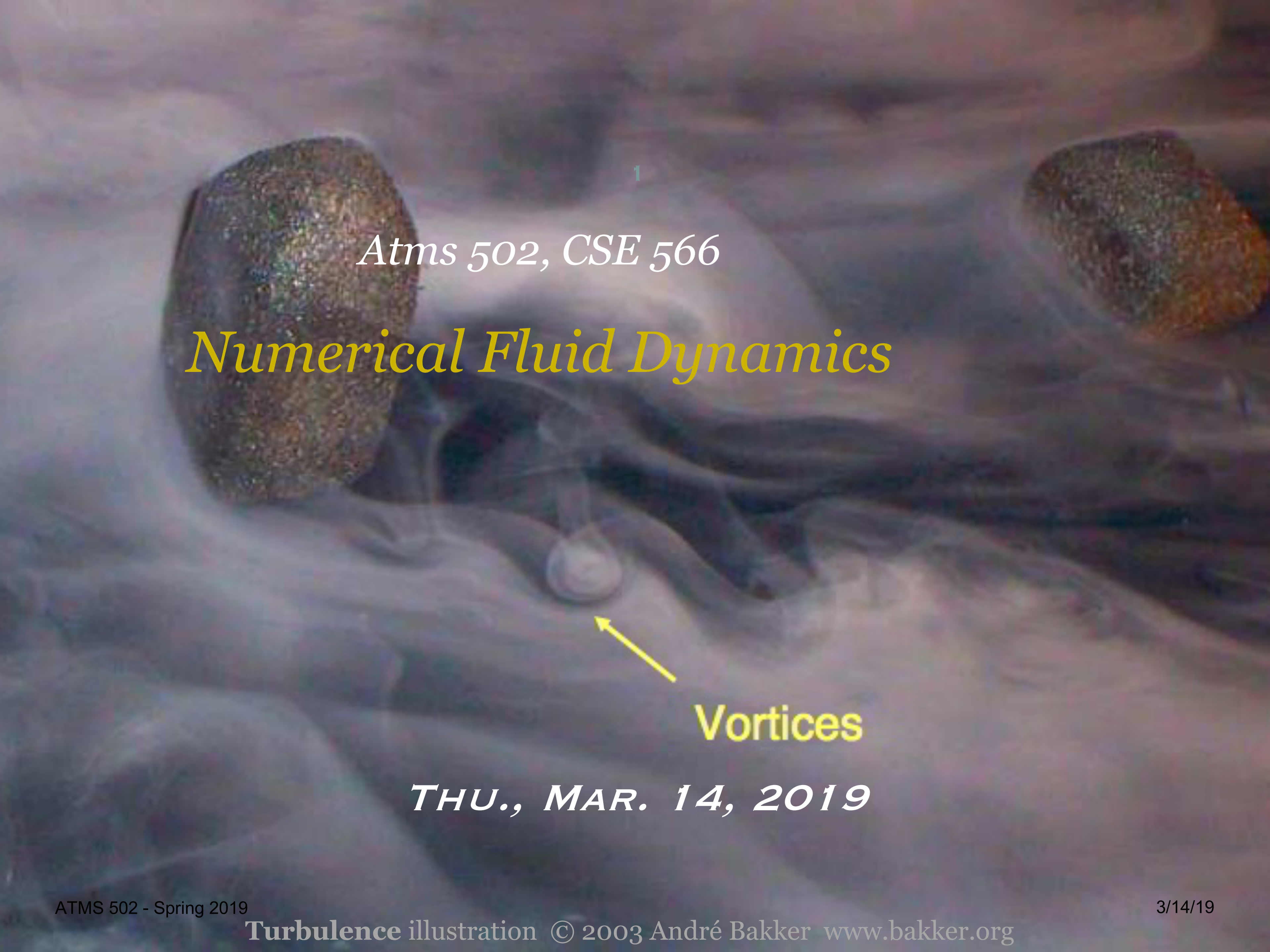
Atms 502, CSE 566

Numerical Fluid Dynamics

Thu., Mar. 14, 2019



ATMS 502
CSE 566

Thursday,
14 March 2019

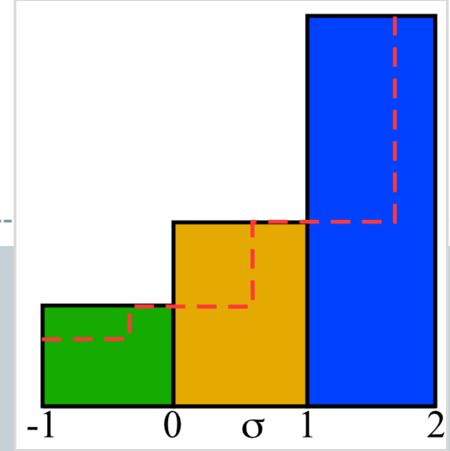
Class #18

Plan for Today

- 1) Van Leer, continued
  - Piecewise linear; monotonic slopes
- 2) Parallel performance
  - Continued; Amdahl's law
- 3) Workflows
  - Optimizing a sequence of tasks
- 4) Visualization critique
  - Study of a modeled severe storm

van Leer, continued

Review: van Leer



- New value from integrating under piecewise constant function at time t that *will be in grid zone [0,Δx] at t+Δt*.

$$q^{n+1} \equiv \overline{q}^{1/2} = \int_{0}^{1-\sigma} q_{1/2}\,dx + \int_{-\sigma}^{0} q_{-1/2}\,dx \quad \sigma = \frac{u\Delta t}{\Delta x}$$

Grid-point value f(j) represents the average of the function over the grid cell (see Durran, § 1.3.1, p. 27)

- Piecewise linear

$$\overline{q}^{1/2} = \overline{q}_{1/2} - \sigma\left(\overline{q}_{1/2} - \overline{q}_{-1/2}\right) - \frac{\sigma}{2}(1-\sigma)\left(\overline{\Delta}_{1/2}q - \overline{\Delta}_{-1/2}q\right)$$

$$\overline{q}_{1/2} = \int_0^1 q(x,t^0)\,dx \qquad \overline{\Delta}_{1/2}q = \frac{1}{2}\left(\overline{q}_{3/2} - \overline{q}_{-1/2}\right)$$

Implementing Piecewise Linear

- <u>Handling PL inside *advect1d*</u>:
  - In the x-advect or z-advect step: for the x-direction, prepare 1D arrays of a row of theta^n and u^n as input to your advect1D.
    - Inside advect1d, you compute Courant number, slopes and fluxes:

$$F_{i-1/2} = \begin{cases} r_{i-1/2}\left(q_{i-1}^n + \dfrac{1-r_{i-1/2}}{2}\Delta q_{i-1}^n\right), & \text{if } \mathrm{u}_{\text{i-1/2}}^{\text{n}} \geq 0; \\ r_{i-1/2}\left(-q_i^n + \dfrac{1-r_{i-1/2}}{2}\Delta q_i^n\right), & \text{if } \mathrm{u}_{\text{i-1/2}}^{\text{n}} < 0; \end{cases} \quad \text{where} \quad \begin{cases} r_{i-1/2} = \left|\dfrac{\Delta t}{\Delta x} u_{i-1/2}^n\right| \\ \Delta q_i = \dfrac{q_{i+1}^n - q_{i-1}^n}{2} \end{cases}$$

    - Then you can do the integration, still in the 1-D advection code:

$$q_i^{n+1} = q_i^n - \left(F_{i+1} - F_i\right) + \frac{\Delta t}{\Delta x} q_i^n \left(u_{i+1} - u_i\right)$$

PL: monotonic slope limiter

- Simple centered differences

$$\bar{\Delta}_{1/2} q = \frac{1}{2}\left(\bar{q}_{3/2} - \bar{q}_{-1/2}\right)$$

- In English:

$$\bar{\Delta}_{1/2} q = \frac{\left(q(i+1) - q(i-1)\right)}{2}$$

- This is *average slope* in the grid zone
  - *Not monotonic*

- ***monotonic** slope form*

$$IF \ \left(q_i - q_{i-1}\right)\left(q_{i+1} - q_i\right) \geq 0,$$

$$\Delta\theta_i = \operatorname{sgn}\left(q_{i+1} - q_{i-1}\right) \times \min\begin{pmatrix} \left|q_i - q_{i-1}\right|, \\ \left|q_{i+1} - q_i\right|, \\ \left|q_{i+1} - q_{i-1}\right|/2 \end{pmatrix}$$

