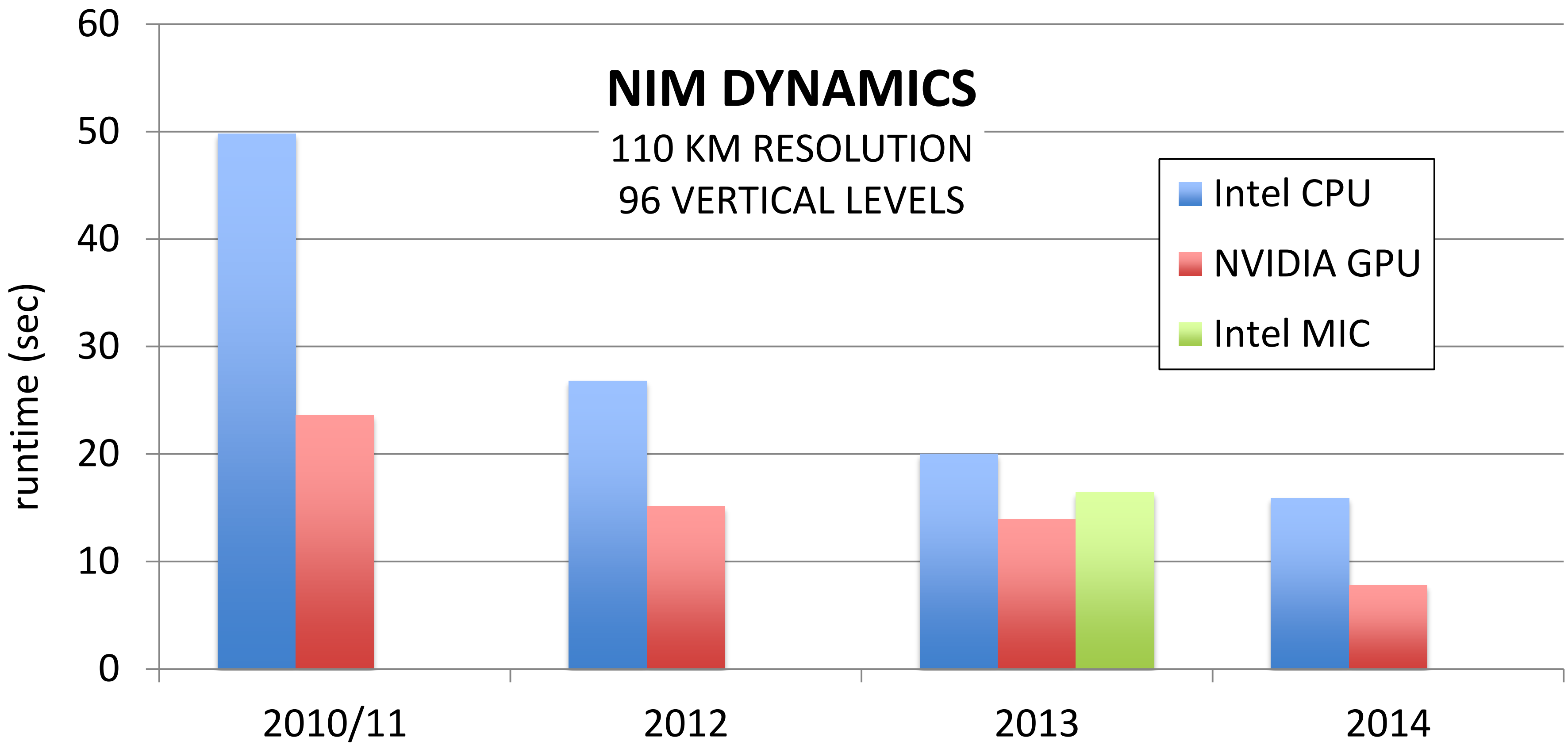
- **GPU**
  - “Blocks” in horizontal
  - “Threads” in vertical
- **CPU, MIC**
  - “Threads” in horizontal
  - “Vectors” in vertical



NIM Single Source Code

```
!SMS$PARALLEL (ipn) BEGIN
!$OMP PARALLEL DO
do i=1,nip
!$ACC loop gang(16), vector(32)
  do k=1,nz
    x(k,i) = z(k,i) + y(k,i)
  end do
!$ACC end loop
end do
!$OMP END PARALLEL DO
!SMS$ PARALLEL END
```

