Volcanic ash transport using parallel, adaptive Ash3d

Donna Calhoun (Boise State University)

Melody Shih (NYU),
Roger Denlinger (USGS),
Larry Mastin (USGS)
Hans Schwaiger (USGS)

PNWNAS-SIAM

October 27-29, 2017

Corvallis, OR

2010 Eyjafjallajökull

The April 15-21, 2010 Ejyafjallajokull eruption in Iceland caused unprecedented disruption to civil European airspace.

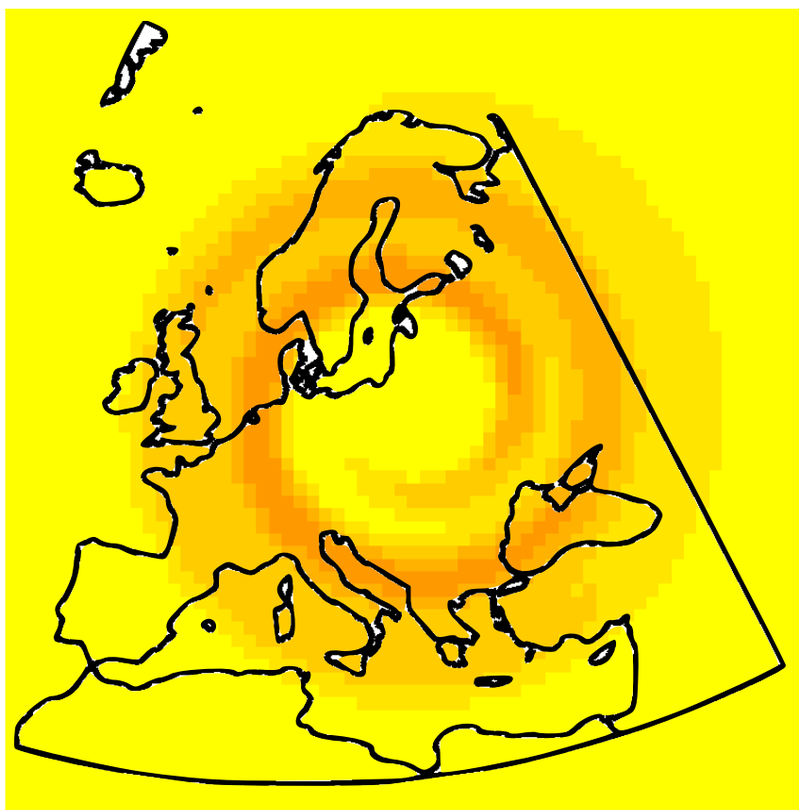
- 25 countries and over 4 million passengers were affected
- Cost to the airlines were measured in the billions of Euros (Oxford-Economics, 2010)

On April 20th, the previous "zero-ash-tolerance" policy were substituted for policies allowing for low level concentrations of ash in commercial airspace.

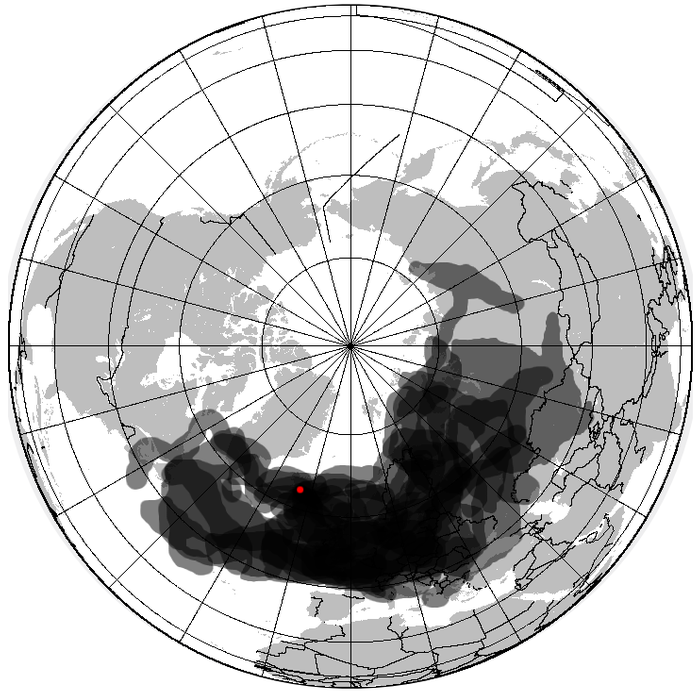
As a result numerical models must now be able to report ash concentration levels accurately.

Volcanic eruption 2010

Horizontal scale ~ O(5000km)

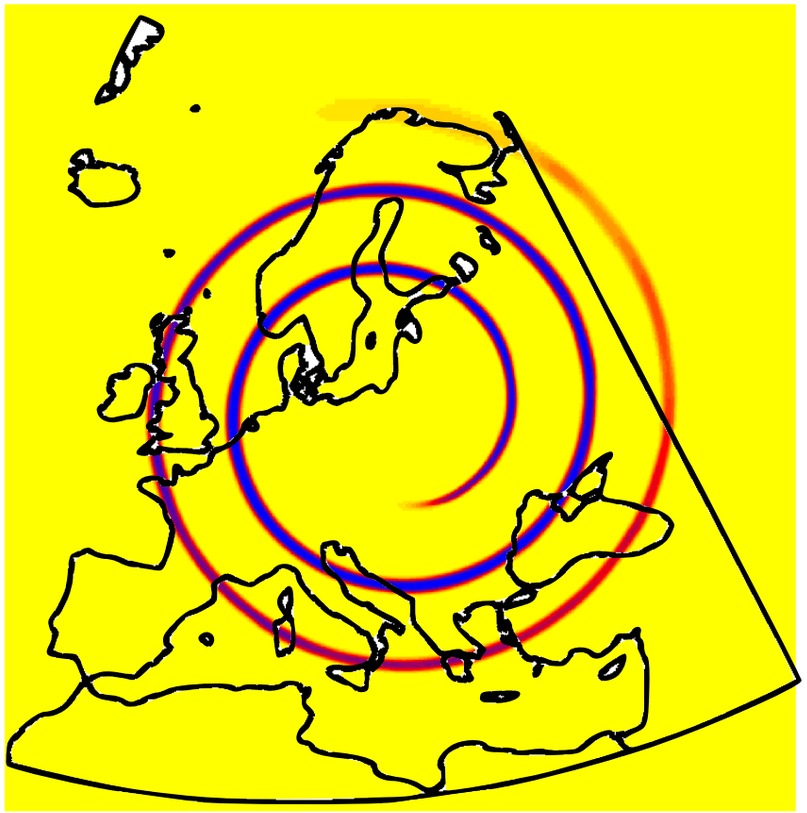


64 x 64 (~80 km resolution)



2010 Eruption in Iceland

Evolution has to be tracked over several days



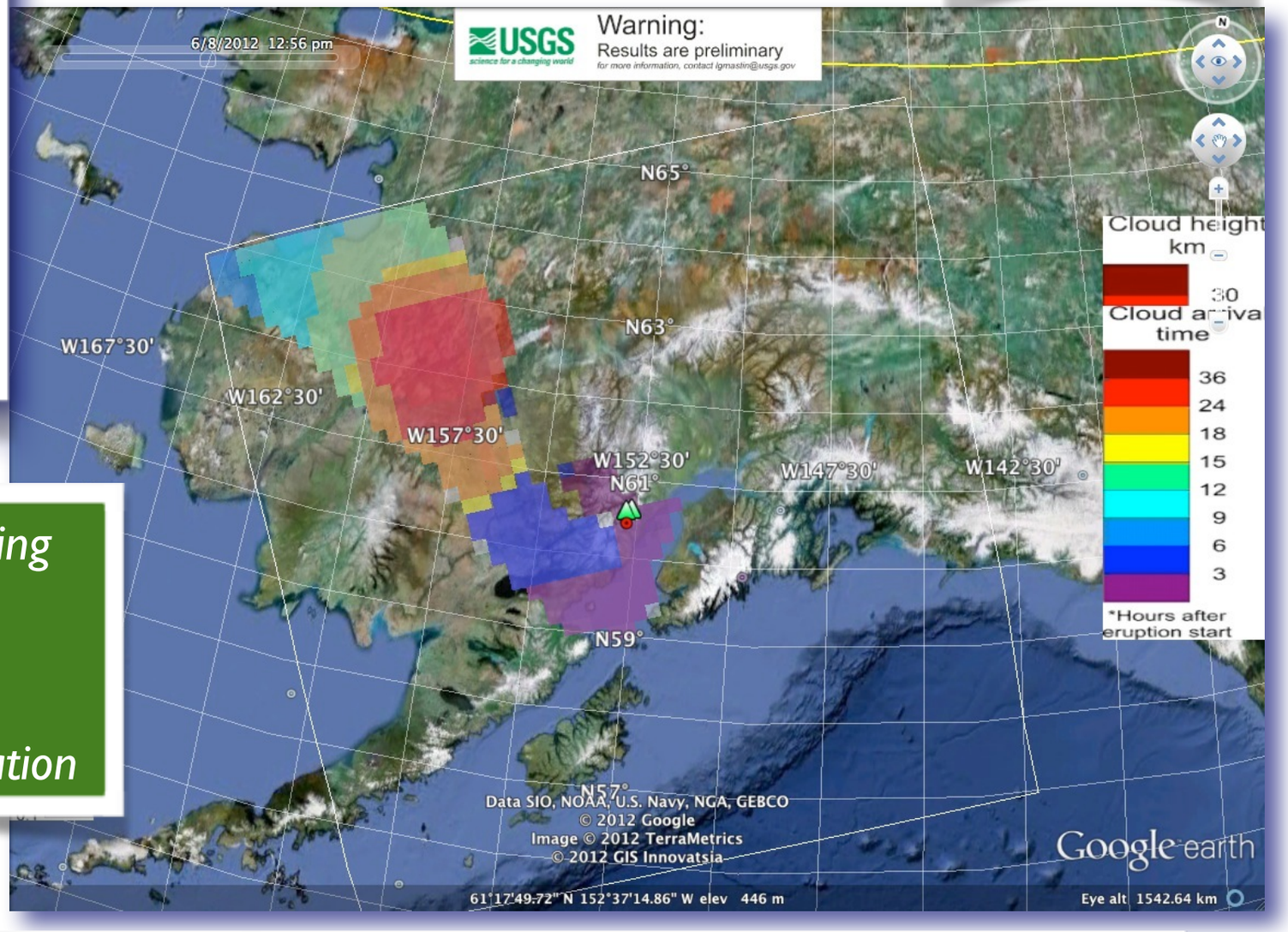
1024 x 1024 (~5 km resolution)

Need for high resolution

- Importance of tracking small scale structures in smooth, even coarse-resolution wind fields (Behrens, MWR 2000)
- Observational evidence supports the idea that there is considerable variation in concentration levels in tracers in the atmosphere, not just large diffuse "clouds" of tracers.
- Eruptions lasting several days or weeks can place heavy demands on computational resources.