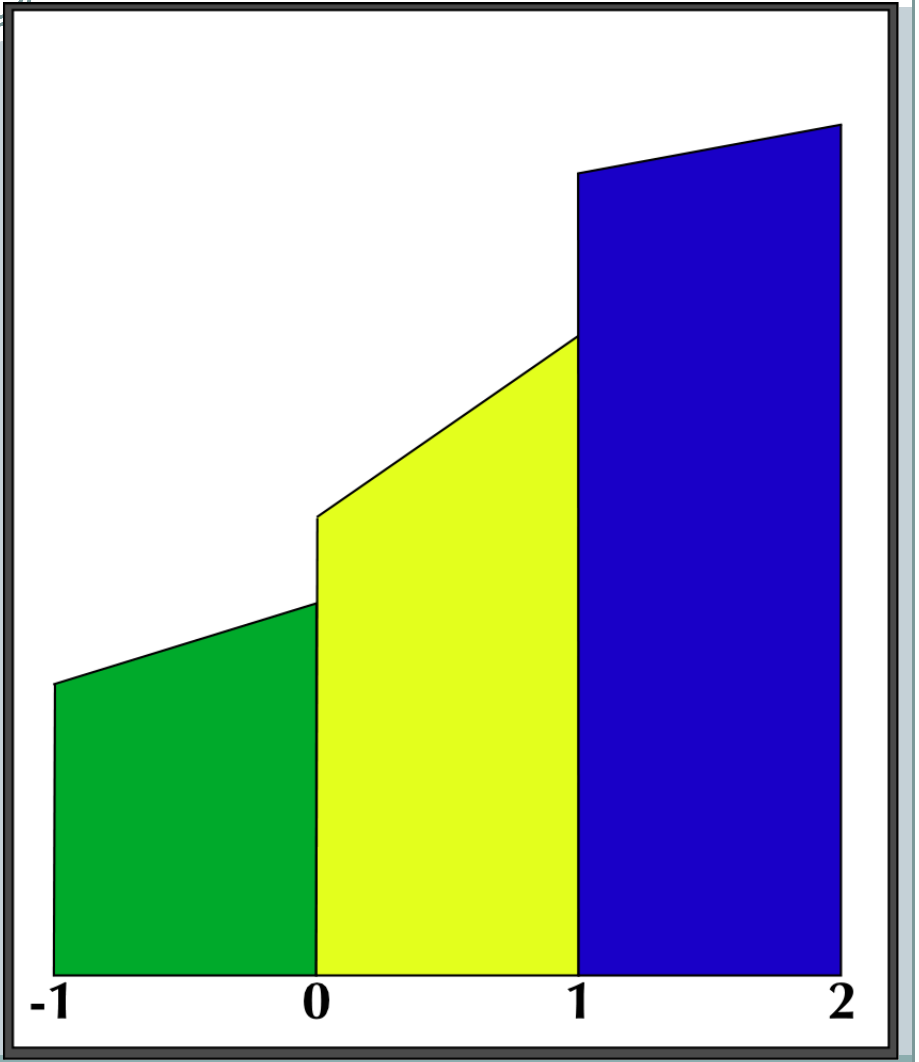
$$Otherwise: \quad \Delta q_i = 0$$

This is known as the "minmod" flux limiter (see Durran, § 5.5.2, p. 230)

Piecewise linear form

$$\overline{q}^{1/2} = \int_0^{1-\sigma} q_{1/2}\,dx + \int_{-\sigma}^{0} q_{-1/2}\,dx$$

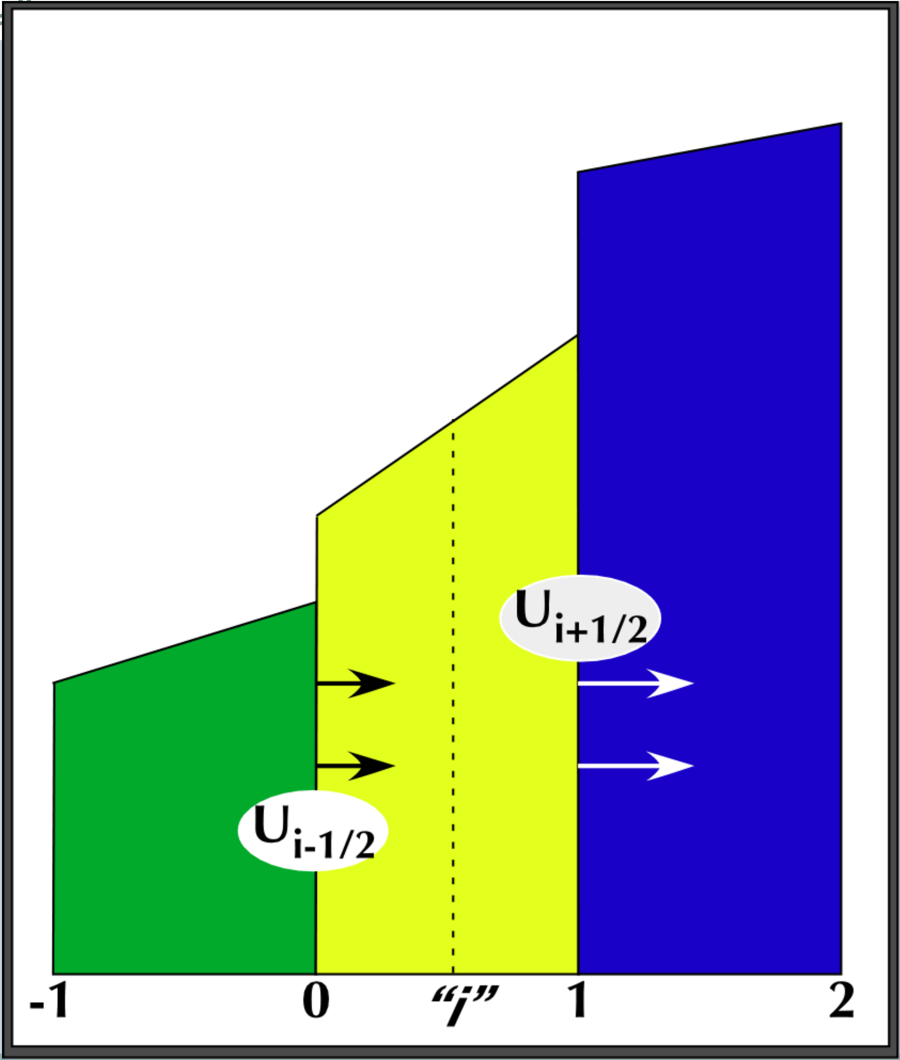
- Key concept:
  - integrate under function at time t *that will be in zone* [0,Δx] *at t+Δt*.



Piecewise linear form

$$\bar{q}^{1/2} = \int_{0}^{1-\sigma} q_{1/2}\,dx + \int_{-\sigma}^{0} q_{-1/2}\,dx$$

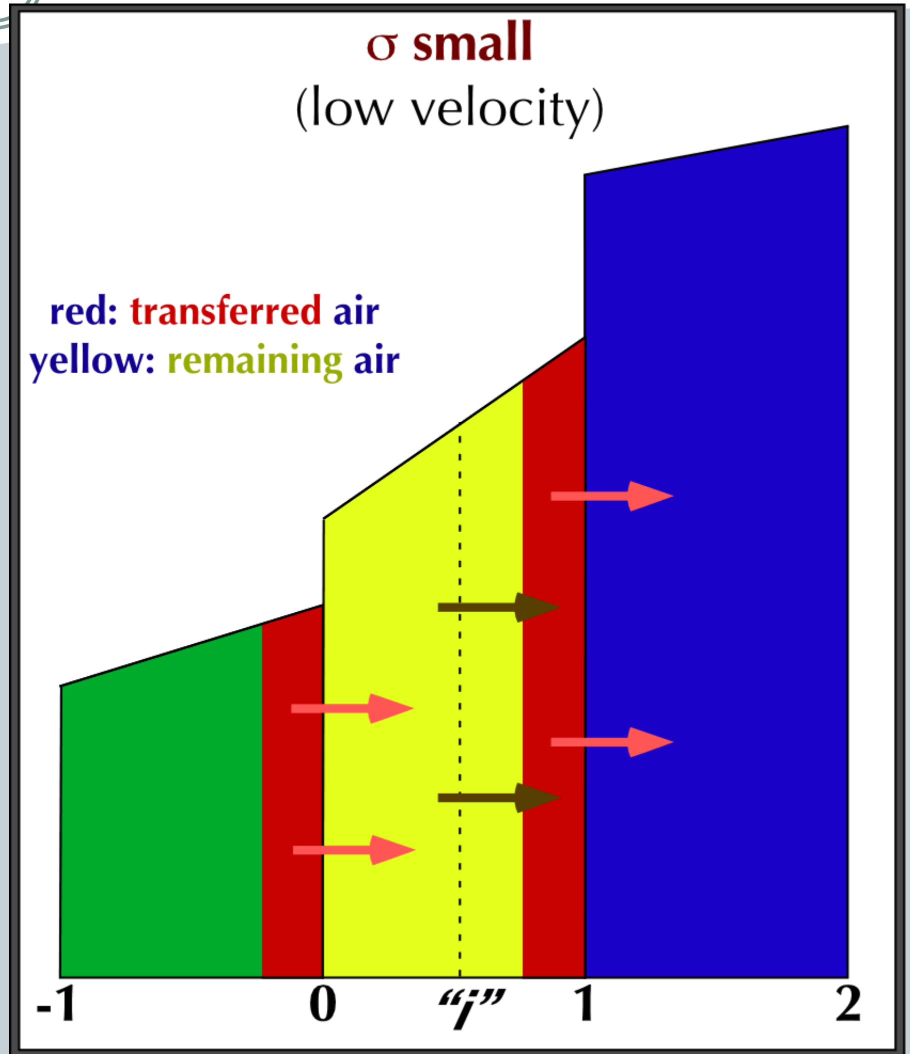
- Fluxes
  - integrate under function at time t *that will be in zone* [0,Δx] *at t+Δt*.
  - Grid zone boundary velocities (C-grid) shown



