Lecture 26:

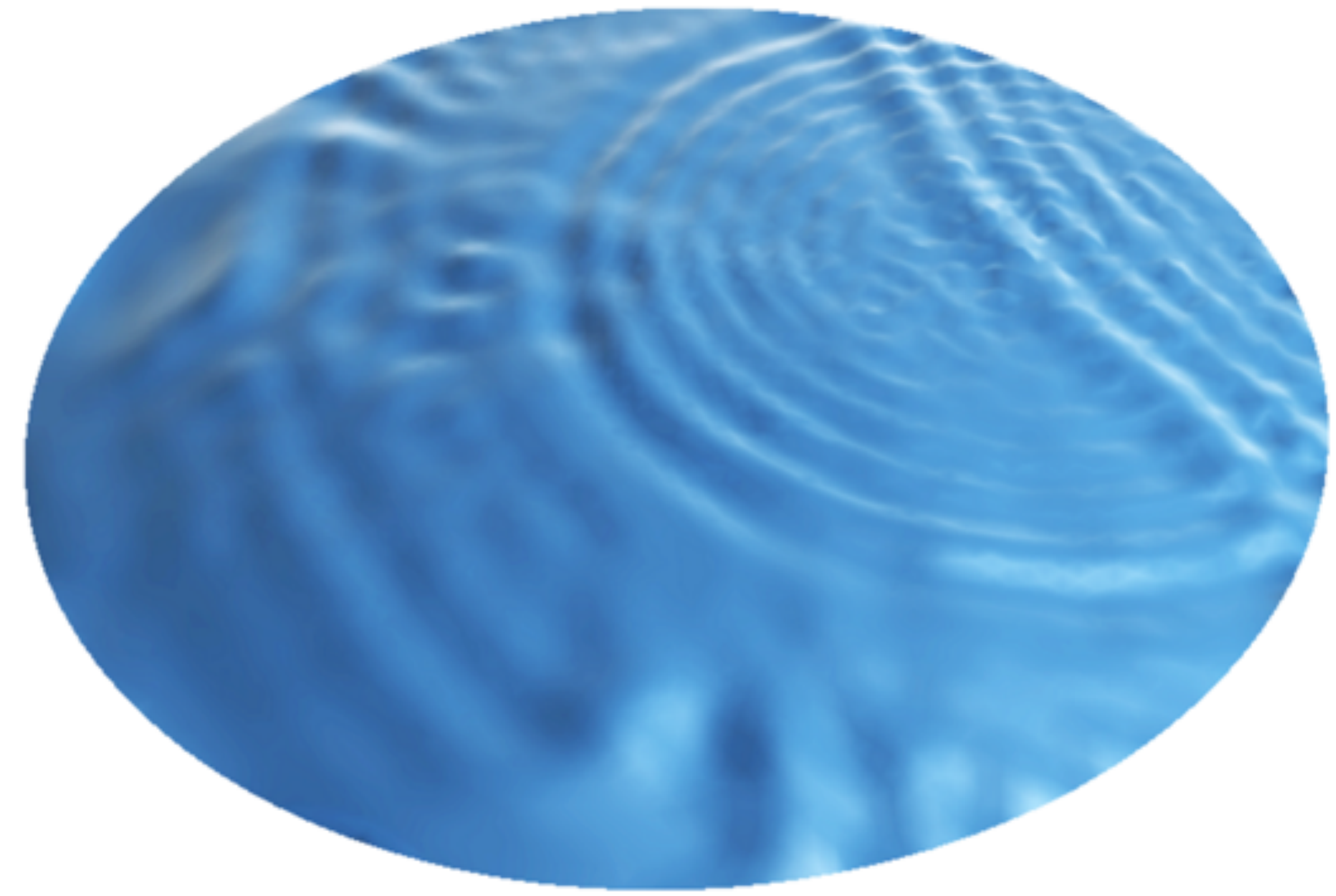
Advanced Topics: Fluid Simulation

Computer Graphics

CMU 15-462/15-662, Fall 2015

Previously: Partial Differential Equations

- Animation via differential equations
- Implicit description of motion
- Solve in space and time
- Considered three model equations:
  - Laplace (elliptic)
  - heat (parabolic)
  - wave (hyperbolic)



- Already saw some cool/interesting behavior (e.g., ripples)
- Real physical phenomena involve harder (nonlinear) PDEs
- TODAY: fluids (smoke, water, fire, ...)
  - Basis of many recent (beautiful!) visual effects

Fluids in Computer Graphics - Smoke

Zhang, Bridson, Greif, "Restoring the Missing Vortices in Advection-Projection Fluid Solvers"

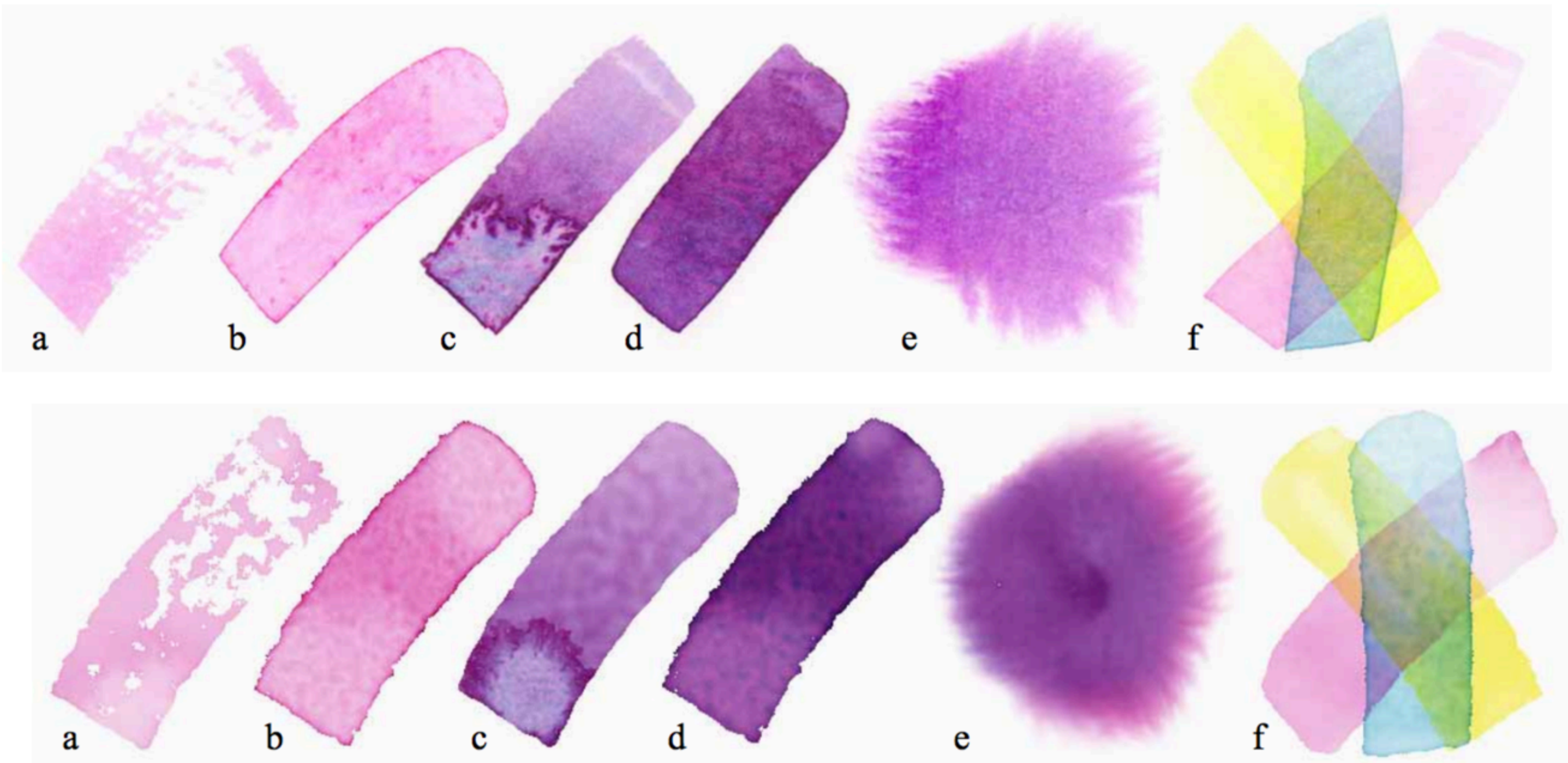
Fluids in Computer Graphics - Water

Ando, Thürey, Wojtan, "Highly Adaptive Liquid Simulations on Tetrahedral Meshes"