Ash 3d : Ash cloud modeling

Ash3d



- *Split horizontal, vertical time stepping*
- *Fully conservative,*
- *Eulerian, finite volume*
- *Algorithms based on wave propagation*

Ash3d :A finite-volume, conservative numerical model for ash transport and tephra deposition, Schwaiger, Denlinger, Mastin, JGR (2012)

Ash3d

The scalar advection equation in conservative form

$$q_t + \nabla(\mathbf{u}q) = \Psi(q)$$

where the velocity field $\mathbf{u}$ is prescribed from wind field data available from various weather services (e.g. NOAA) and q is a concentration of ash. One or more grain sizes may be tracked.

- Use high-resolution wave propagation algorithms
- Variable-coefficient advection problem raises interesting issues (potential loss of hyperbolicity near sonic points, for example).
- Source term models eruption using one of several well-established models

Ash3d

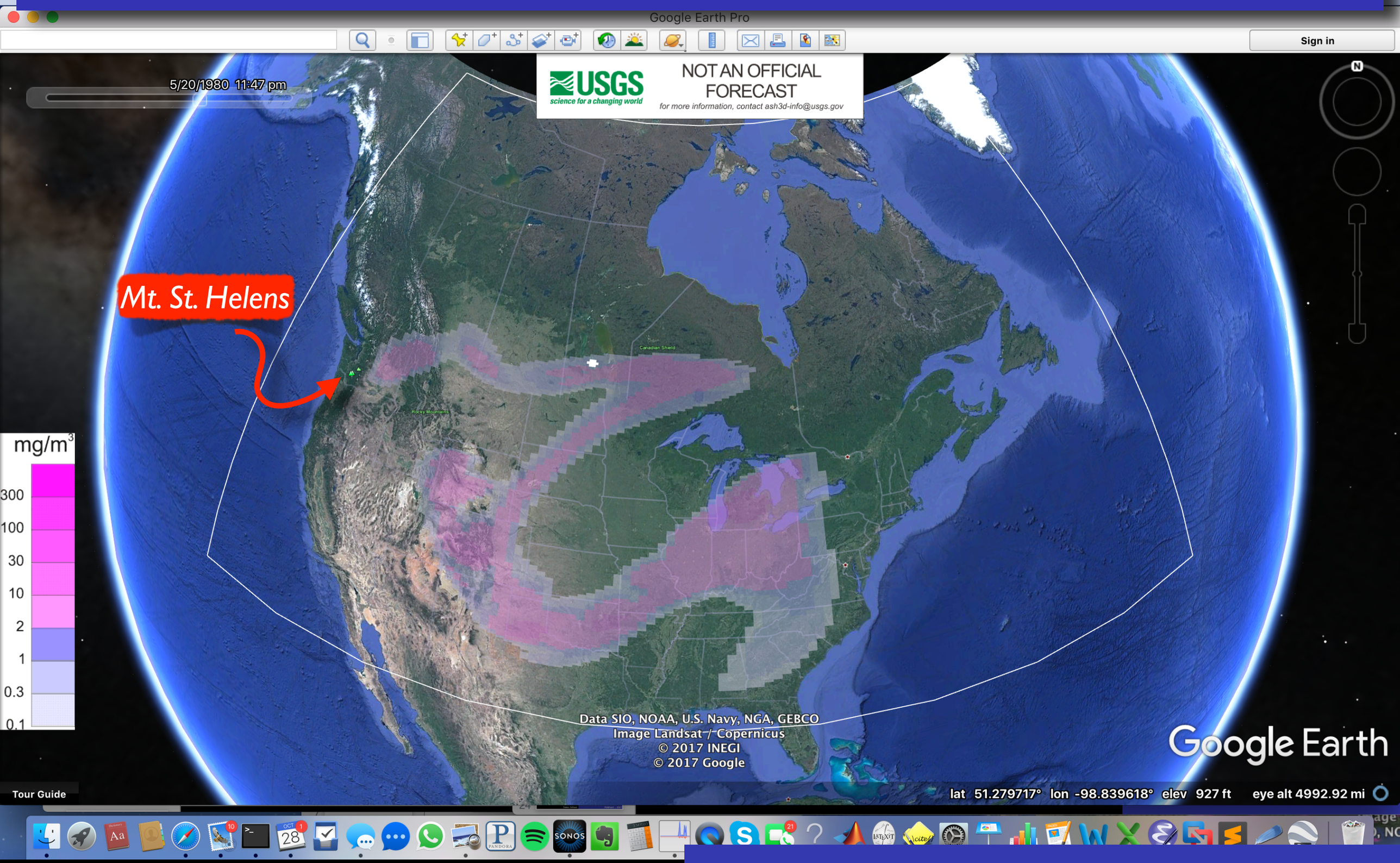
Computational domain

- Regional finite volume, lat-long grid
- 3d computations, with roughly 25 cells in the vertical.
- Runs on a single processor (serial code)
- Wind fields interpolated in space and time from forecast (or re-analysis) data.

Numerical formulation

- Explicit finite volume scheme with wave limiters (ala Clawpack)
- Eulerian formulation ensures conservation of ash mass,
- Conservative discretization designed to preserve second order accuracy

Ash3d Results



ForestClaw Project

A parallel, adaptive library for logically Cartesian, mapped, multi-block domains

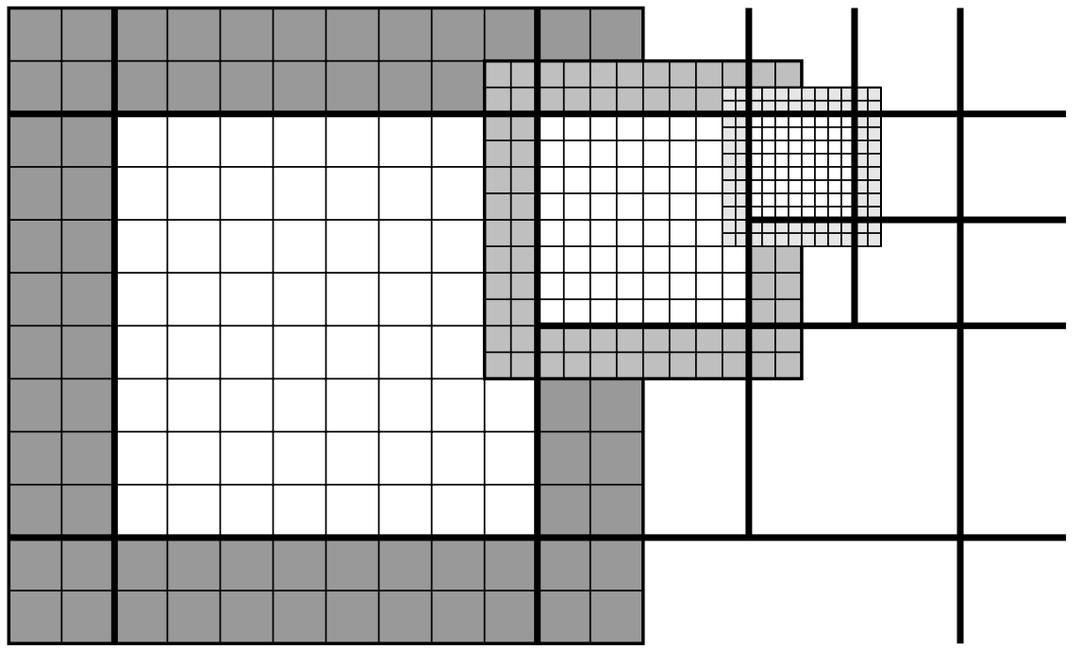
Features of ForestClaw include :

- Uses the **highly scalable p4est** dynamic grid management library (C. Burstedde, Univ. of Bonn, Germany)
- Each leaf of the quadtree contains a fixed, uniform grid,
- Optional multi-rate time stepping strategy,
- Has **mapped, multi-block** capabilities, (cubed-sphere, for example) to allow for flexibility in physical domains,
- Modular design gives user flexibility in extending ForestClaw with Cartesian grid based solvers and packages.
- Uses essentially the same algorithmic components as patch-based AMR