Piecewise linear form

$$\bar{q}^{1/2} = \int_0^{1-\sigma} q_{1/2}\,dx + \int_{-\sigma}^{0} q_{-1/2}\,dx$$

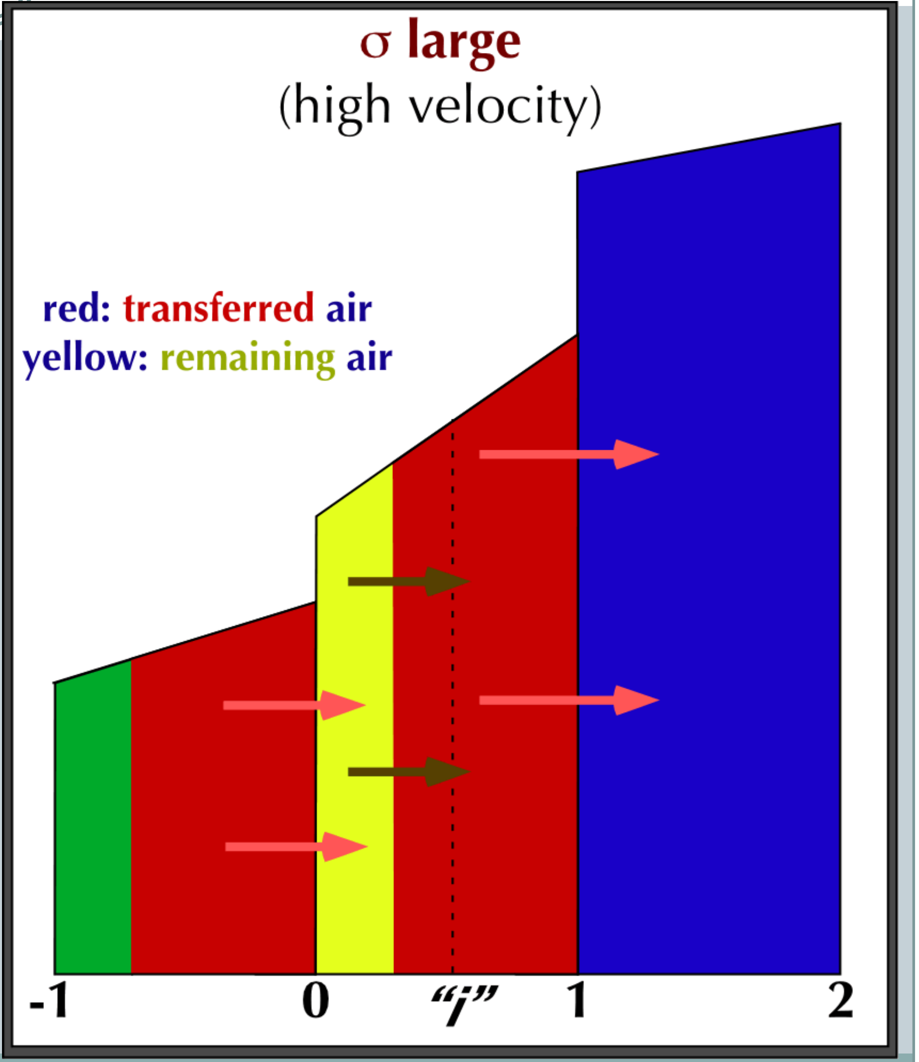
- Fluxes
  - Small σ: area in red transferred; yellow remains in the zone [0,Δx]



Piecewise linear form

$$\bar{q}^{1/2} = \int_0^{1-\sigma} q_{1/2} dx + \int_{-\sigma}^{0} q_{-1/2} dx$$

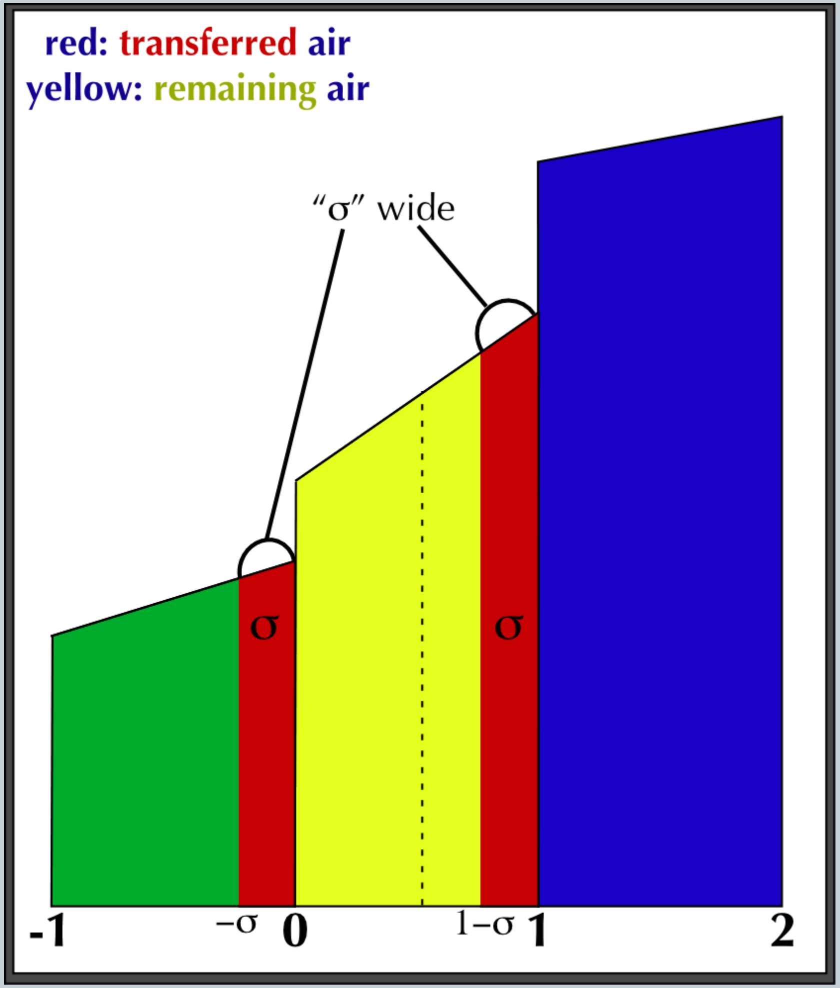
- Fluxes
  - High σ: area in red transferred; yellow remains in the zone [0,Δx]



van Leer, integration

$$\bar{q}^{1/2} = \int_0^{1-\sigma} q_{1/2}\,dx + \int_{-\sigma}^{0} q_{-1/2}\,dx$$

- Integration bounds
  - *Current* area [0,1-σ] remains in zone.
  - *Upstream* area [-σ,0] is transferred into the zone [0,1].
- Local function
  - is independently determined for each grid zone.



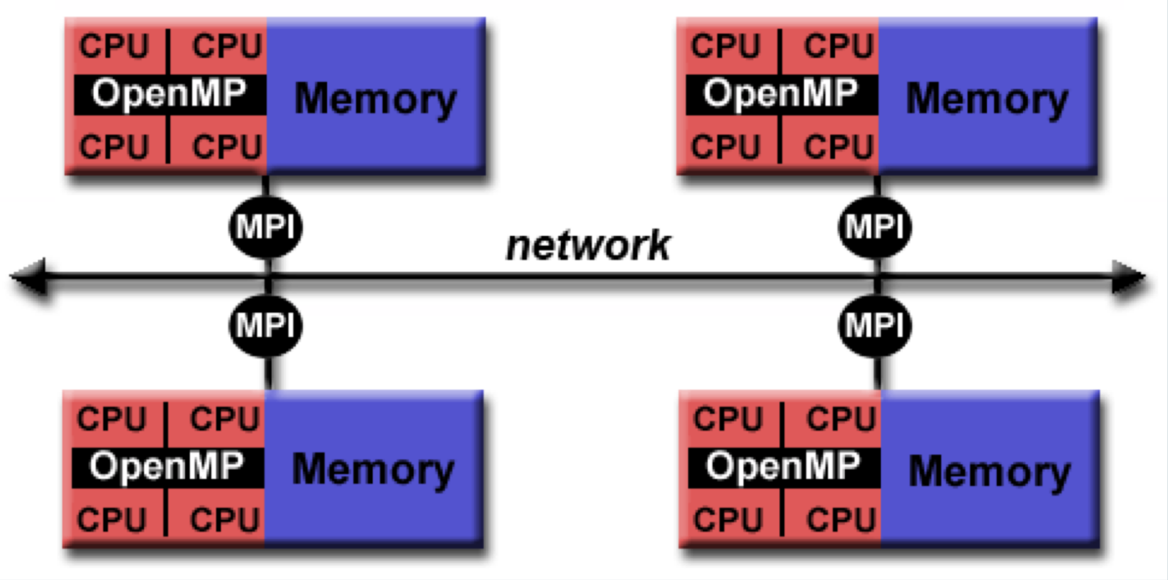
Continued: Parallel performance

Some figures from
High Performance Computing
by David Kuck (Oxford Press, NY)