Fluids in Computer Graphics - Fire

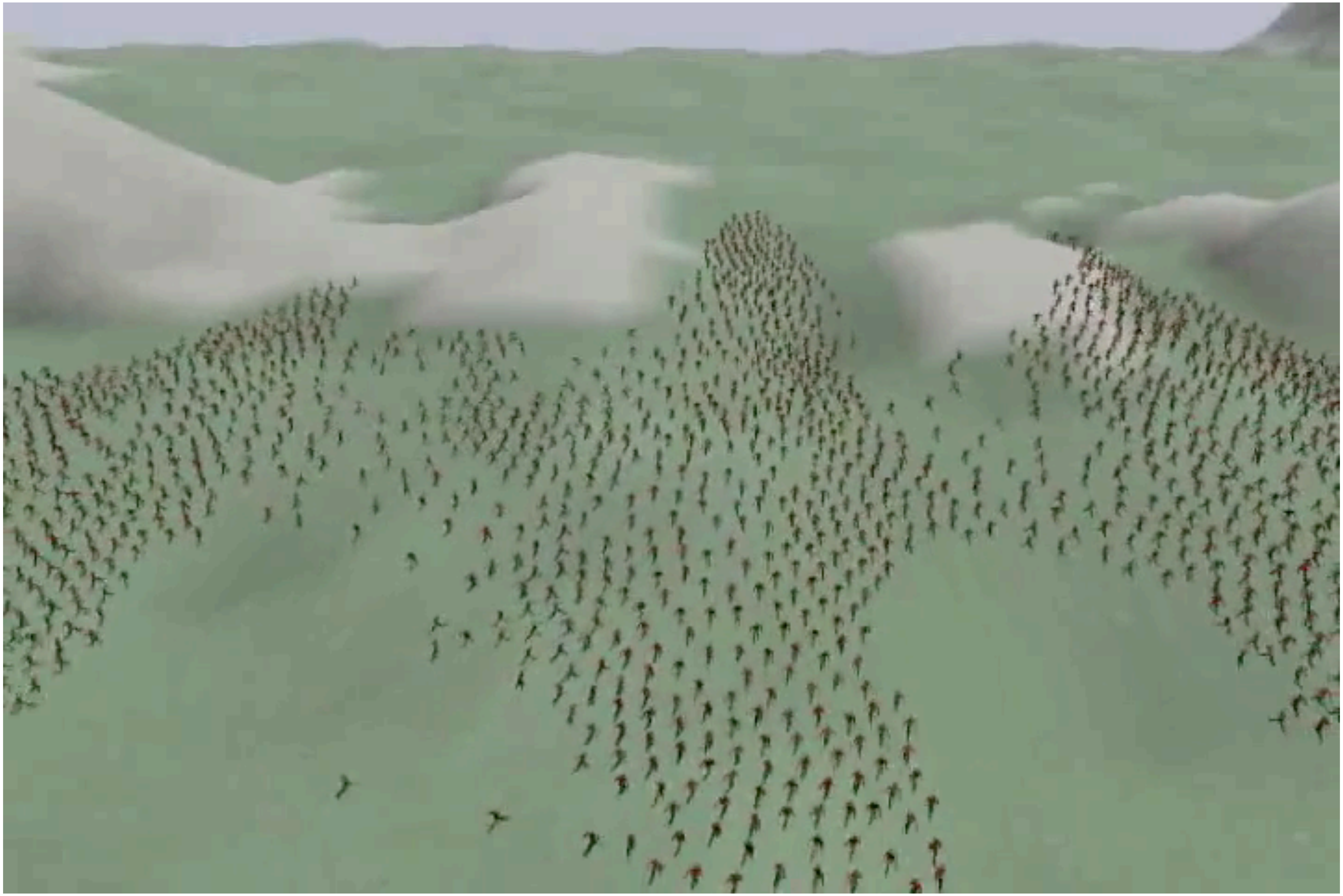
Hong, Shinar, Fedkiw, *"Wrinkled Flames and Cellular Patterns"*