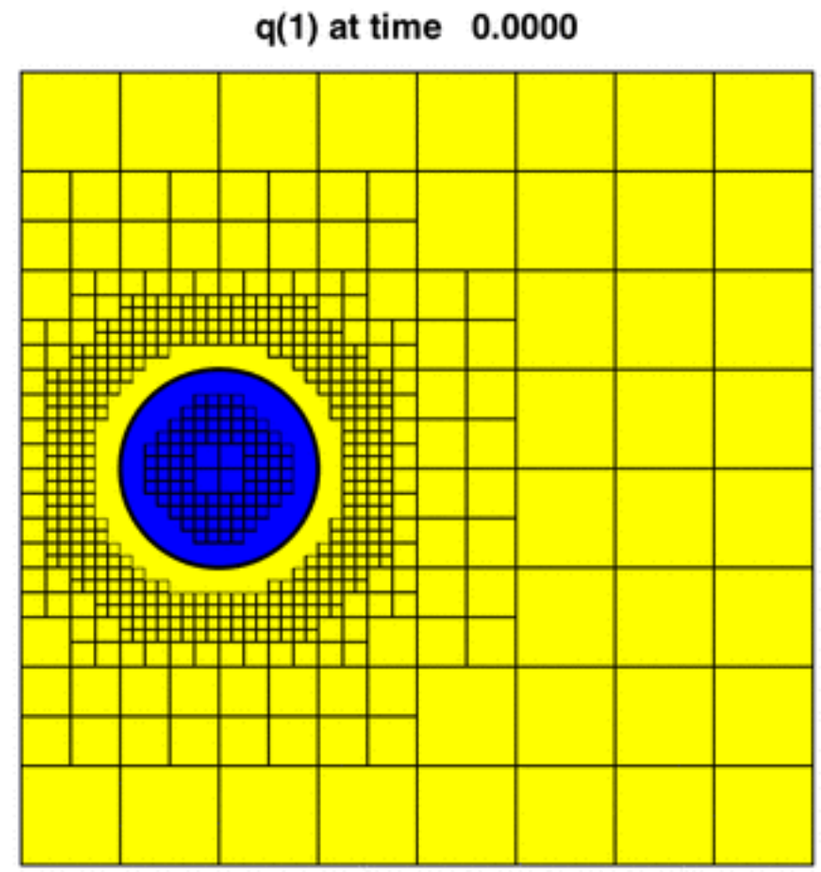
Thanks to NSF for supporting this work

www.forestclaw.org

ForestClaw adaptivity



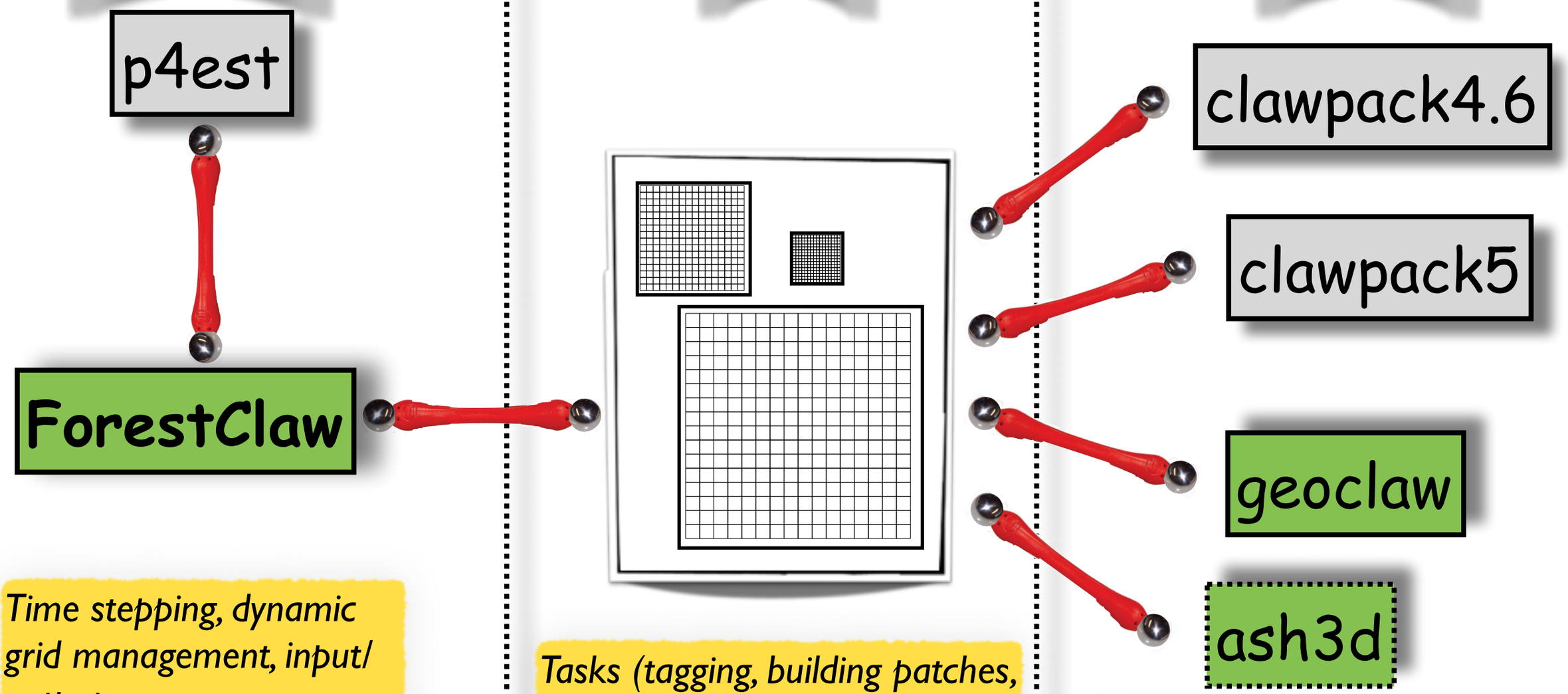
Each quadrant is a single logically grid, designed for finite volume or finite difference solvers.



(Behrens, MWR 2000)

Regridding, connectivity done using p4est (www.p4est.org)

Extending ForestClaw

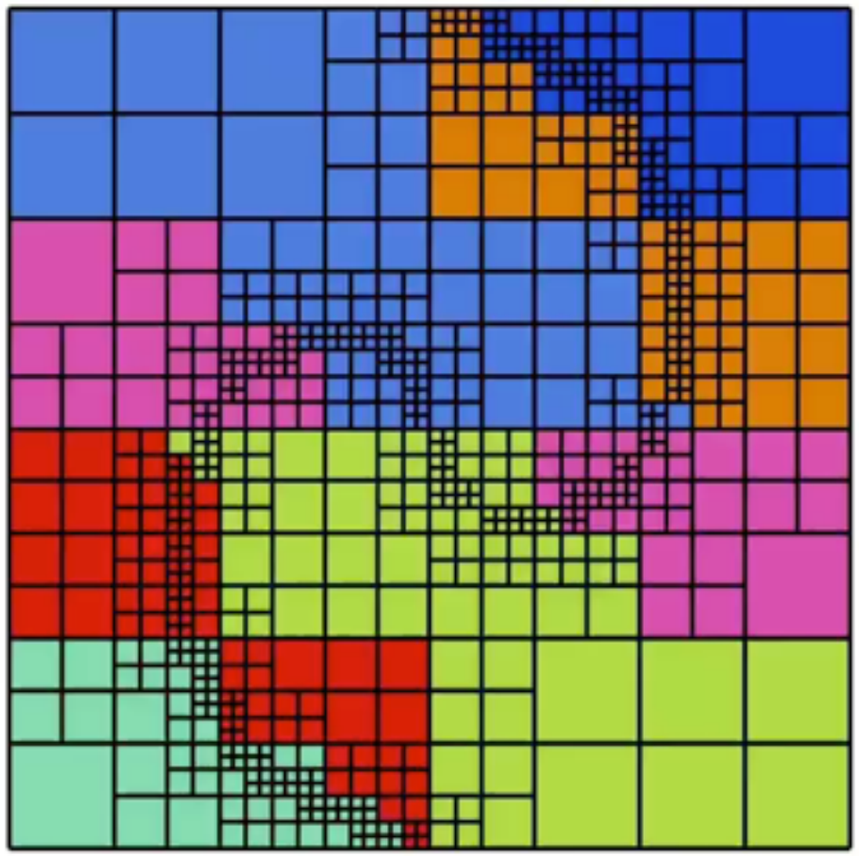


Time stepping, dynamic grid management, input/output.