Others: LLNL pages on parallel computing:
https://computing.llnl.gov/tutorials/parallel_comp/

Hybrid architectures & parallelism

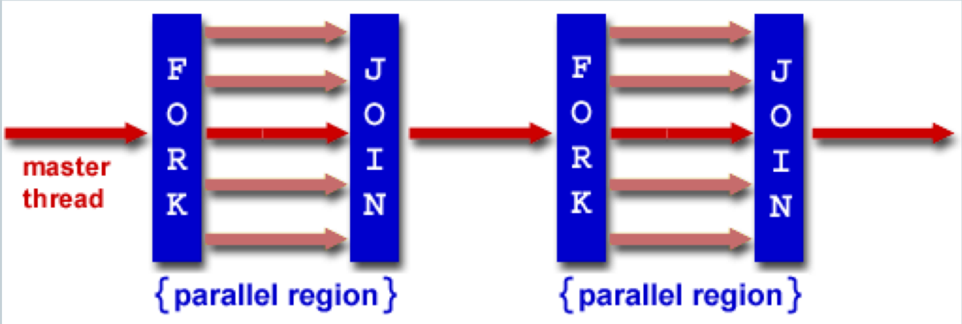
- **Hybrid parallelism** combines OpenMP threads model with the message passing (MPI) model
  - threads handle local (on-node) data
  - communication between nodes is done via MPI



 all from: https://computing.llnl.gov/tutorials/parallel_comp/

Program #6: parallelization

- Insert OpenMP directives




```
!$OMP   PARALLEL DO PRIVATE (i,j,k)
        do k = 2,nz-1
         do j = 1,ny
          do i = 1,nx
            u3(i,j,k) = u3(i,j,k) + tstep*diff*( u1(i+1,j,k)...
          enddo
         enddo
        enddo
!$OMP   END PARALLEL DO
```

Parallel performance

- Grid zones/second
  - how many grid points computed per wallclock second

$$\text{Zps} = \frac{\text{NX} \bullet \text{NY} \bullet \text{NZ} \bullet (\text{simulation time})/\Delta t}{\text{wallclock of parallel execution}}$$

  - somewhat higher with larger data size
  - strong function of #cpu
  - can use to 'scale up' a known problem size
- What is *speedup*?

$$\text{Sp} = \frac{\text{wallclock of serial execution}}{\text{wallclock of parallel execution}}$$

  - What do we expect to happen to speedup as we add more processors?

Performance vs. # processors

- Small #cpu:
  - Good speedup as increase #cpu
- More:
  - Falloff as cores don't have enough data = *not enough work to do*
- Many cpu:
  - limited by the system communication σ

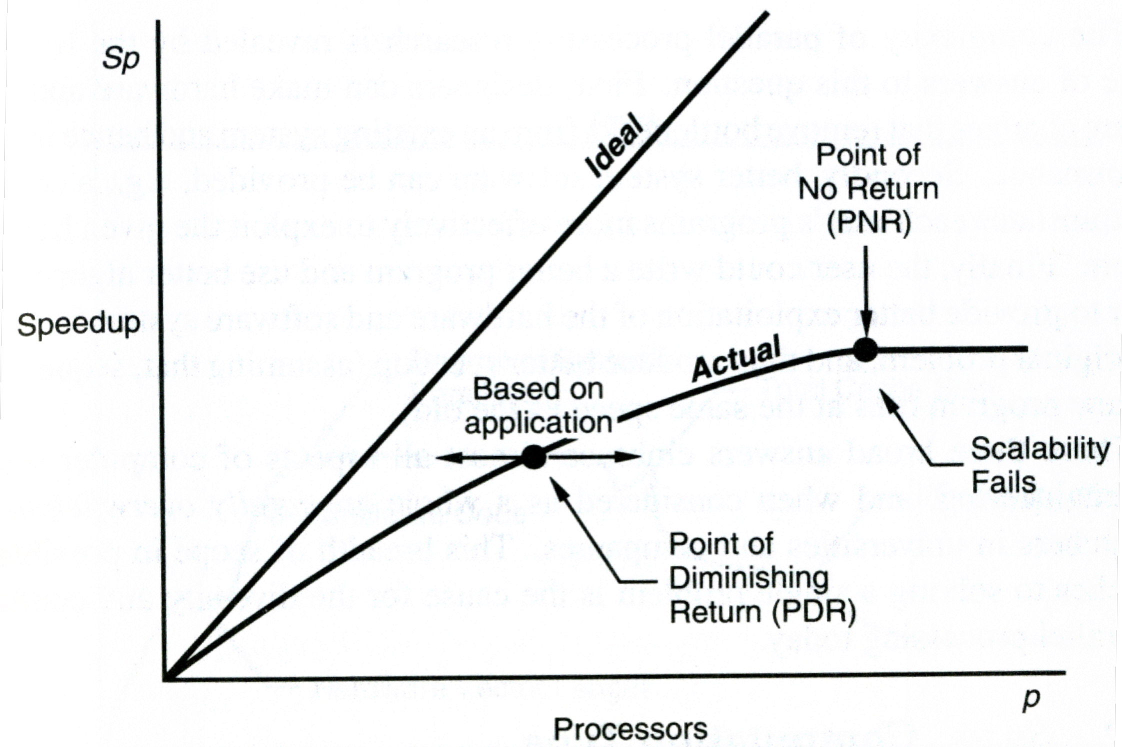


Figure 5.1. Parallel System Performance *Sp vs. p*

Performance vs. data size

- What to do?
  - Max out vector length
    - *Vectorize more*
  - Max out # processors
    - *Parallelize more*
  - Max out cache
    - *Data locality*
  - Max out main memory (virtual memory I/O bound)
    - *Sets of virtual mem*