Fluids in Computer Graphics - Paint



Curtis, Anderson, Seims, Fleischer, Salesin, *"Computer-Generated Watercolor"*

Fluids in Computer Graphics - Crowds



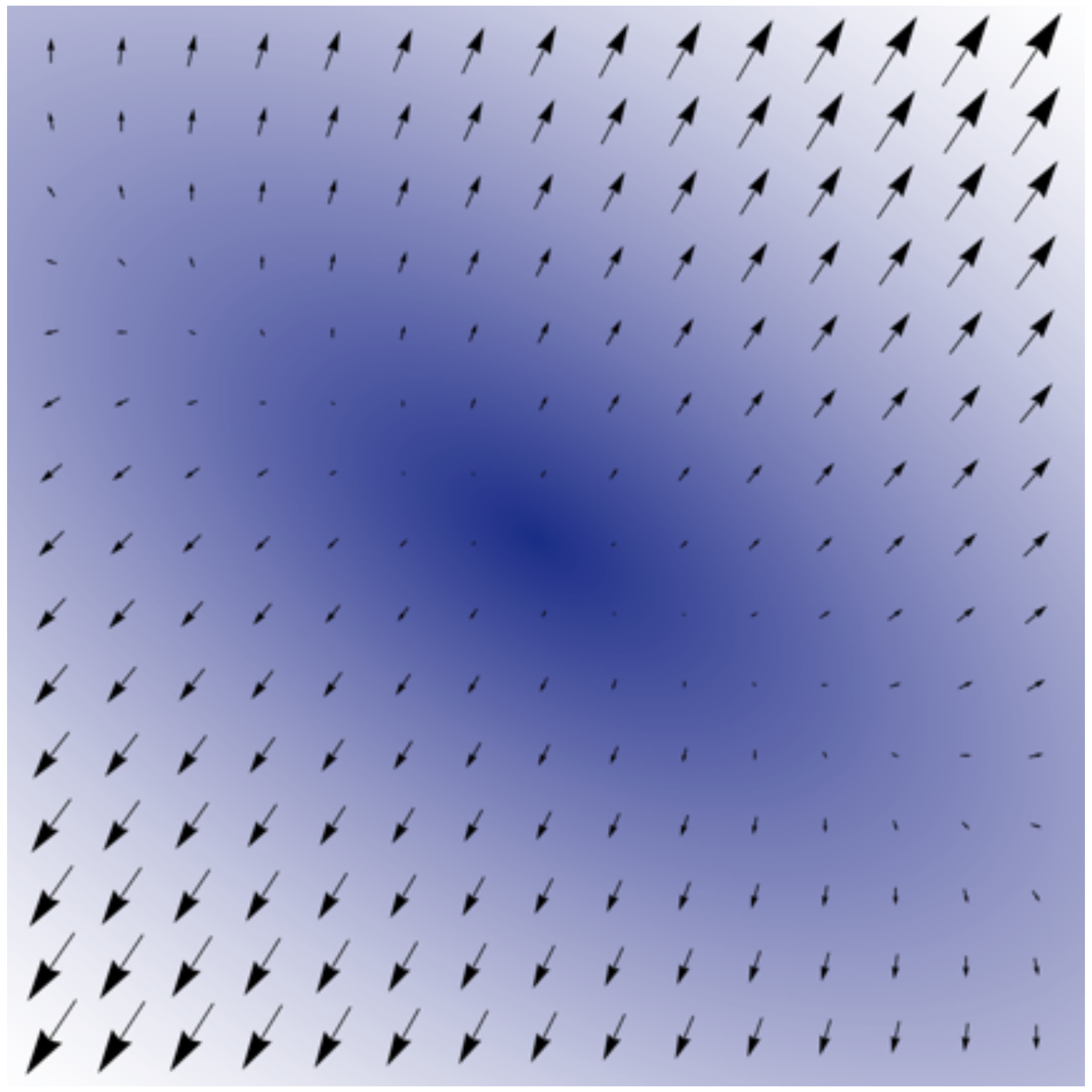
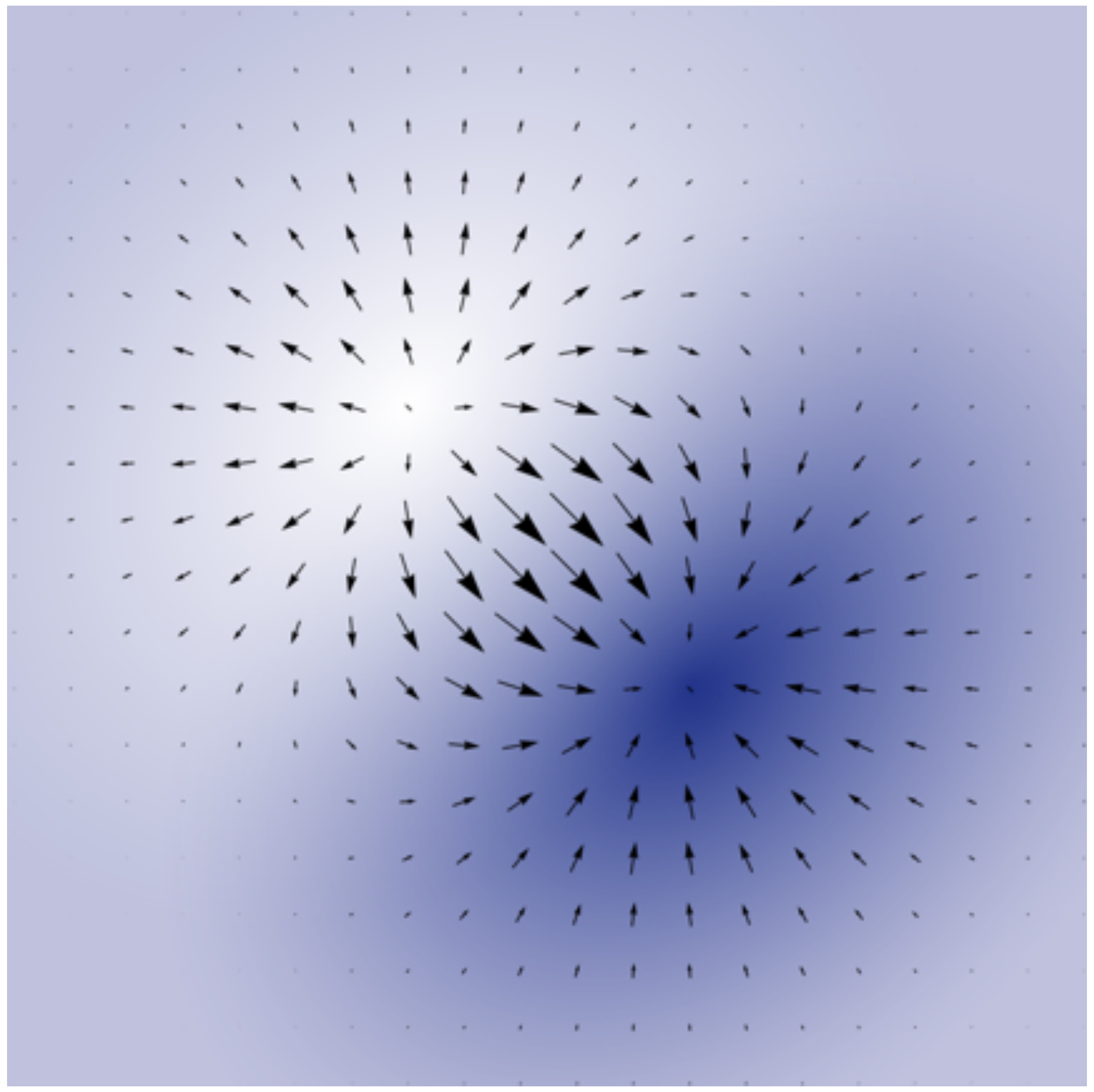
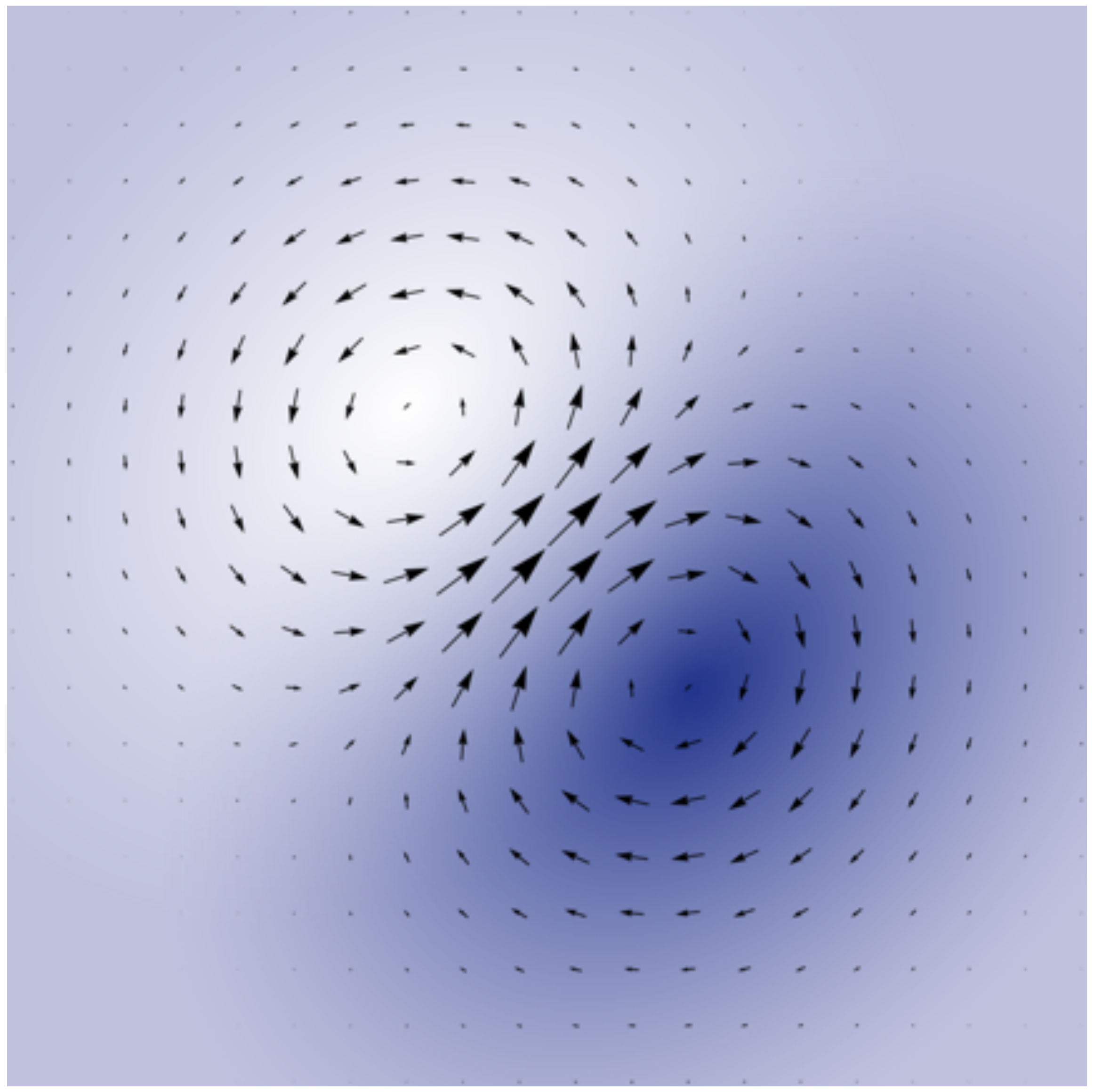
Treuille, Cooper, Popovic, *"Continuum Crowds"*