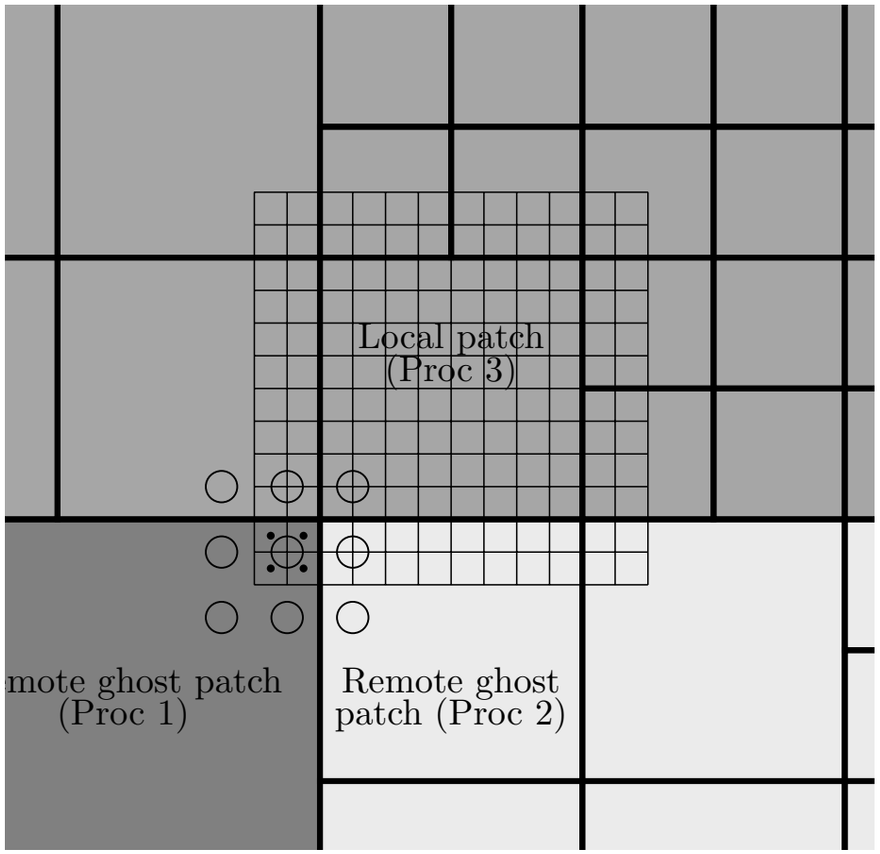
Tasks (tagging, building patches, etc) customized through use of function pointers stored in "virtual tables"

ForestClaw is mostly C; solvers extension libraries left largely untouched (in original Fortran)

ForestClaw - Parallelism



p4est : Load balancing using a space filling curve

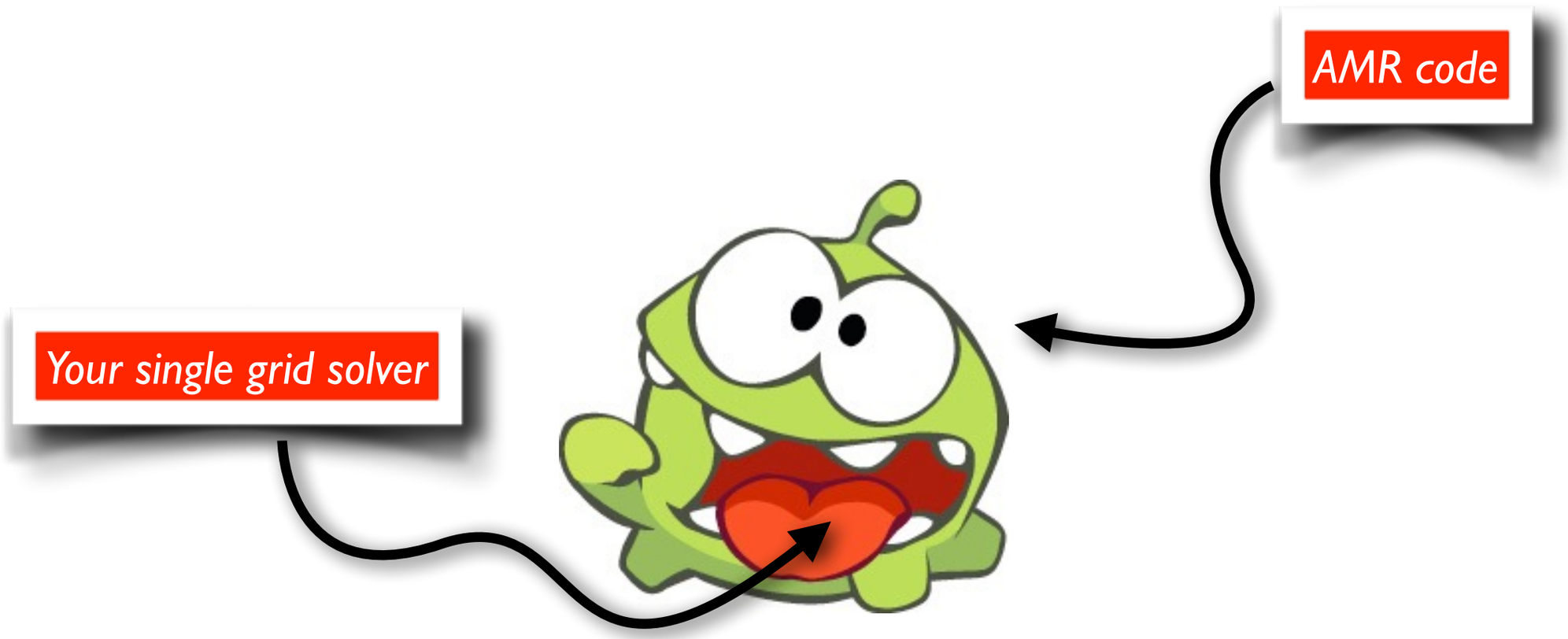


Fine grid corner ghost cells at corners where 3 or more processors meet

D. Calhoun and C. Burstedde, *"ForestClaw : A parallel algorithm for patch-based adaptive mesh refinement on a forest of quadtrees", (submitted), 2017. (arXiv:1703.03116)*

AMR ?

Mental model of how this might work :



** Idea for code name : OmNum*

Block-structured AMR

The Dream

```
AMR.run(max_time, max_steps);
```

Your single grid solver is called from here.

Block-structured AMR

The Reality

```
Tuple< RefCountedPtr<AMRLevelOpFactory<
LevelData<FArrayBox> > >, SpaceDim> velTGAOpFactoryPtrs;

for (int idir = 0; idir < SpaceDim; idir++)
{velTGAOpFactoryPtrs[idir] =
    RefCountedPtr<AMRLevelOpFactory<LevelData
    <FArrayBox> > >
    ((AMRLevelOpFactory<LevelData<FArrayBox> >*)
    (new AMRPoissonOpFactory())); //.....
```

Your single grid solver

you

Retro...

```
node(ndjhi,mptrnx) = node(ndjhi,mptr)
node(ndjhi,mptr)   = node(ndjlo,mptr) + nyl - 1
node(ndjlo,mptrnx) = node(ndjhi,mptr) + 1
node(ndihi,mptrnx) = node(ndihi,mptr)
node(ndilo,mptrnx) = node(ndilo,mptr)

rnode(cornxlo,mptrnx)   = cxlo
rnode(cornylo,mptrnx)   = cymid
rnode(cornyhi,mptrnx)   = cyhi
rnode(cornxhi,mptrnx)   = cxhi
node(nestlevel,mptrnx)  = node(nestlevel,mptr)
rnode(timemult,mptrnx)  = rnode(timemult,mptr)
go to 10
```

Ash3d - Fortran modules

```
MODULE ash3d
    REAL(kind=8),ALLOCATABLE :: concen(:,:,:,:)
    !! ...
END
```

```
SUBROUTINE allocate()
    USE ash3d
    IMPLICIT NONE

    ALLOCATE(concen(0:n+1,0:n+1,0:n+1,2))
    !! ...
END
```

Original code: Single grids allocated at start of run

```
SUBROUTINE update_concen()
USE ash3d
IMPLICIT NONE

concen(:,:,:,1) = concen(:,:,:,0) + dt*(your favorite scheme)
!! ...
END
```

Ash3d - Fortran modules

```
struct patch
{
    double *concen;
    /*  ... + 30 more arrays */
};
```

```
void init_patch(patch* p)
{
    allocate_();
    store_ptrs_(&p->concen);
}
```

```
void update(patch* p)
{
    copy_ptrs2mod_(p->concen);
    update_concen_();
}
```

Original code

```
MODULE ash3d
REAL(kind=8), POINTER :: concen(:,:,:,:)
INTEGER(KIND=4) :: n
!! ... + 30 more arrays !!
END
```

```
SUBROUTINE allocate()
    USE ash3d
    IMPLICIT NONE

    ALLOCATE(concen(0:n+1,0:n+1,0:n+1,2))
END
```

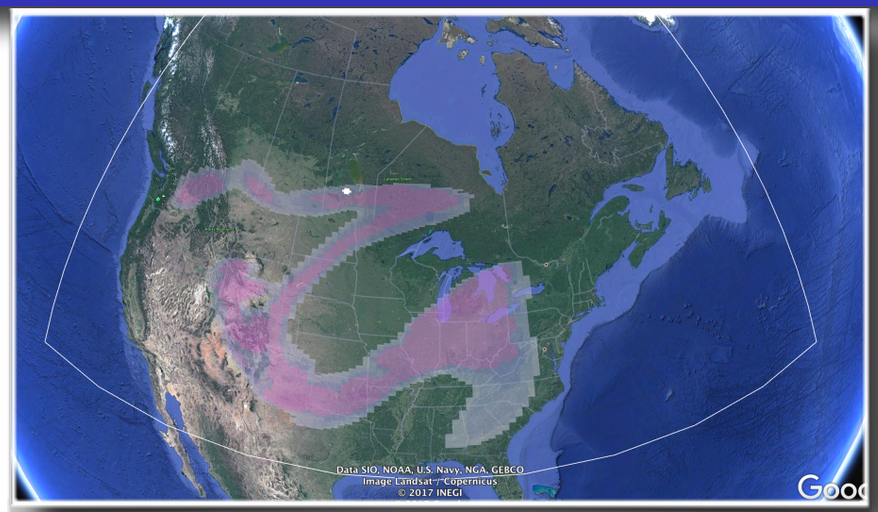
```
SUBROUTINE update_concen()
USE ash3d
IMPLICIT NONE

concen(:,:,:,1) = concen(:,:,:,0) &
                + dt*(your favorite scheme)
END
```