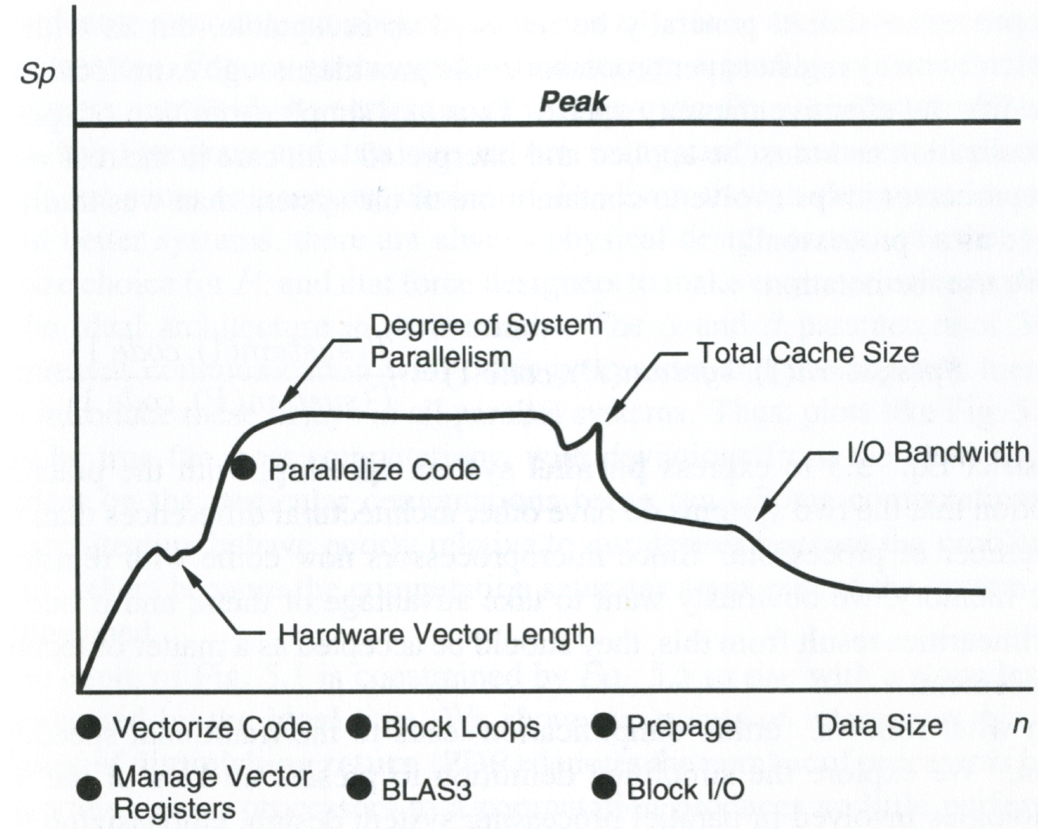


Figure 5.2. Architectural Saturation Sp *vs.* n

Parallel performance

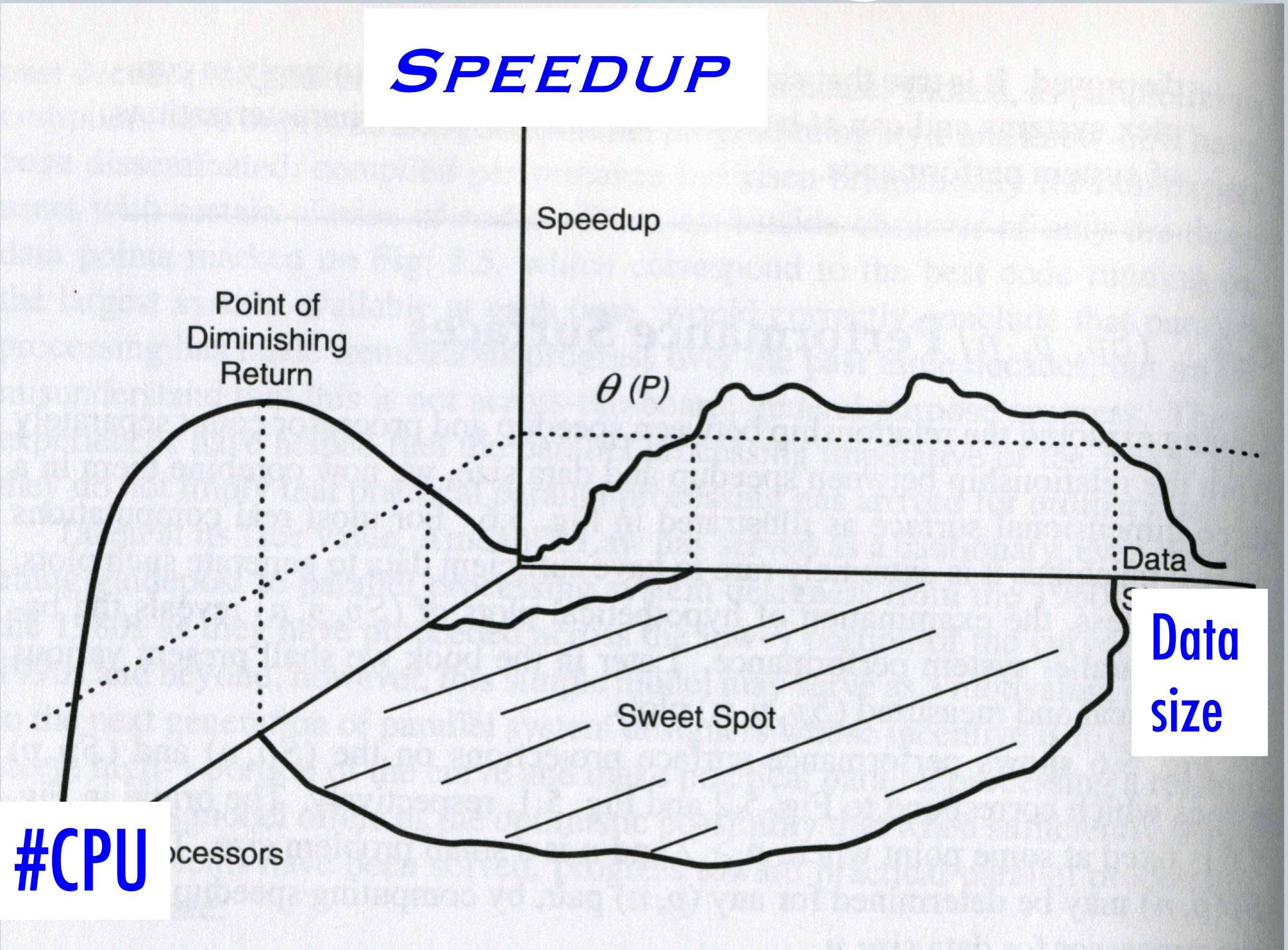


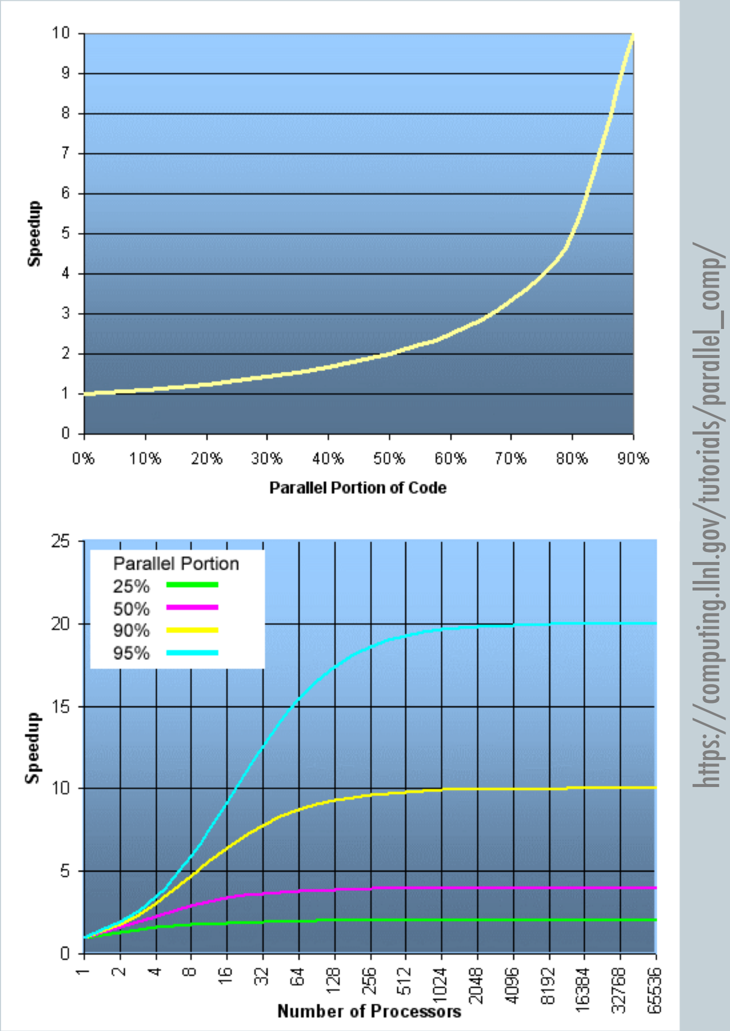
Figure 5.6. Performance Sweet Spot

Amdahl's Law

- Speedup *Sp* = T(1) / T(P)
  - … **but** not all of code parallelizable
  - Potential speedup defined by:
    - *fraction of code that can be parallelized*
- Amdahl's law:

$$Speedup = \frac{1}{\left(\frac{P}{N} + S\right)} = \frac{1}{1-P}$$

  - N = #cpu
  - P = parallel fraction
  - S =1-P = serial fraction
  - Efficiency may *vary with data size*



Amdahl's Law

- Amdahl's law:
  - N = #cpu
  - P = parallel fraction
  - S =1-P = serial fraction

$$Speedup = \frac{1}{\left(\frac{P}{N} + S\right)} = \frac{1}{1-P}$$

N	P=0.50	P=0.90	P=0.99	P=1.0
10	1.82	5.26	9.17	10
100	1.98	9.17	50.25	100
1,000	1.99	9.91	90.99	1,000
10,000	1.99	9.99	99.02	10,000

A. Wachsmann - Stanford

8 WRF runs @ 4 CPU/member = 32 CPU total; assume 30 min wallclock each
16 Post-Each @ 2 CPU/member = 32 CPU total; assume 60 min wallclock each
8 Post-All @ 2 CPU/member = 16 CPU total; assume 5 min wallclock each
2 STORE @ 2 CPU/member = 4 CPU total; assume 10 min wallclock *per run stored* each