Visual Vector Calculus—Div, Grad, & Curl

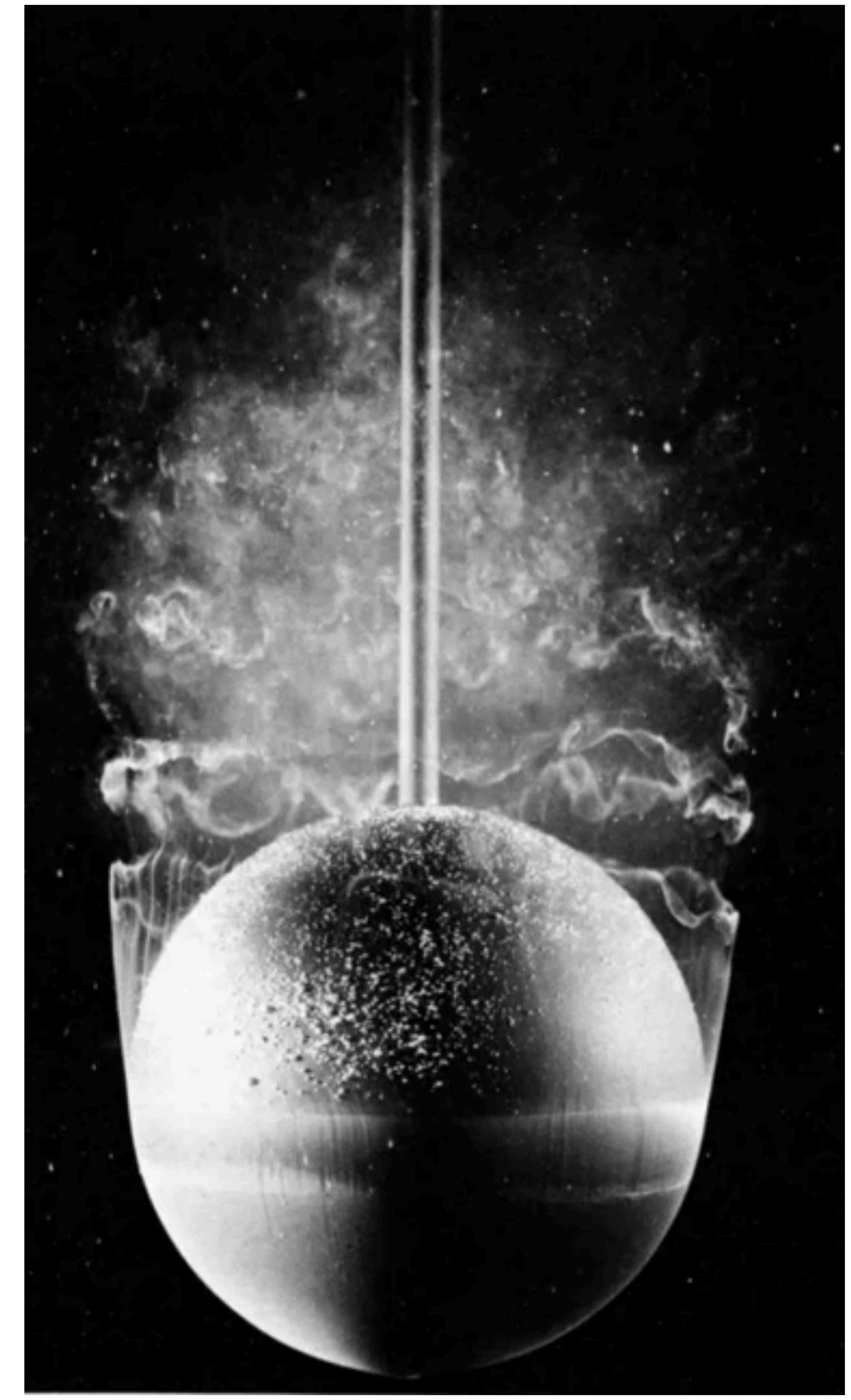
- We're going to use these a *lot* today
- *Keep these three pictures in mind!*

φ — scalar function

u — vector field

GRADIENT	DIVERGENCE	CURL
		
$\nabla \varphi$	$\nabla \cdot u$	$\nabla \times u$
("steepest direction")	(how much "spreading out?")	(how much "spinning?")

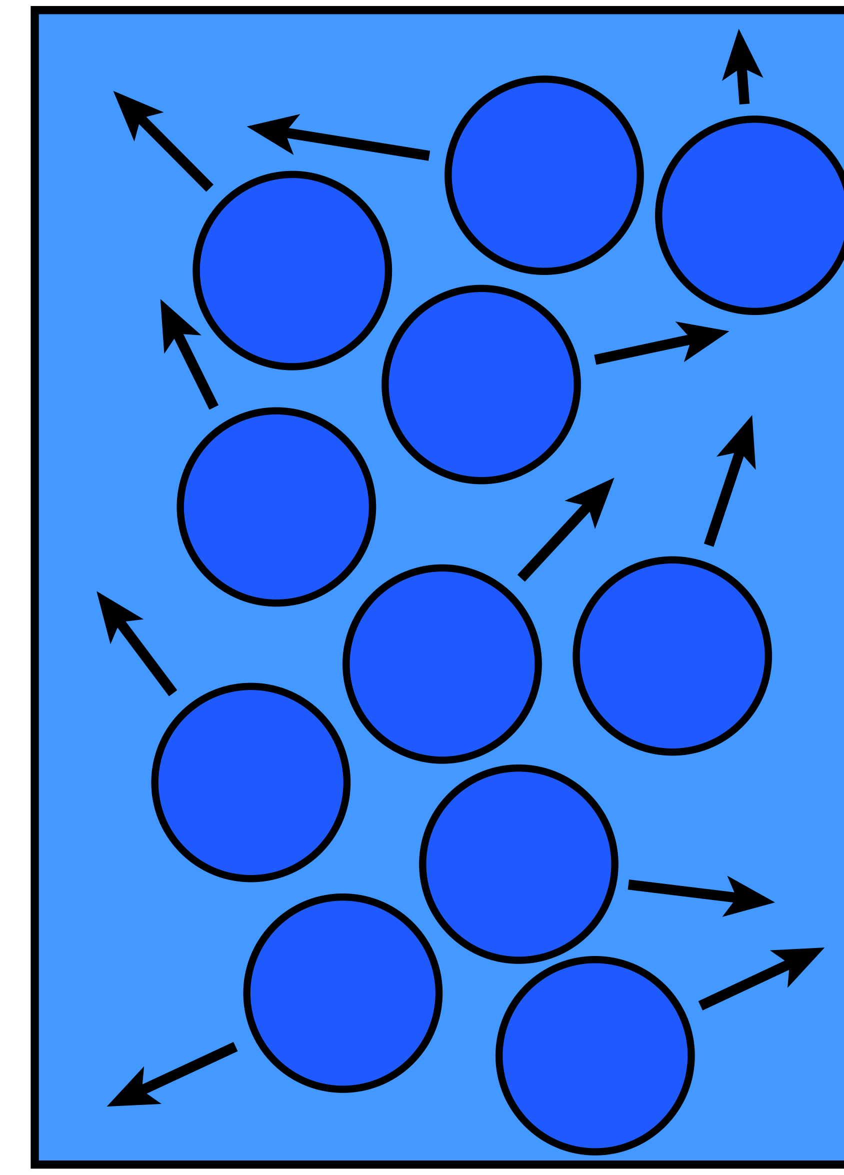
The Fluid Equation?



- How do we describe the motion of a fluid?
- IMPORTANT: There are many different ways
  - Navier-Stokes equation
  - vorticity equation
  - Euler-Arnold equation
  - molecular dynamics
  - ...
- For mainly historical reasons, Navier-Stokes (velocity-based) approach largely dominates computer graphics
- Moving forward, does *not* mean it's the right way for computation
- As with everything: *pick the right tool for the job!*

Incompressible Navier–Stokes

- Navier-Stokes “most traditional” description
- Illustrates many* important phenomena
- Expressed in terms of *fluid velocity* u(t,x)
- At each point, how fast is fluid moving?
- Nonlinear PDE, 1st-order in time, 2nd-order in space:



u

change in velocity over time

advection (“particles move forward”)

pressure (“particles should not collide”)

viscosity / diffusion (one of our linear model equations!)

external forces (e.g., gravity)

$$\dot{u} = -u \cdot \nabla u - \frac{1}{\rho}\nabla p + \nu \nabla^2 u + g$$

incompressibility (assume fluid doesn’t gain/lose volume)

$$\nabla \cdot u = 0$$

*but not all!

Where does this equation come from?

For the moment, let's ignore viscosity, incompressibility, and external forces...

Review: Chain Rule

- **Consider a composite function f(g(x))**
- **Once upon a time, you may have memorized the formula**

$$\frac{df}{dx} = \frac{df}{dg}\frac{dg}{dx}$$

"chain rule"

Q: Where does this “rule” come from?

A: It comes from your high school calculus teacher.