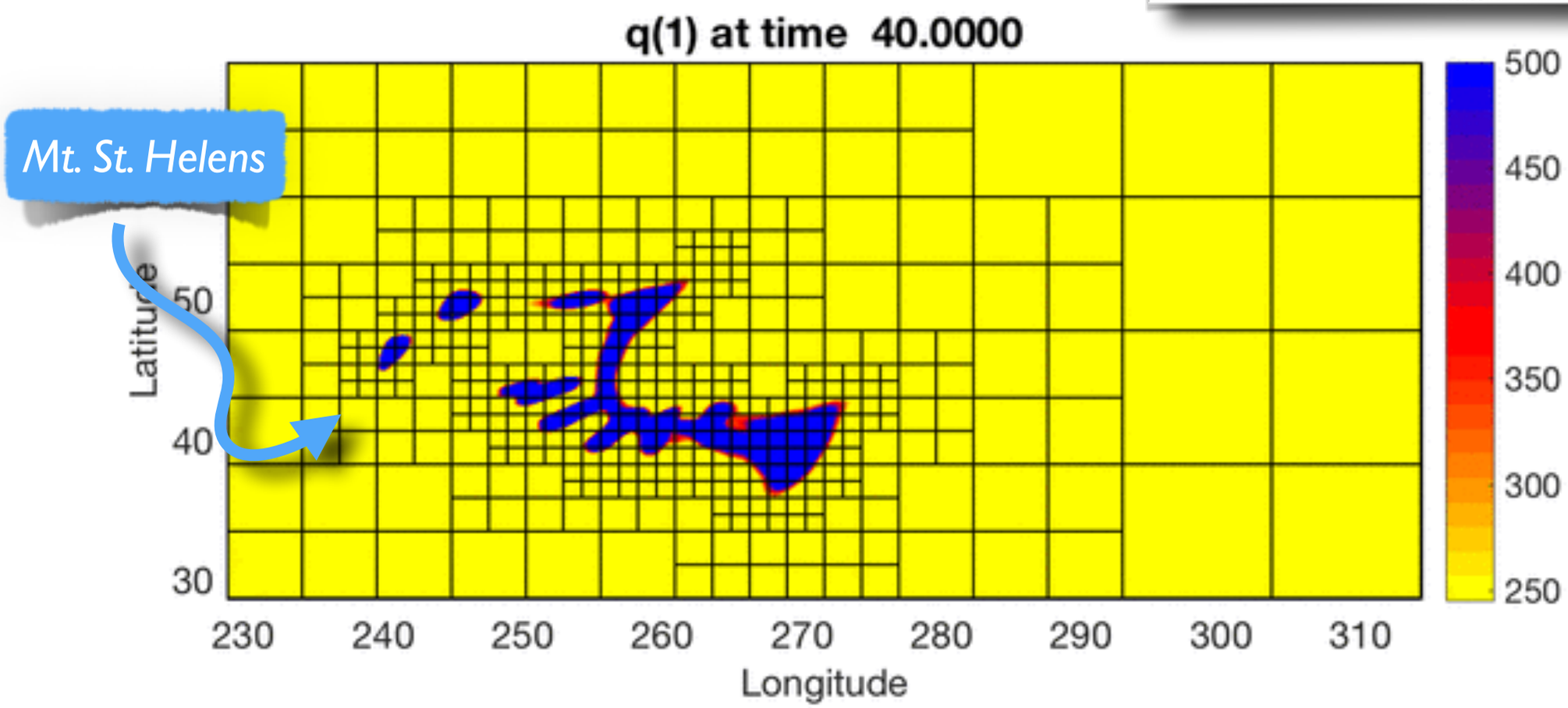
```
SUBROUTINE store_ptrs(fc_concen_ptr)
    USE iso_c_binding
    USE ash3d
    IMPLICIT NONE

    TYPE(C_PTR) :: fc_concen_ptr

    fc_concen_ptr = C_LOC(concen)
END
```

```
SUBROUTINE copy_ptrs2mod(fc_concen_ptr)
    USE iso_c_binding
    USE ash3d
    IMPLICIT NONE

    TYPE(C_PTR) :: fc_concen_ptr
    REAL(KIND=8), POINTER :: ptr(:)

    CALL C_F_POINTER(fc_concen_ptr,ptr,[1])
    concen(0:n+1,0:n+1,0:n+1,2) => ptr
END
```

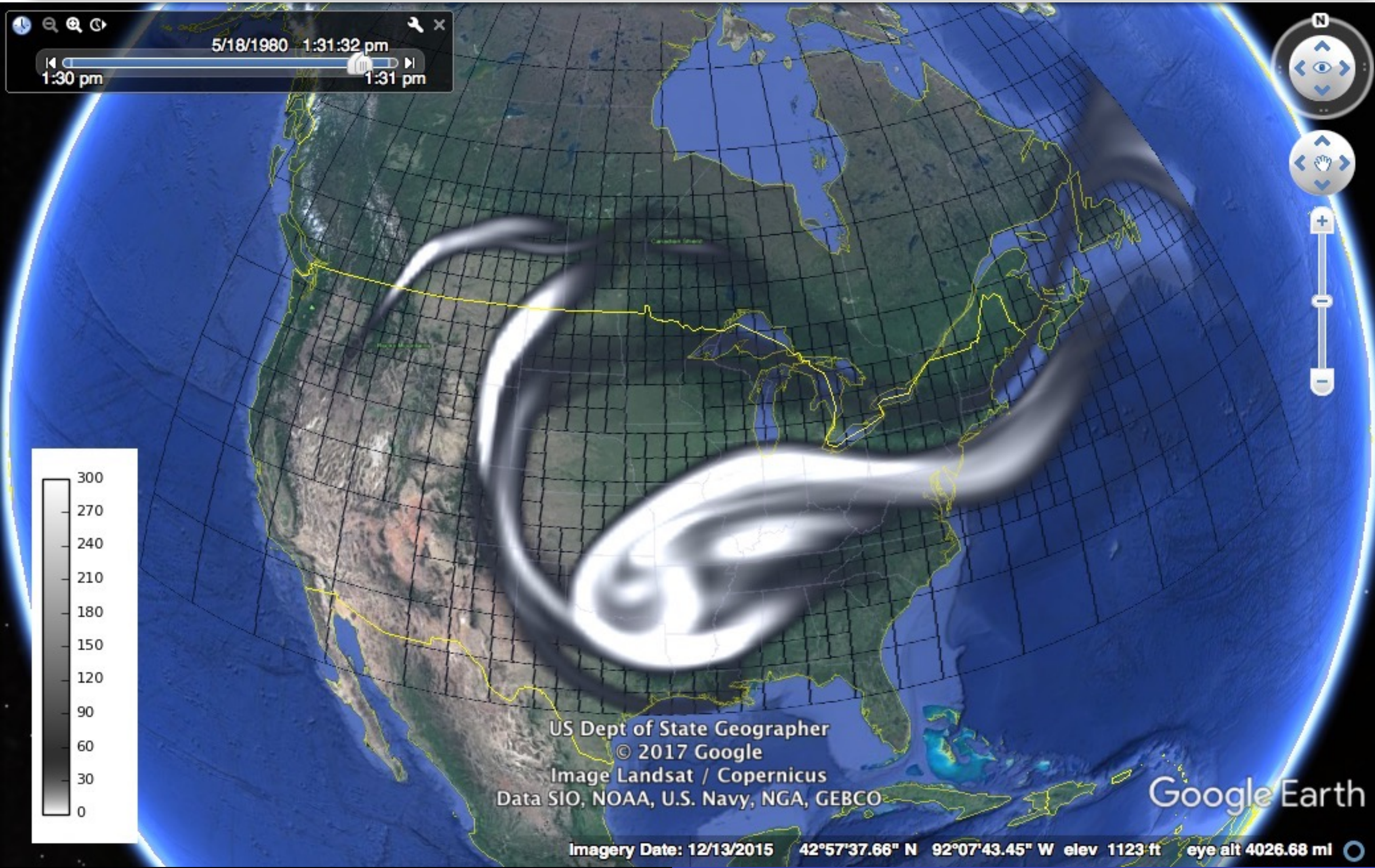
Adaptive Ash3d

- 96 hour simulation
- 1024 x 512 effective resolution
- 3-4 times faster than single grid