Example of Workflows

Timeline

00:00 WRF-0 starts

00:30 WRF-0 done, *WRF-4 starts;* PostEach0 starts

01:30 PostEach0 done; PostAll0 starts

01:35 PostAll0 done, *PostAll tasks wait for work..*

00:30 WRF-4 starts

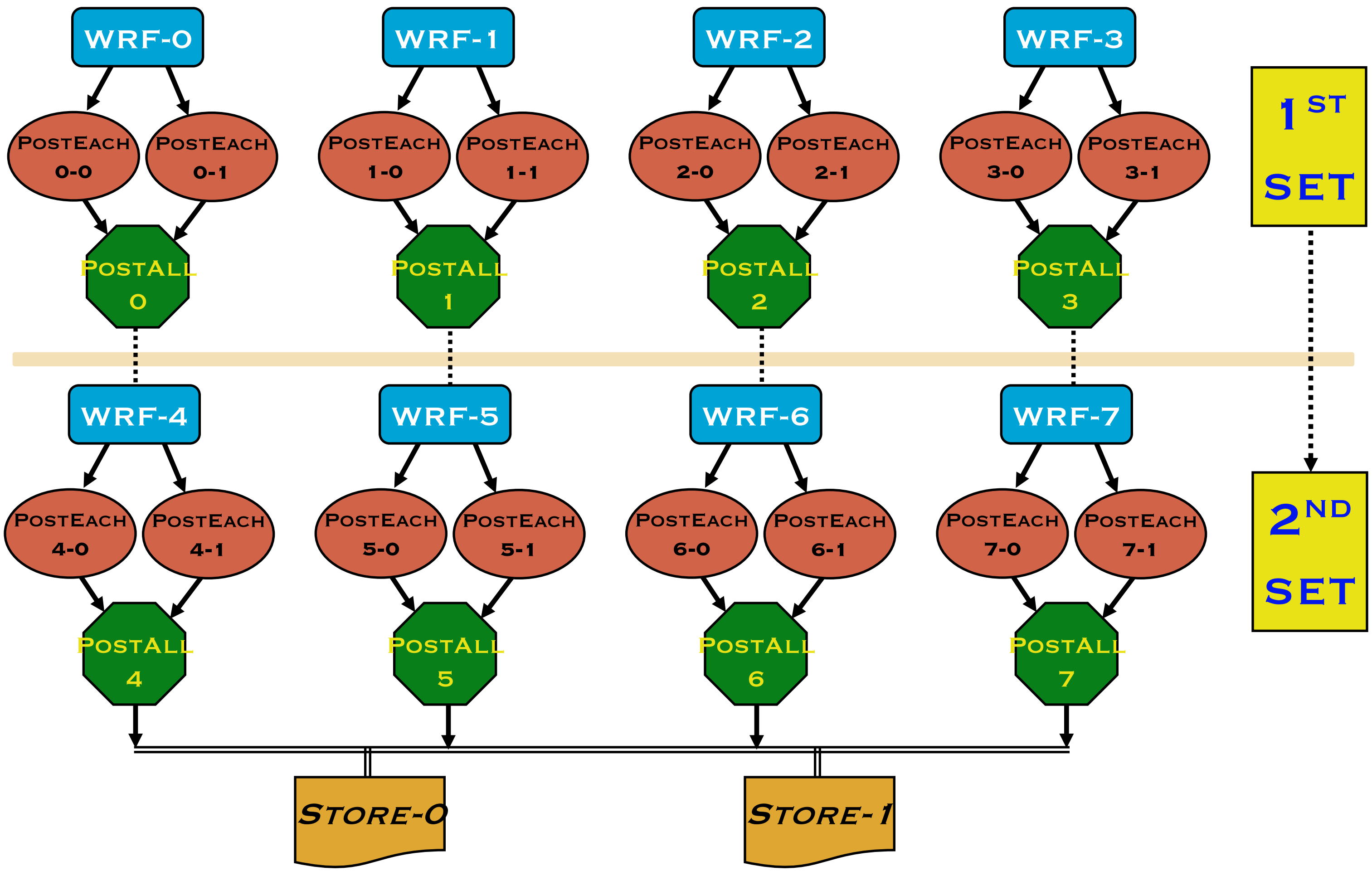
01:00 WRF-4 done, WRF tasks end //

01:30 PostEach4 starts

02:30 PostEach4 done, PostAll4 starts

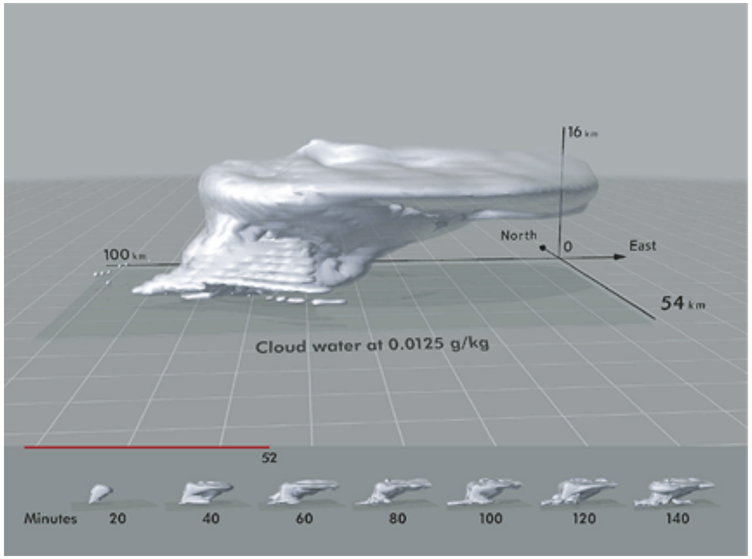
02:35 PostAll4 done; STORE tasks start…

03:15 STOREs finish.



Study of a Numerically Modeled Severe Storm

ANALYSIS AND CRITICISM



REFERENCE:

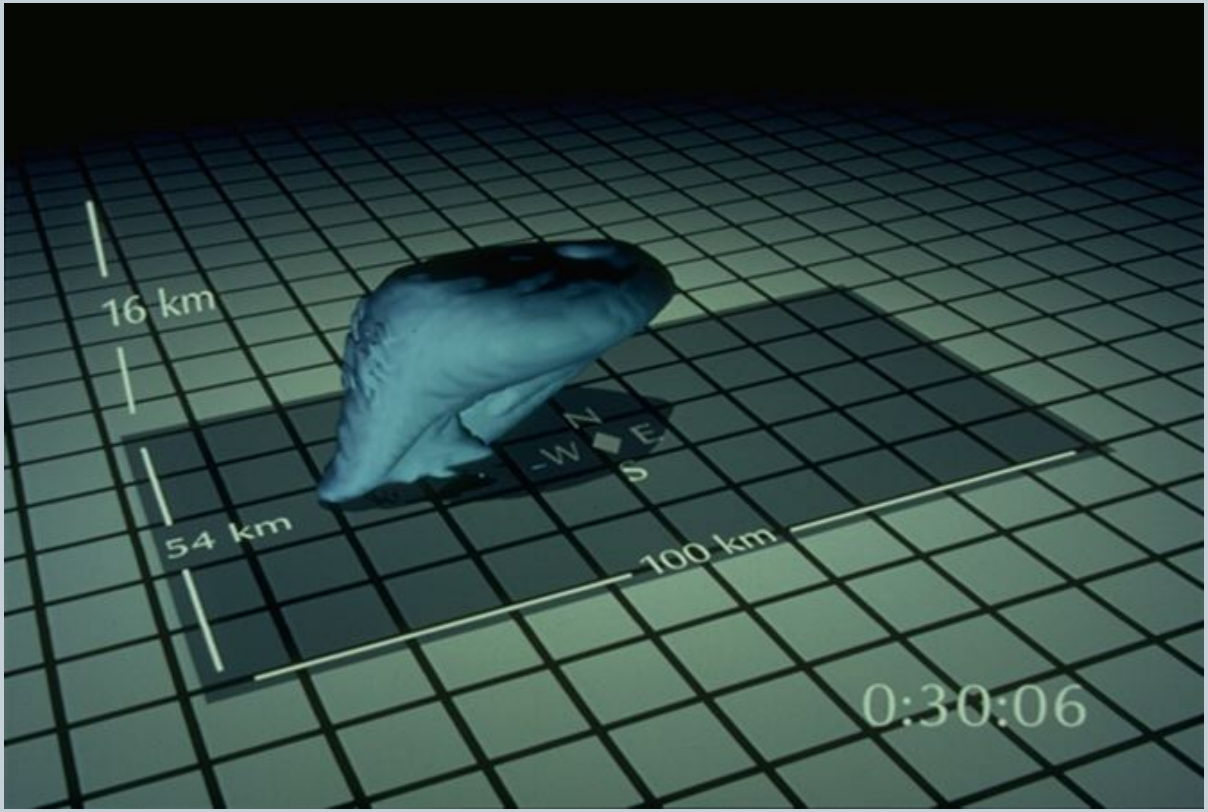
Edward Tufte, *Envisioning Information*, © 1990, Graphics Press -- *available at UI library online*

Modeled Storm video

- What visual idioms are used?
  - What is their purpose?
  - Are they used effectively?