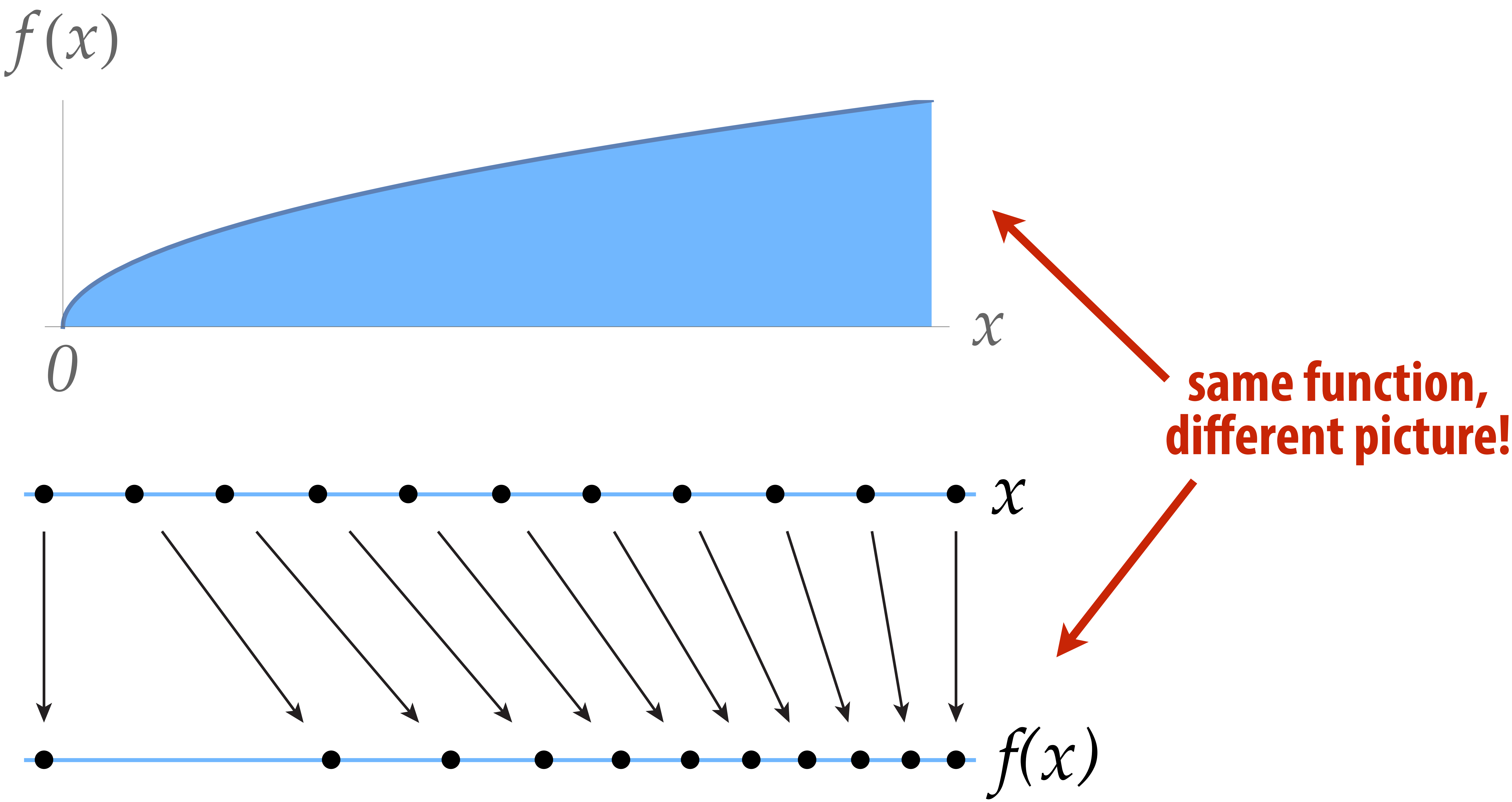
Rules are something you're told by an authority to blindly obey.

What we'd rather understand is the undeniably self-evident "chain *fact*."

So how do we see it?

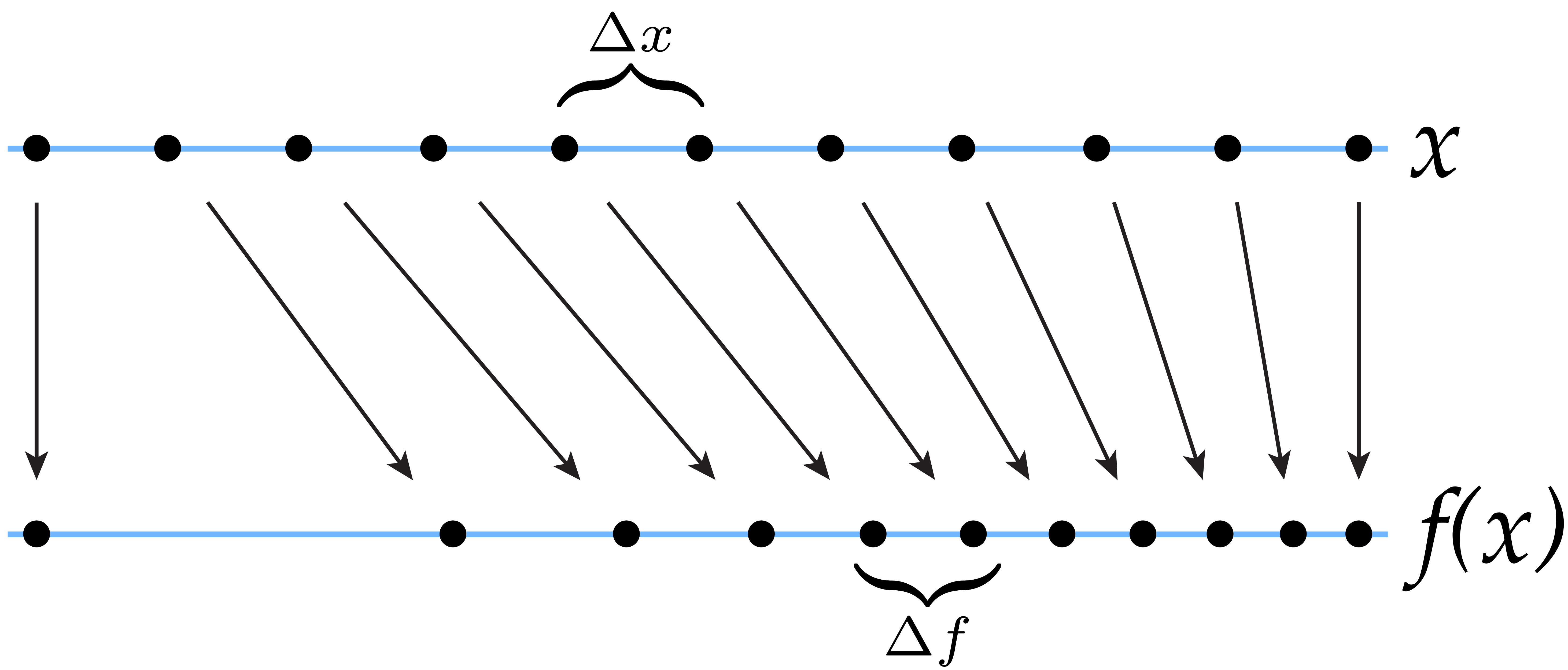
Visual Calculus: The Chain Fact

- **First, stop thinking about a function in terms of its graph; instead, think about it as a map from one space to another:**



Visual Calculus: The Chain Fact

- **The derivative is no longer the *slope*; it is now the amount by which the function locally gets squashed/stretched:**