NIM Dynamics Performance



<u>Performance of the NIM dynamics:</u>

- CPU: Westmere (2010), SandyBridge (2012), IvyBridge (2013), Haswell (2014)
- GPU: Fermi (2011), Kepler K20x (2012), Kepler K40 (2013), Kepler K80 (2014)
- MIC: Knights Corner (2013)
- NOTE: CPU times are for two sockets, GPU and MIC for one device

Use F2C-ACC To Improve OpenACC

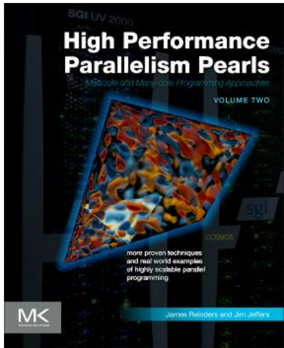
- OpenACC performance is now competitive with "domain-specific" F2C-ACC compiler
  - In 2014 PGI and Cray OpenACC compilers were ~2x slower than F2C-ACC for the NIM routines shown below

NIM Routine	**F2C-ACC v5.8 (sec)**	**PGI v15.7 (sec)**	**PGI Speedup**
vdmints	4.58	5.40	0.84
vdmintv	1.63	2.29	0.71
diag	1.24	0.65	1.90
flux	0.98	1.03	0.95
force	0.61	0.52	1.17
trisol	0.36	0.28	1.28
TimeDiff	0.18	0.15	1.20
Total	**11.77**	**12.34**	**0.95**

Performance of WSM6 "Physics" Package

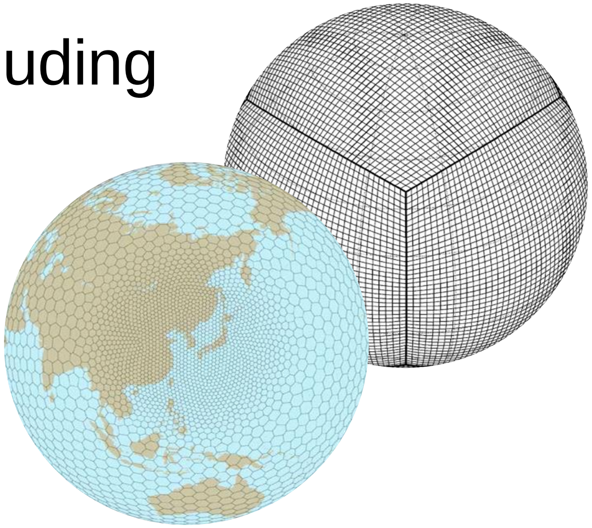
Device	Threads	Chunk Width (DP words)	Baseline Time	Optimized Time	Speedup
KNC	240	8	13.2	8.7	1.52
SNB	32	4	9.4	7.5	1.25
IVB-EP	48	4	4.7	3.4	1.38
HSW-EP	56	4	--	2.6	--
K20X GPU	--	--	--	5.3	--

- Most performance improvement due to compile-time constants and alignment
- GPU result "preliminary", expect ~2x further speedup on K80x
  - … and even more speedup on KNL



Next Generation Global Prediction System (NGGPS)

- Implement a cloud-permitting coupled NWP system
- Through accelerated development and implementation of
  - Current global weather prediction models and physics
  - Improved data assimilation techniques, and
  - Improved software architecture and system engineering
- Extend forecast skill beyond two weeks
- Improve high-impact weather forecasts including hurricane track and intensity
- Build upon HIWPP successes
- Apply NIM optimization techniques
- Continue strong vendor interactions



Summary

- Must make U.S. global NWP “second to none”
- More HPC needed for NWP and assimilation
  - BUT, clock frequencies have stalled
- “Accelerators” to the rescue?
  - Dynamics: some benefit
  - Physics: TBD
- Expect “high-bandwidth memory” to give an edge to future MIC & GPU
  - KNL, “Pascal”
- Optimizations benefit MIC, GPU, ***and CPU***
- Procurement of exploratory “fine-grained” system

Thanks to…

- GSD: Tim Schneider, Mark Govett, Jim Rosinski, Jacques Middlecoff, Paul Madden, Julie Schramm
- NWS: John Michalakes
- Strong vendor interactions
  - Cray
  - Intel
  - NVIDIA
  - PGI
- HIWPP, NGGPS, NOAA HPCC