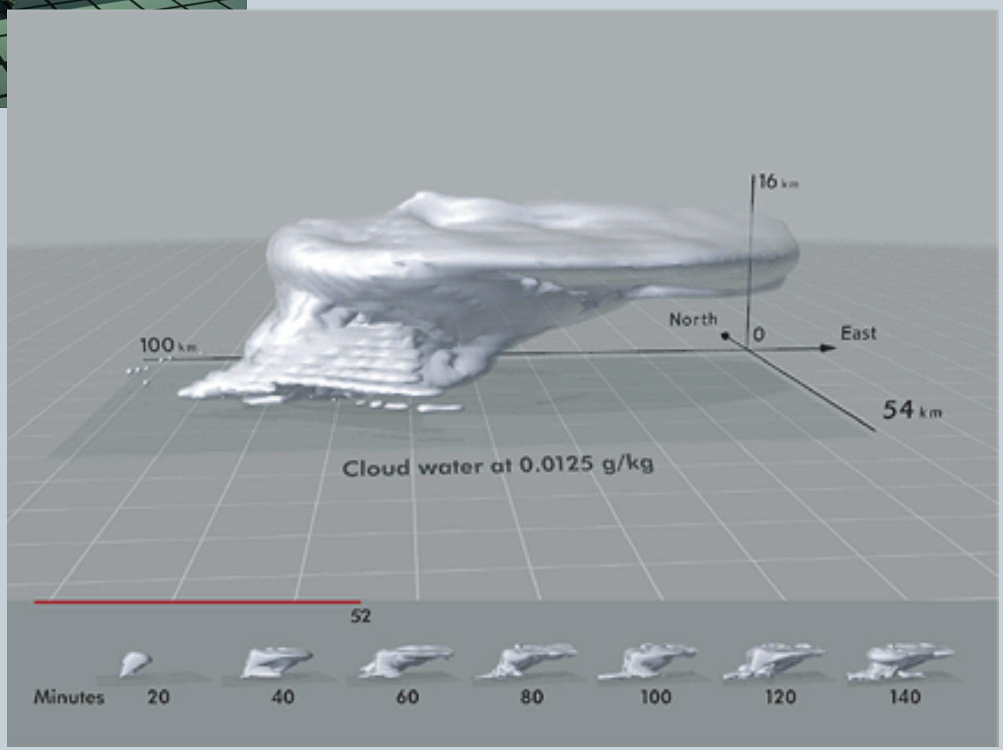
- Evaluate:
  - Use of color
  - Use of shadows
  - Use of transparency
  - Use of animation
  - Display of multiple fields clearly

Modeled Storm video



- Tufte:
  - Scale should remain
  - Advancing line rather than clock
  - Tile too prominent

Modeled Storm video

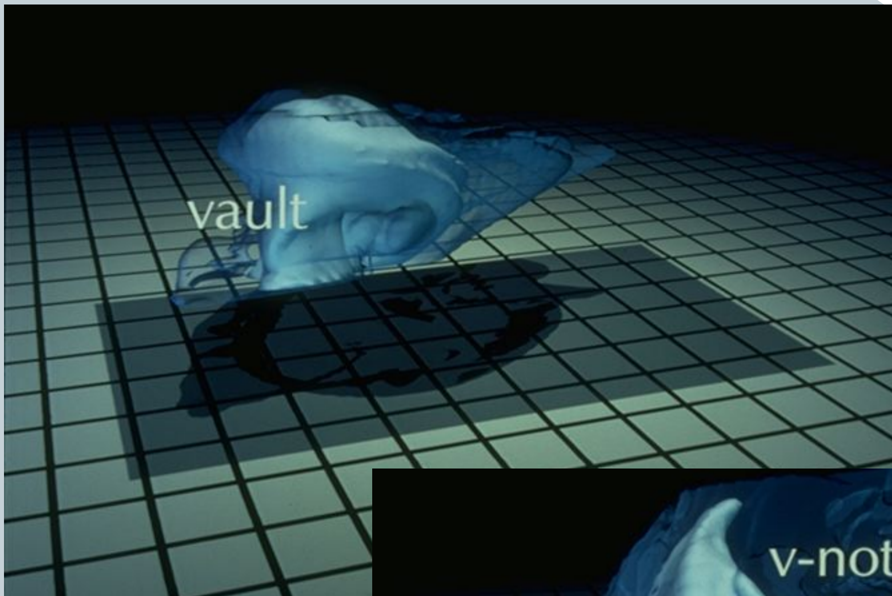


- Tufte:
  - Scale should remain
  - Advancing line rather than clock; small cloud images illustrate the temporal evolution
  - Tile too prominent

Modeled Storm video

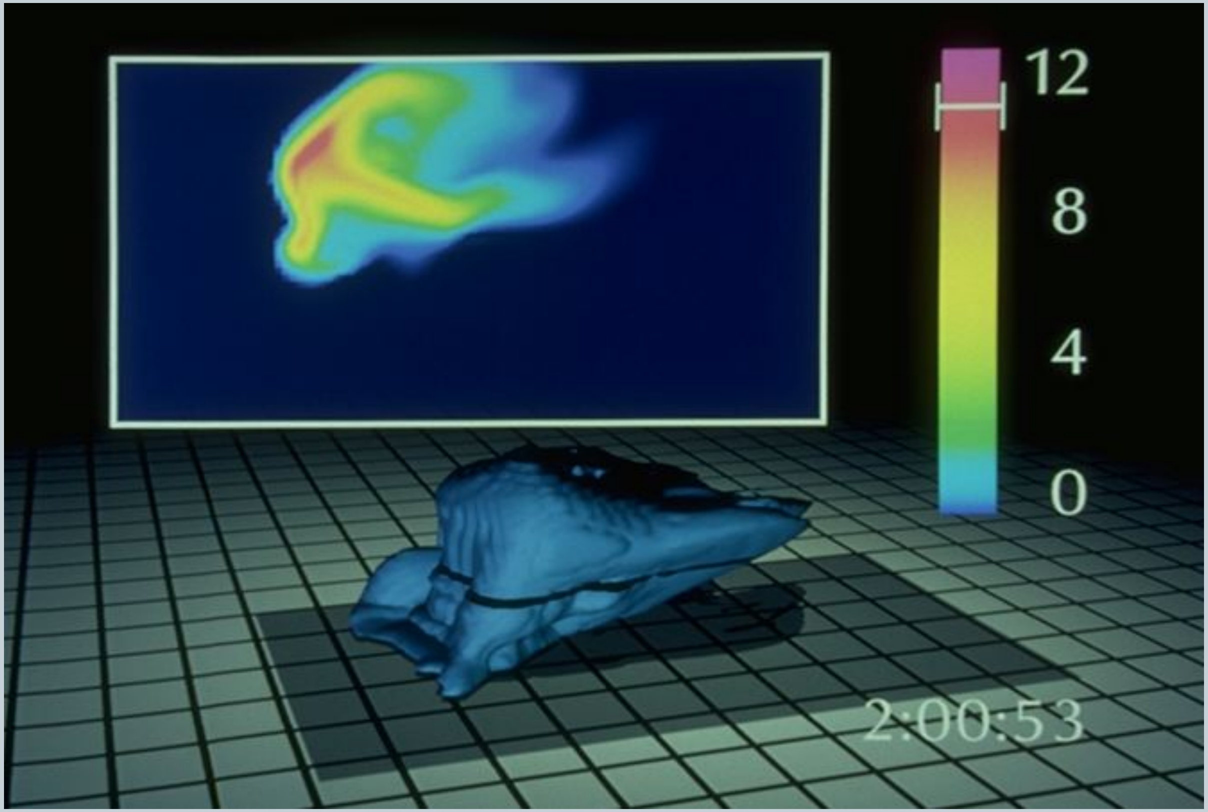
- Thoughts:
  - Shadow useful?
  - Perspective changes are helpful
  - Need explanation that moving tile = model domain

Modeled Storm video



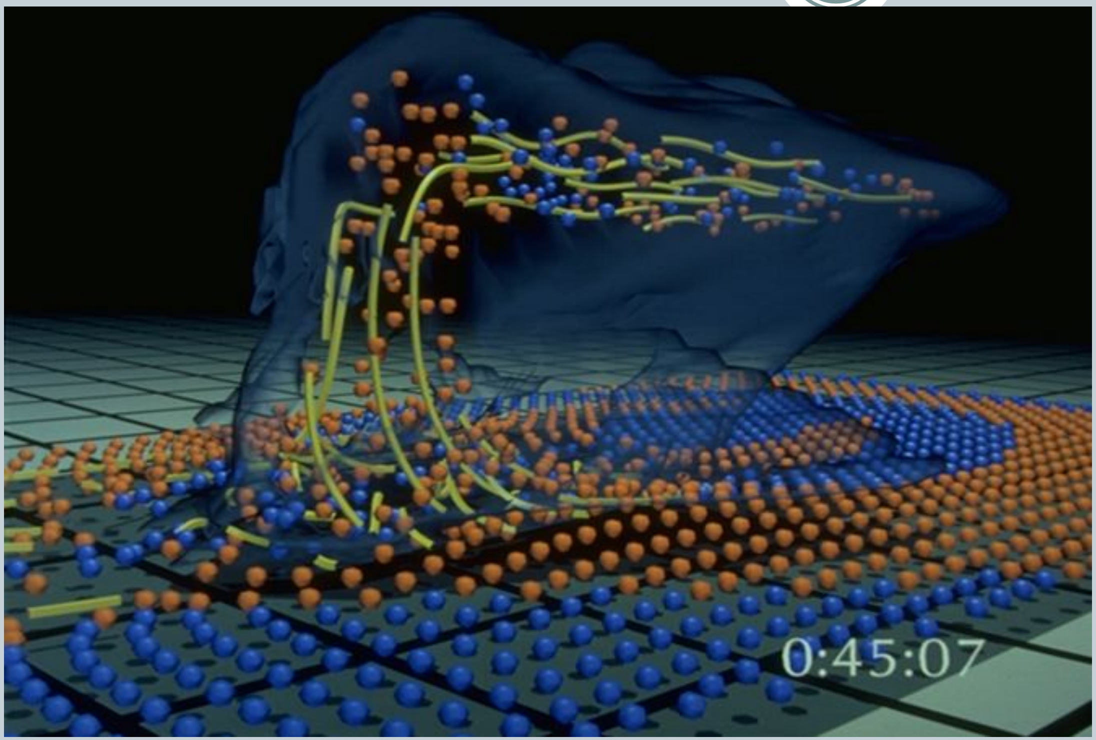
- Viewpoint:
  - Intent was to approximate the view of an actual thunderstorm
  - Educational effort, using terms from radar meteorology