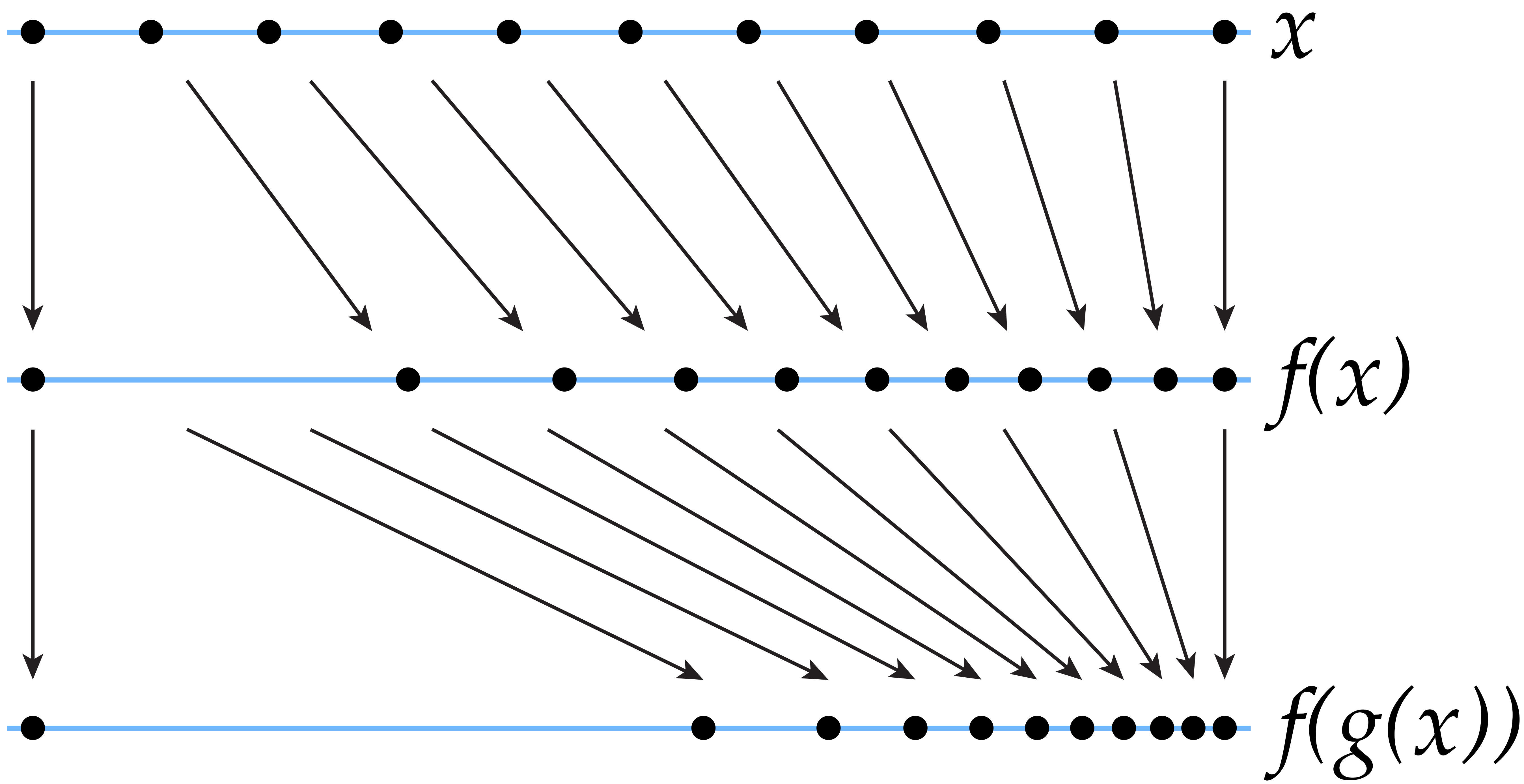


"squash factor"

$$\frac{df}{dx} \approx \frac{\Delta f}{\Delta x}$$

Visual Calculus: The Chain Fact

- So, the chain rule just says: "to get the total amount of squashing, just multiply the squash factors."



$$\frac{df}{dx} = \frac{df}{dg}\frac{dg}{dx}$$

"repeated squashing rule"

The more you do graphics, the more you discover that you need to take "derivatives of weird functions."

Keeping this "stretching" picture in mind can be enormously helpful.

Review: The Total Derivative

- **Now suppose we have a function f that depends on some parameter t both directly and indirectly:**

$$f(t, x(t), y(t), \ldots)$$

- **How does f change as we vary t? Another "rule"...**

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial x}\frac{dx}{dt} + \frac{\partial f}{\partial y}\frac{dy}{dt} + \cdots$$

- **(Food for thought: what does it mean intuitively/geometrically?)**

The Material Derivative

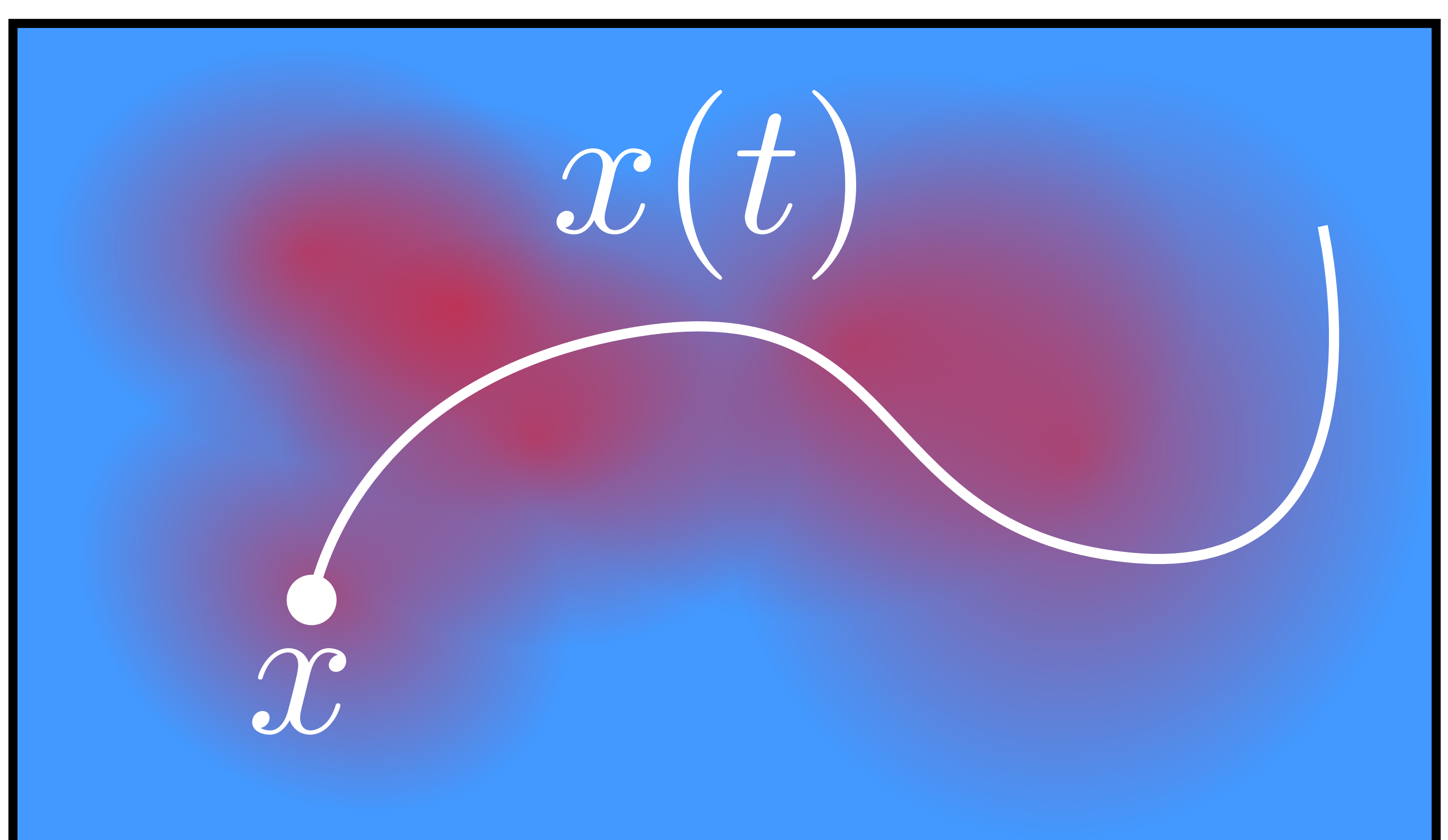
- **For fluids, the derivative we really need to understand is the so-called *material derivative* or *Lagrangian derivative***
- **As a running example, think about the temperature φ in a swimming pool, at a point x, at a time t:**