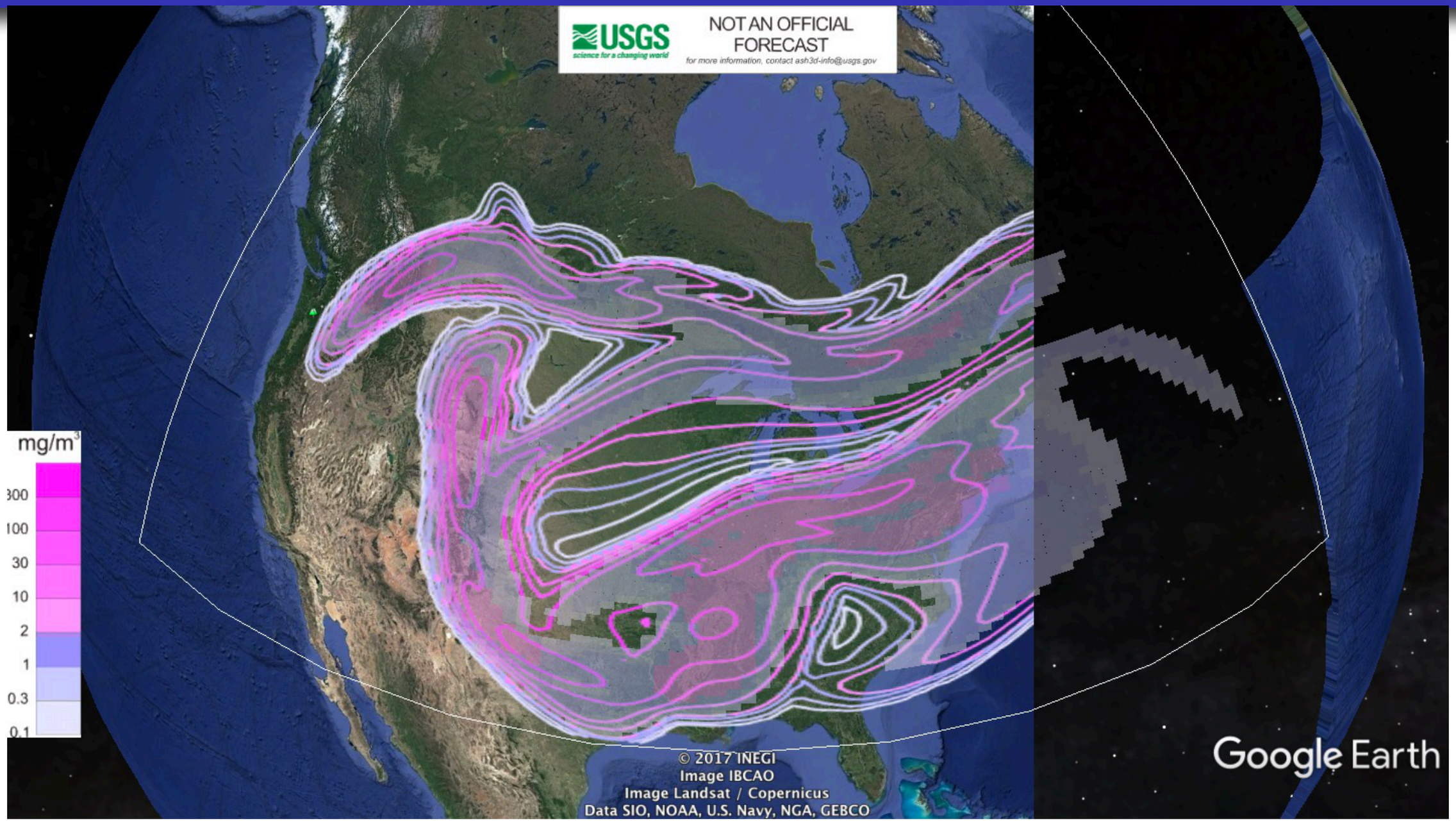




Adaptive results

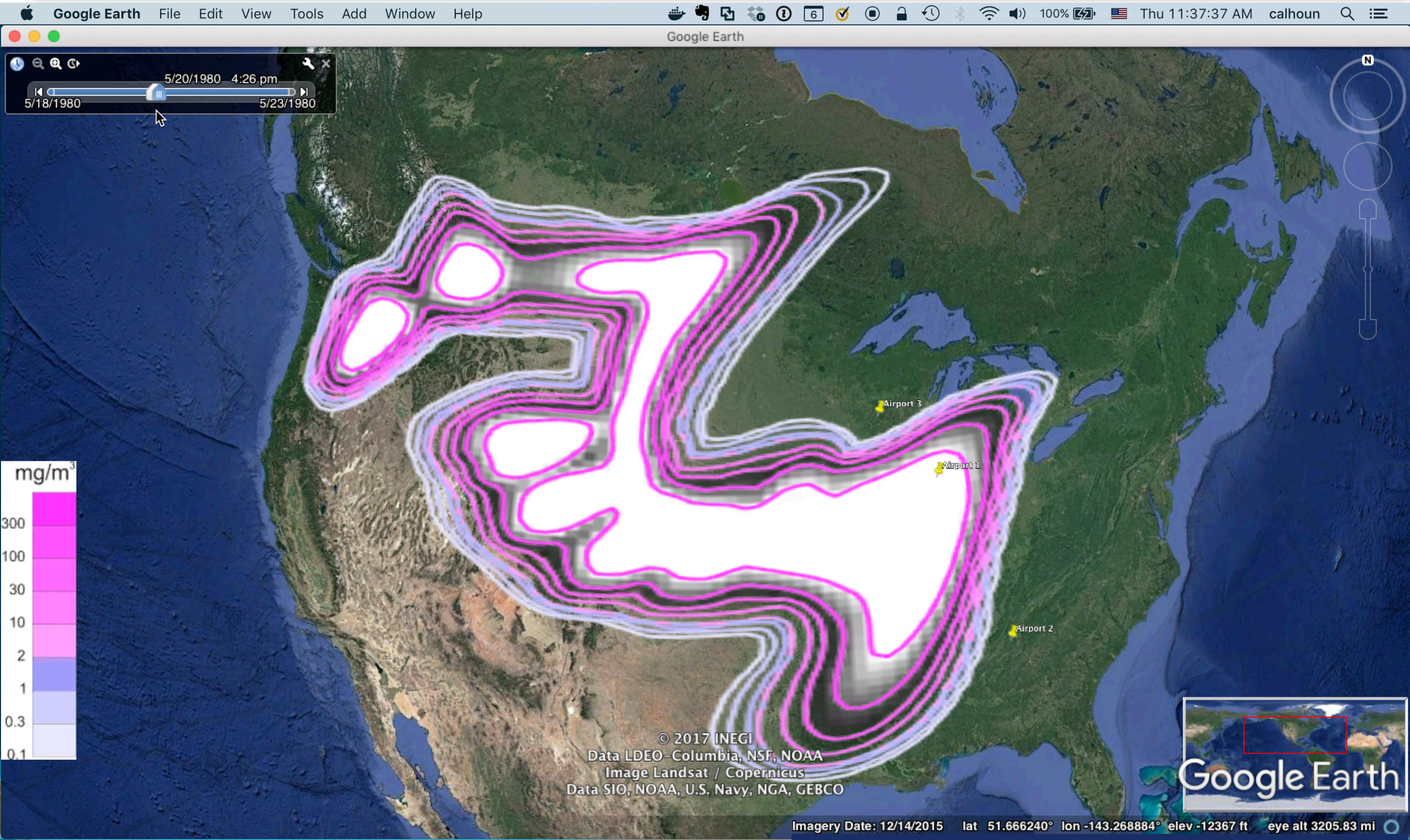


Single grid comparison



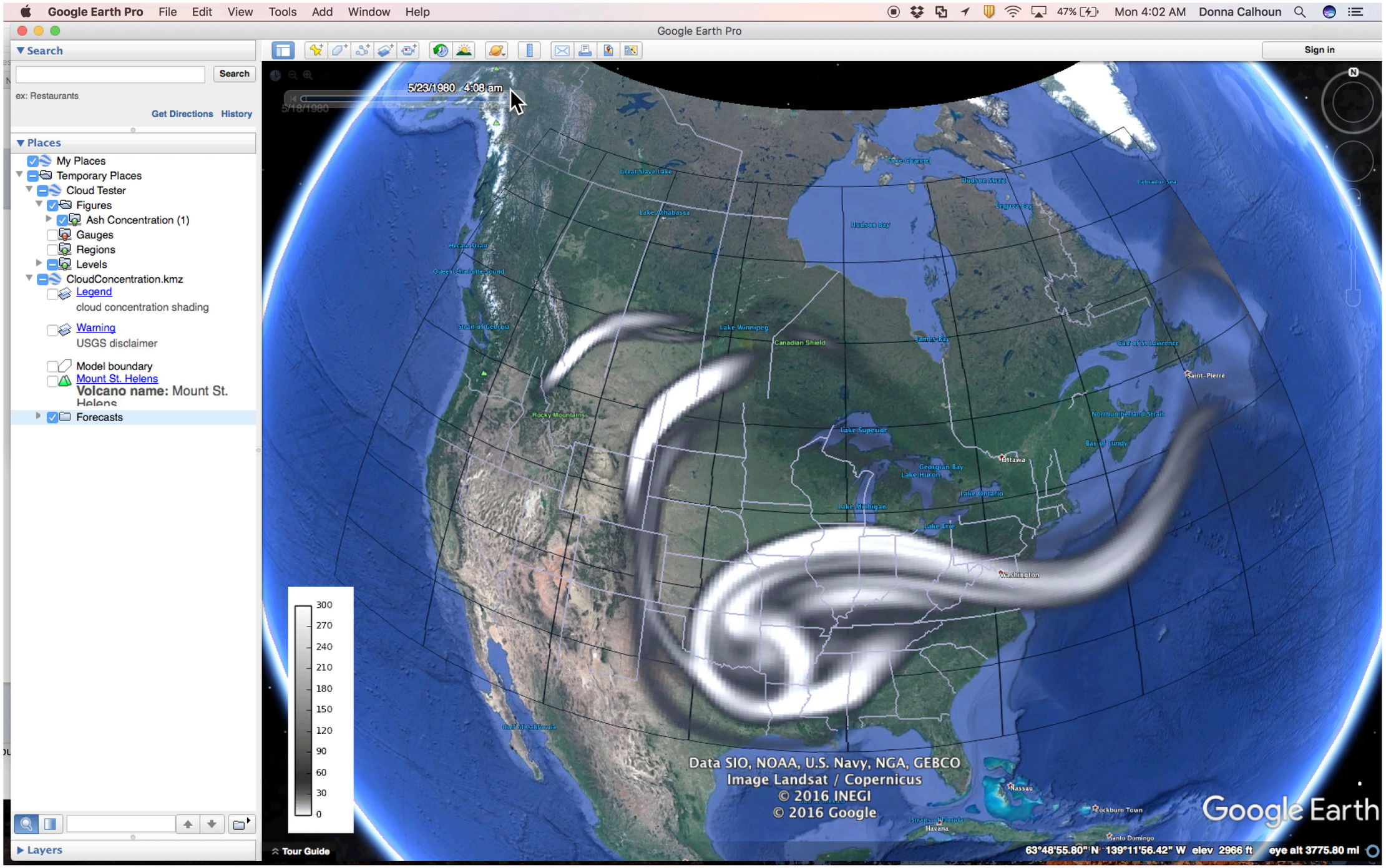
- Comparison between original Ash3d (pcolor plot) and ForestClaw (contours).
- Calculations done on a single 128x128 grid