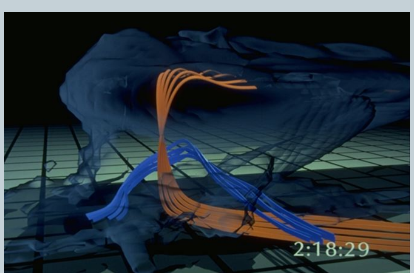
Modeled Storm video

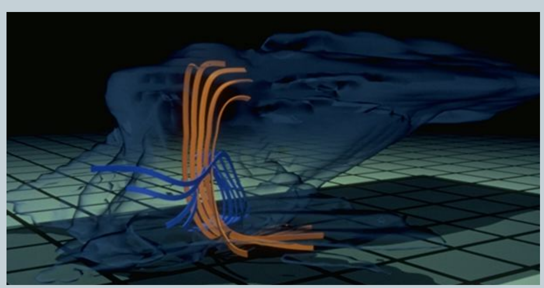


- Radar view - effective use of color (not enough otherwise?)
- Simultaneous 2d, 3d detail

Modeled Storm video

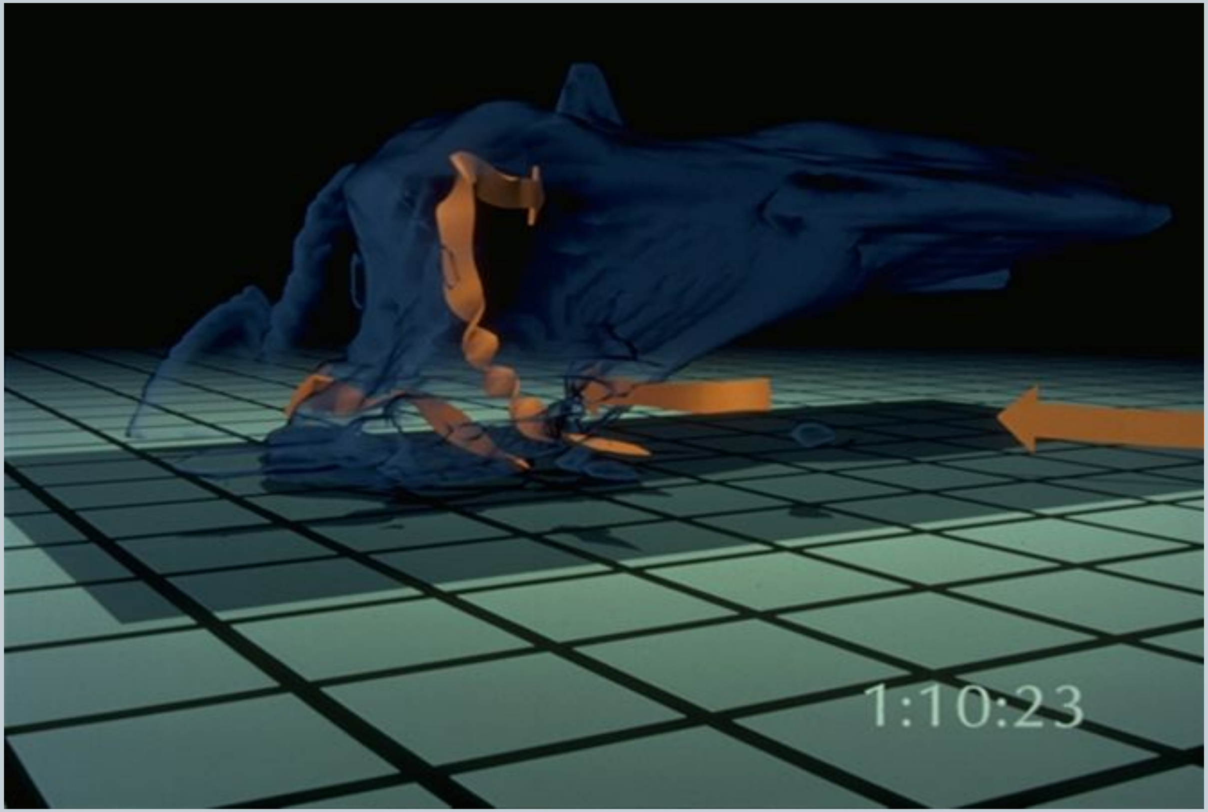






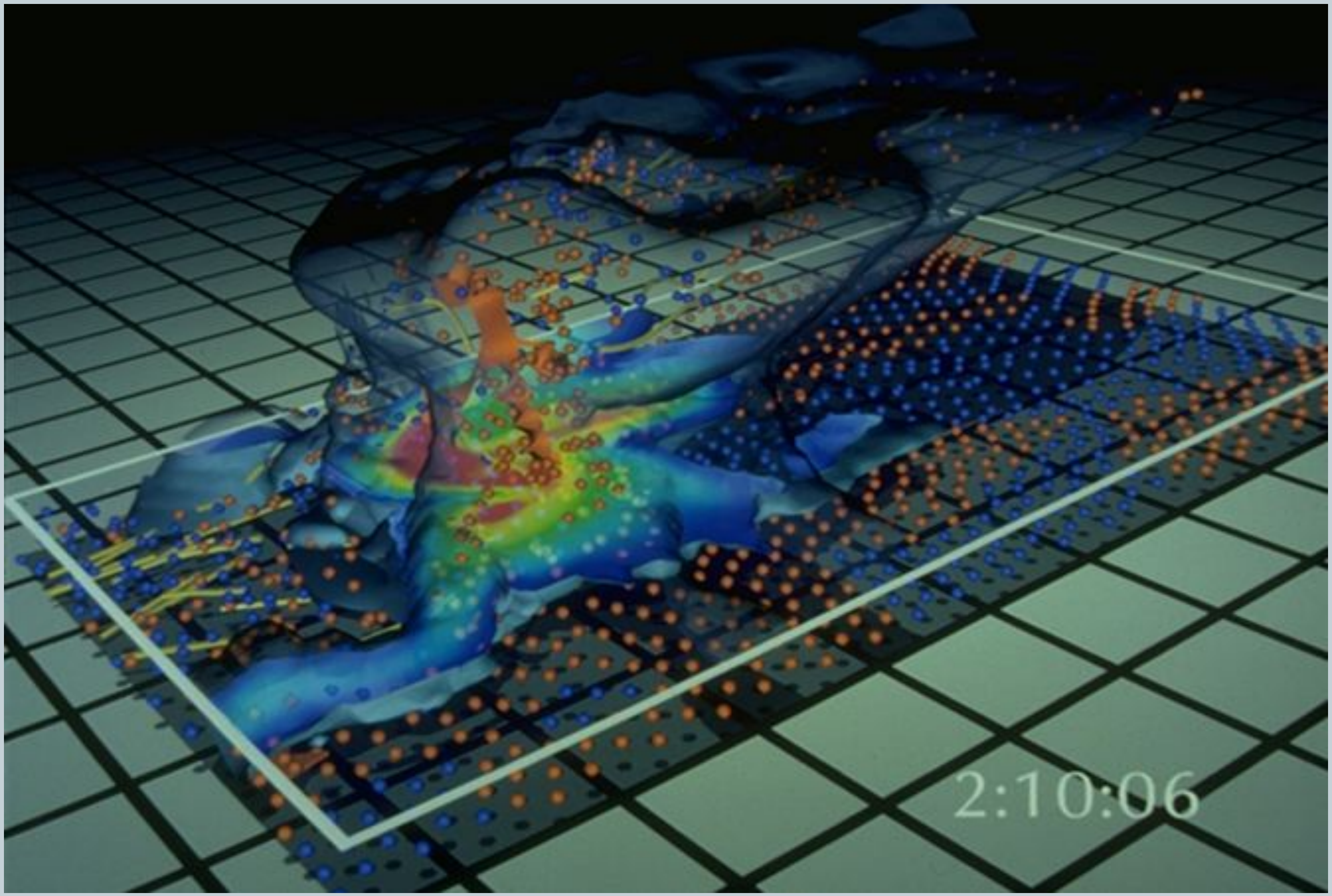
- Weightless particles and ribbons
- Particles: instantaneous up- and down-draft, position
- Ribbons: history

Modeled Storm video



- Introducing another visual idiom
- New field (vorticity) put in context of familiar (cloud) idiom

Modeled Storm video



- Cutaway (in lieu of radar slice); use of transparency