time position

temperature

$$\varphi(t, x(t))$$

- **Notice that the point x is itself moving over time.**



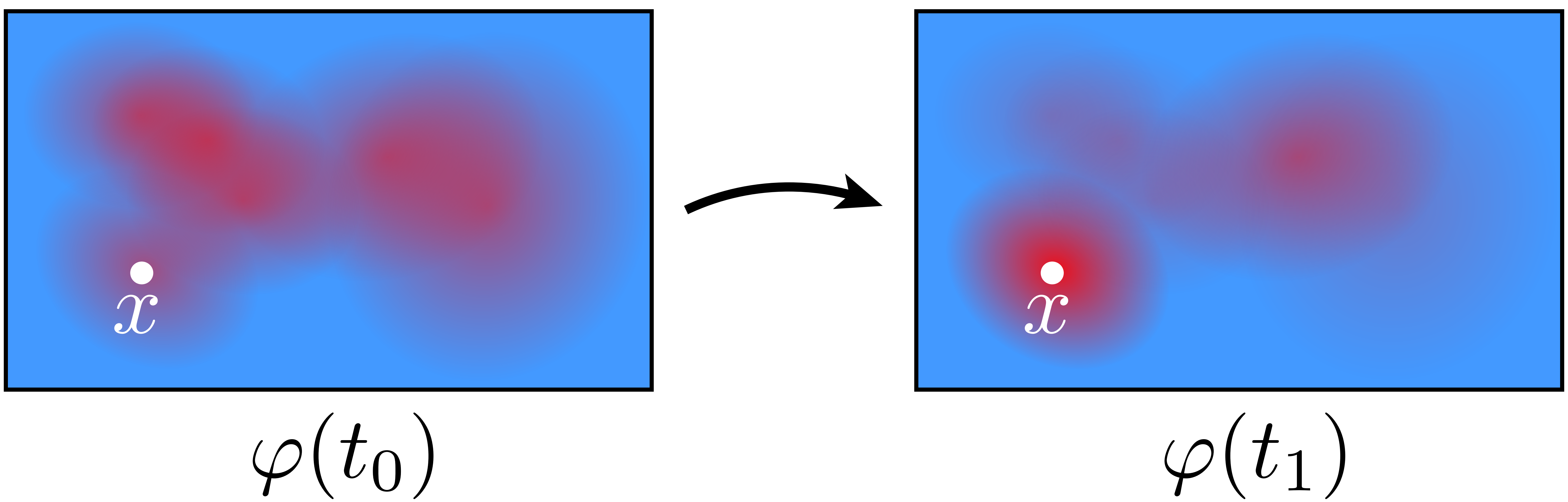
φ

The Material Derivative - Time

- **Suppose our point x is not moving. The observed change in temperature is then just**

"how much is the pool heating up/ cooling down *at the current point x?*"

$$\frac{d\varphi}{dt} = \frac{\partial \varphi}{\partial t}$$

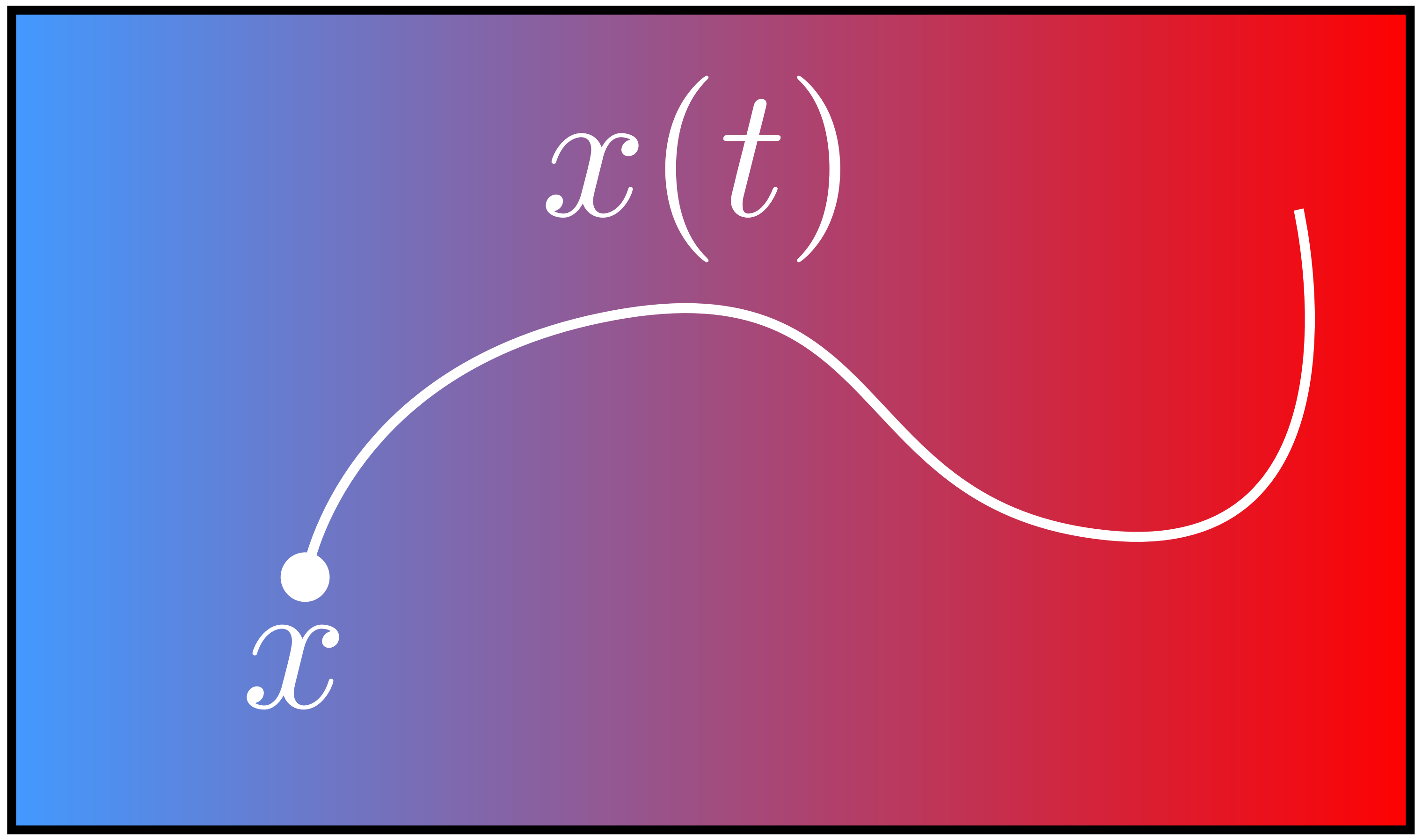


The Material Derivative - Space

- **Suppose we swim from one end to the other, without changing the temperature in the pool. Then the observed change in temperature is**

"how much does the temperature change if we swim in a given direction?"

$$\frac{d\phi}{dt} = D_{\dot{x}}\varphi$$



Spatial derivative alone is called the "directional derivative" (more generally: *covariant derivative* or *connection*.) Can be written as

$$\nabla\varphi \cdot \dot{x}$$

The Material Derivative

- So what's the overall change in temperature?
- Well, we could apply the "total derivative rule":

"pool is changing temperature"

$$\frac{d\varphi}{dt} = \frac{\partial \varphi}{\partial t} + \underbrace{\frac{\partial \varphi}{\partial x}}_{\nabla \varphi} \underbrace{\frac{dx}{dt}}_{\dot{x}} = \frac{\partial \varphi}{\partial t} + \nabla \varphi \cdot \dot{x}$$

"swimmer is moving to hotter / cooler part of the pool"

- And... hey! That's exactly what we got by thinking about the change in time and space independently.