Single grid comparison

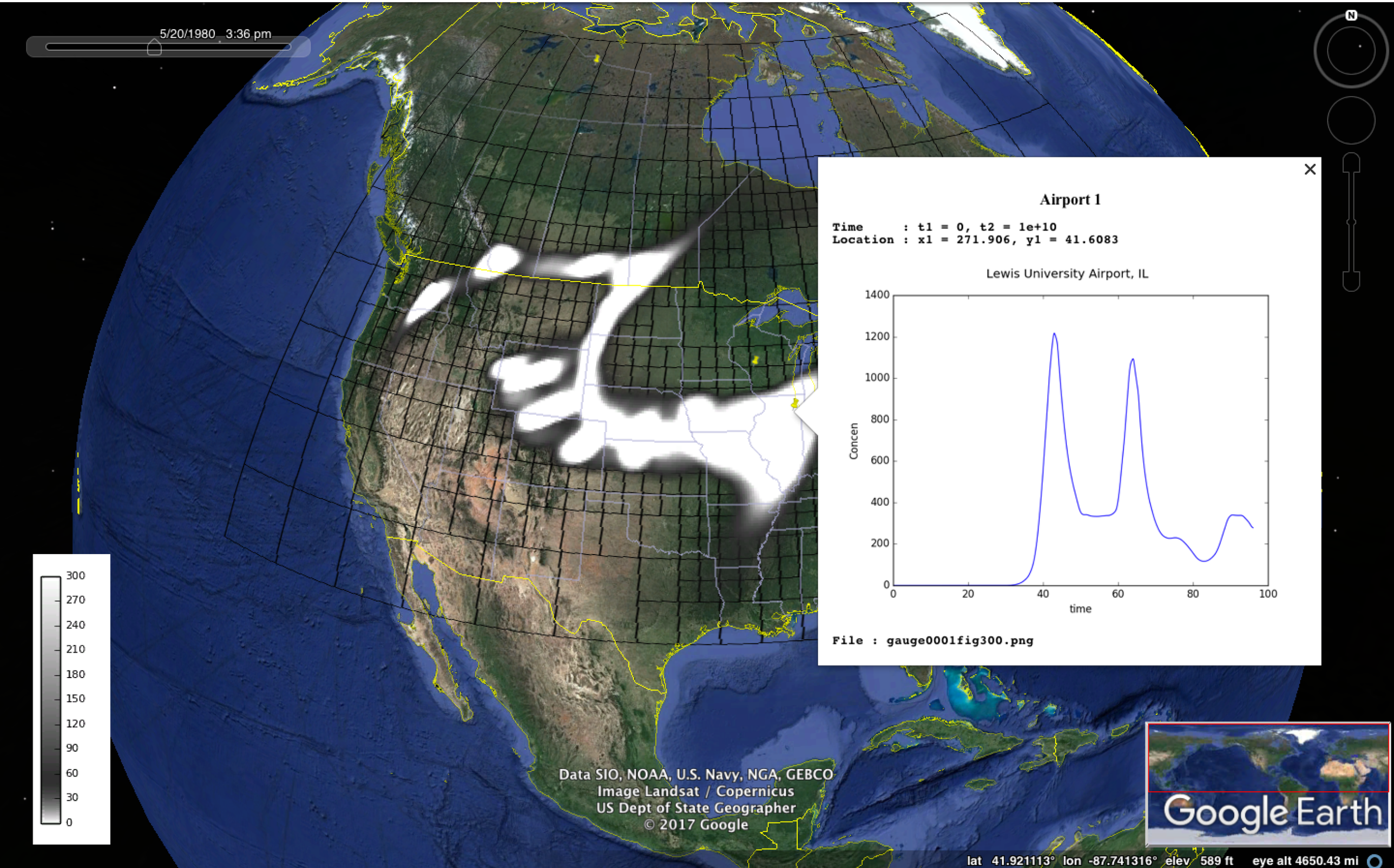


* Single grid to show smooth contours

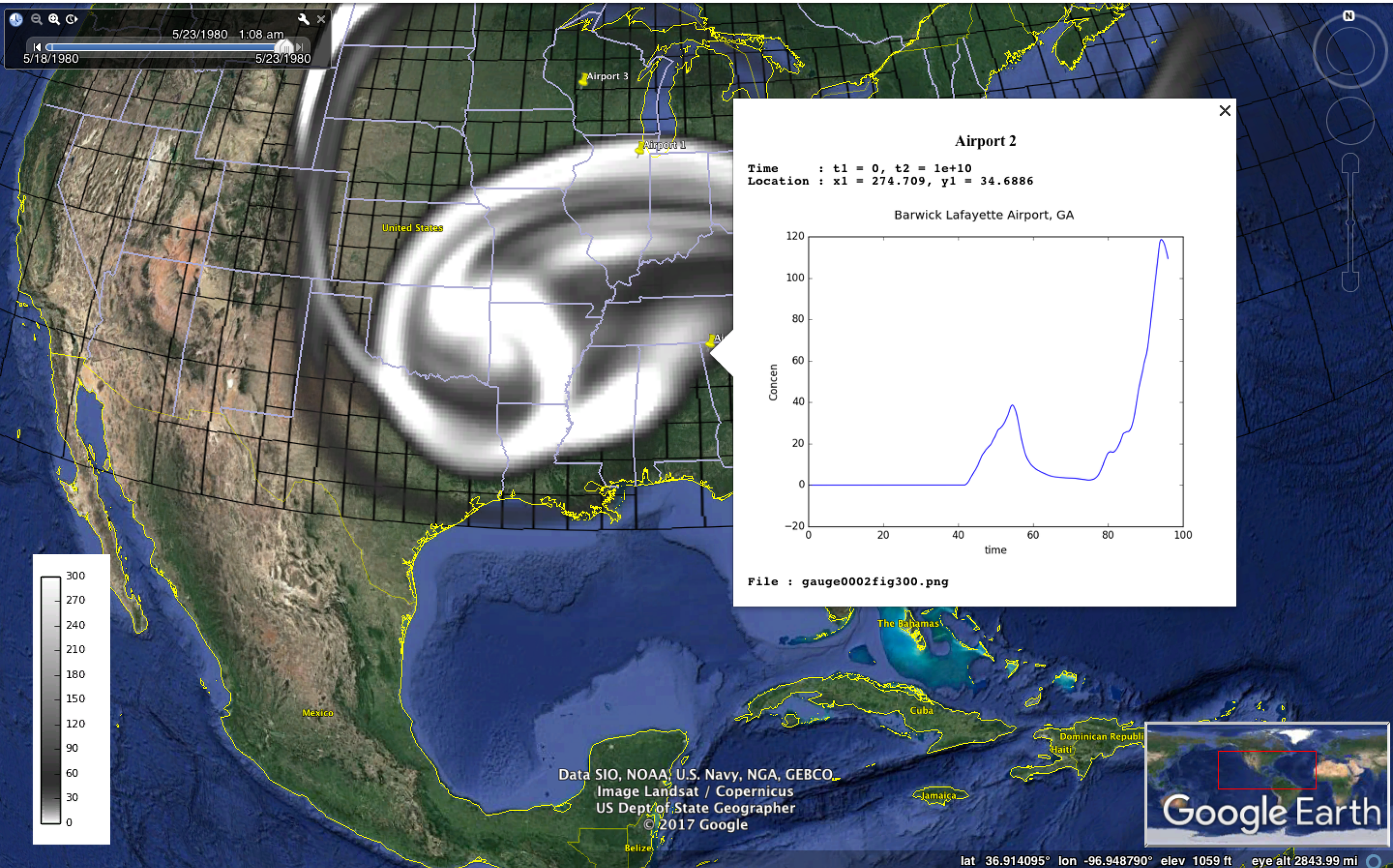
Single grid comparison



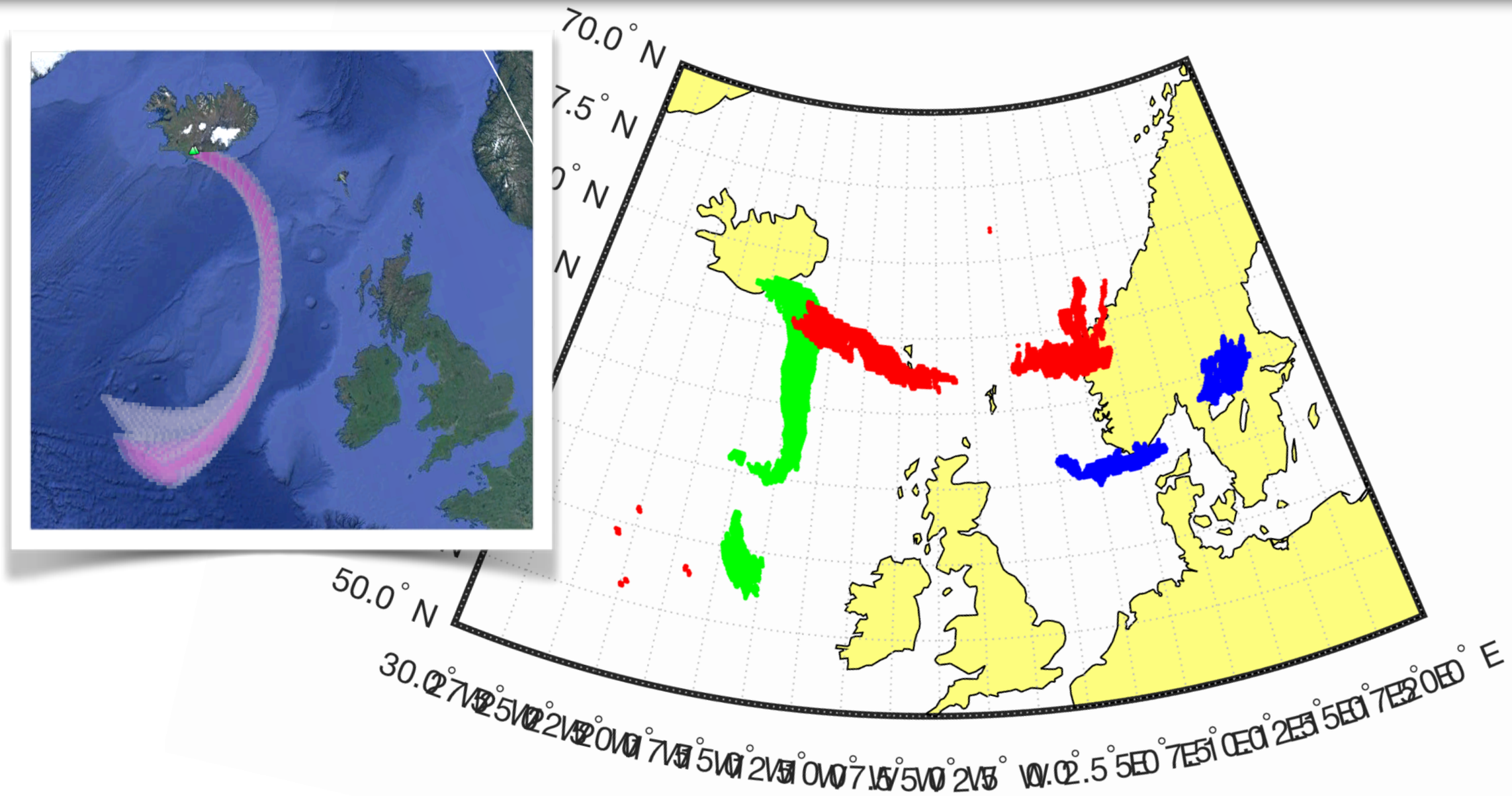
Airports


5/20/1980 3:36 pm
Airport 1
Time : t1 = 0, t2 = 1e+10
Location : x1 = 271.906, y1 = 41.6083
Lewis University Airport, IL
Concen
time
File : gauge0001fig300.png
300
270
240
210
180
150
120
90
60
30
0
Data SIO, NOAA, U.S. Navy, NGA, GEBCO
Image Landsat / Copernicus
US Dept of State Geographer
© 2017 Google
Google Earth
lat 41.921113° lon -87.741316° elev 589 ft eye alt 4650.43 mi

Airports


5/23/1980 1:08 am
5/18/1980
5/23/1980
Airport 3
Airport 1
United States
Mexico
The Bahamas
Cuba
Dominican Republi
Haiti
Jamaica
Belize
Data SIO, NOAA, U.S. Navy, NGA, GEBCO
Image Landsat / Copernicus
US Dept of State Geographer
© 2017 Google
300
270
240
210
180
150
120
90
60
30
0
Airport 2
Time : t1 = 0, t2 = 1e+10
Location : x1 = 274.709, y1 = 34.6886
Barwick Lafayette Airport, GA
Concen
time
120
100
80
60
40
20
0
−20
0
20
40
60
80
100
File : gauge0002fig300.png
Google Earth
lat 36.914095° lon -96.948790° elev 1059 ft eye alt 2843.99 mi

Iceland (satellite data)



Satellite data available for comparison

Future plans

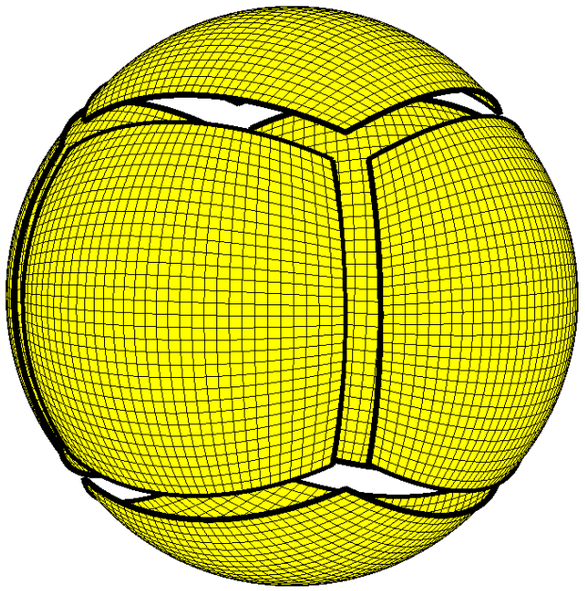
- Extend to cubed sphere grid for problems over the poles (difficult with lat-long grids)
- Consider other numerical schemes to handle conservation (f-waves, for example).
- Performance testing, and effects of IO

Tracer transport is an important component in larger weather and climate models and can often take up the the majority of the computational time

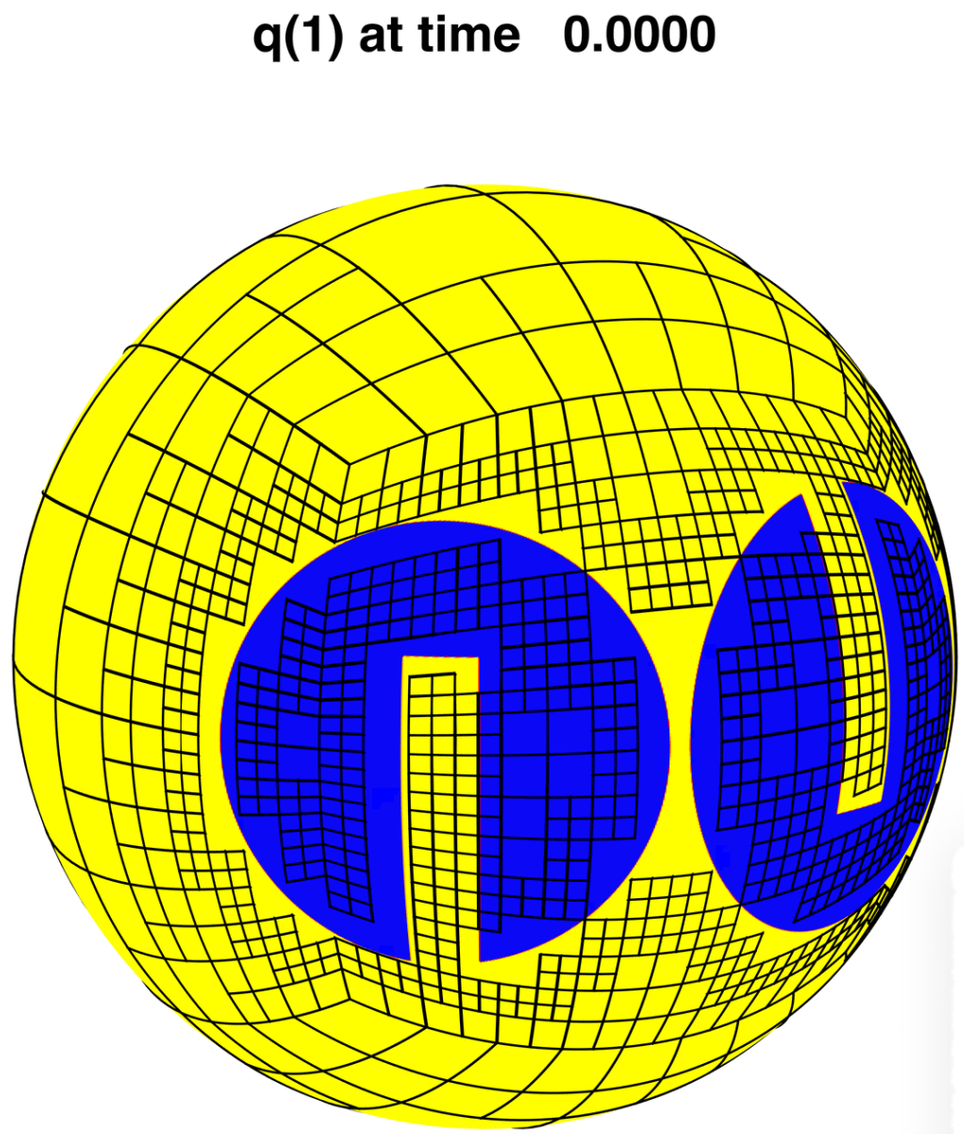
- Other transport problems : smoke, pollution, vapor, C02, and so on.

www.forestclaw.org www.github.com/ForestClaw

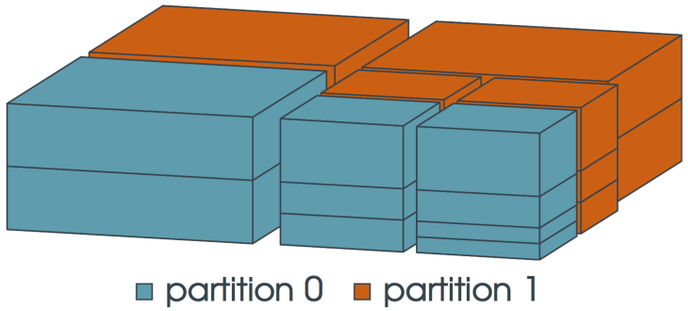
ForestClaw cubed sphere mesh



Cubed sphere grid in ForestClaw



Horizontal scale ~ O(1000km)
Vertical scale ~O(10km)



Extrude cubed-sphere to 3d and use p6est (T. Isaac)