The Material Derivative

- **In a fluid, the direction of motion is the fluid velocity u:**

$$\dot{x} = u$$

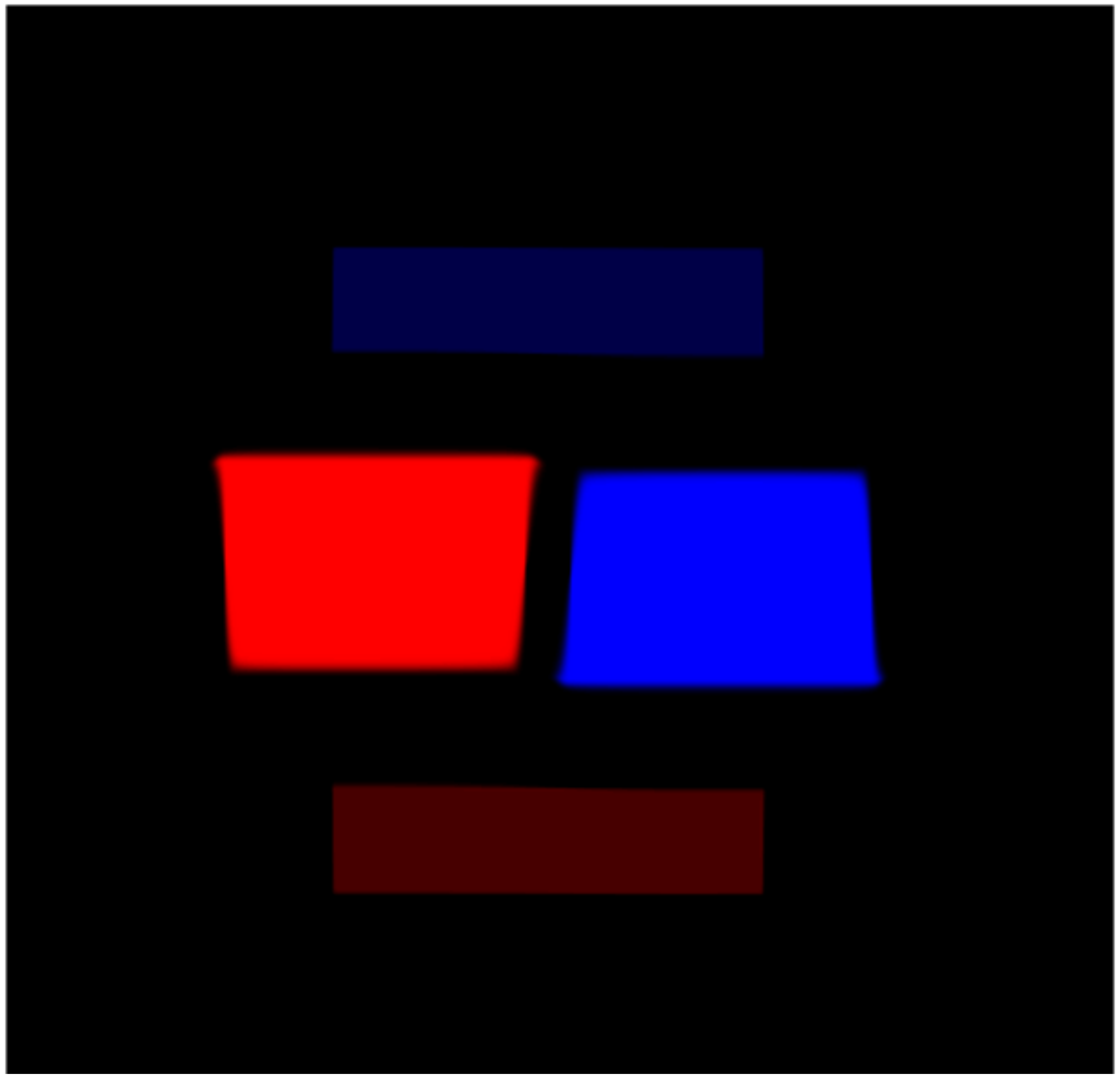
- **The *material derivative* is the change in a quantity due to fluid motion, i.e., due to the phenomenon of "advection":**

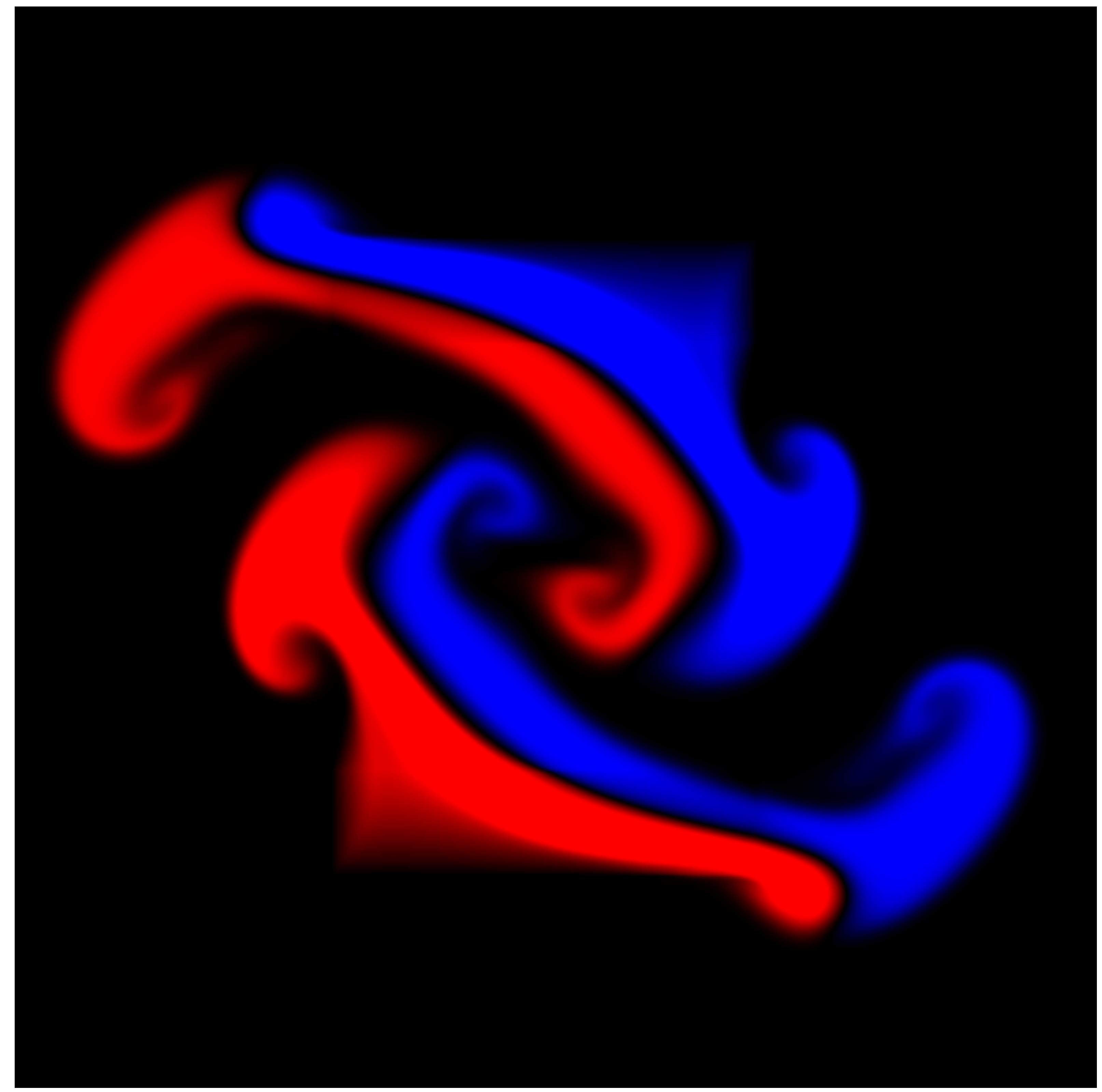
$$\frac{D\varphi}{Dt} := \frac{\partial \varphi}{\partial t} + u \cdot \nabla \varphi$$

"material derivative"

Example of Advection

Color gets "pushed along" by velocity:

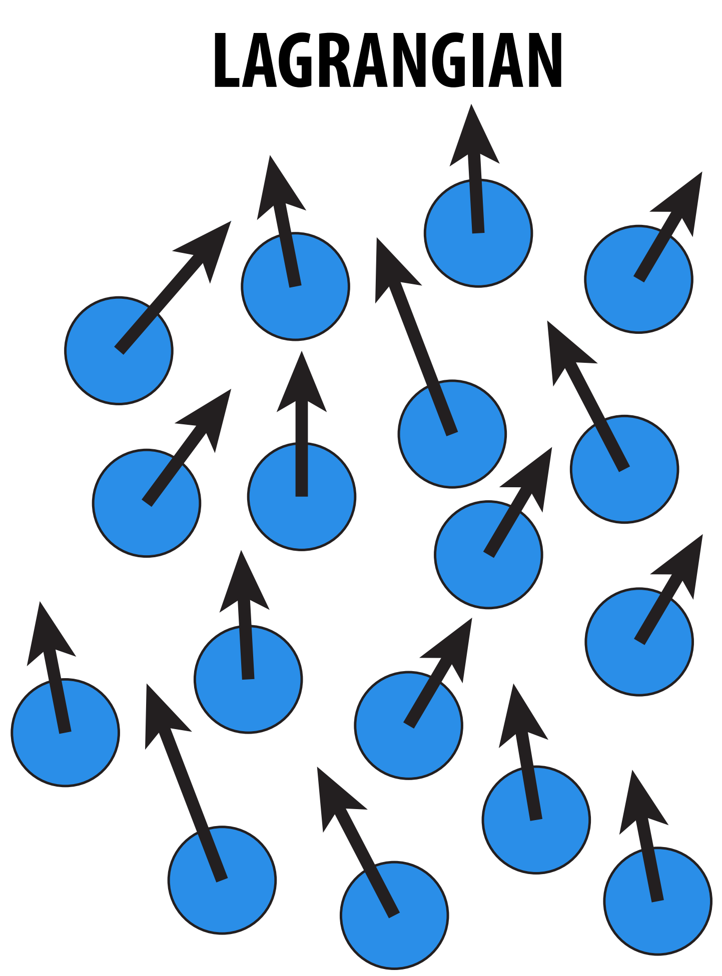




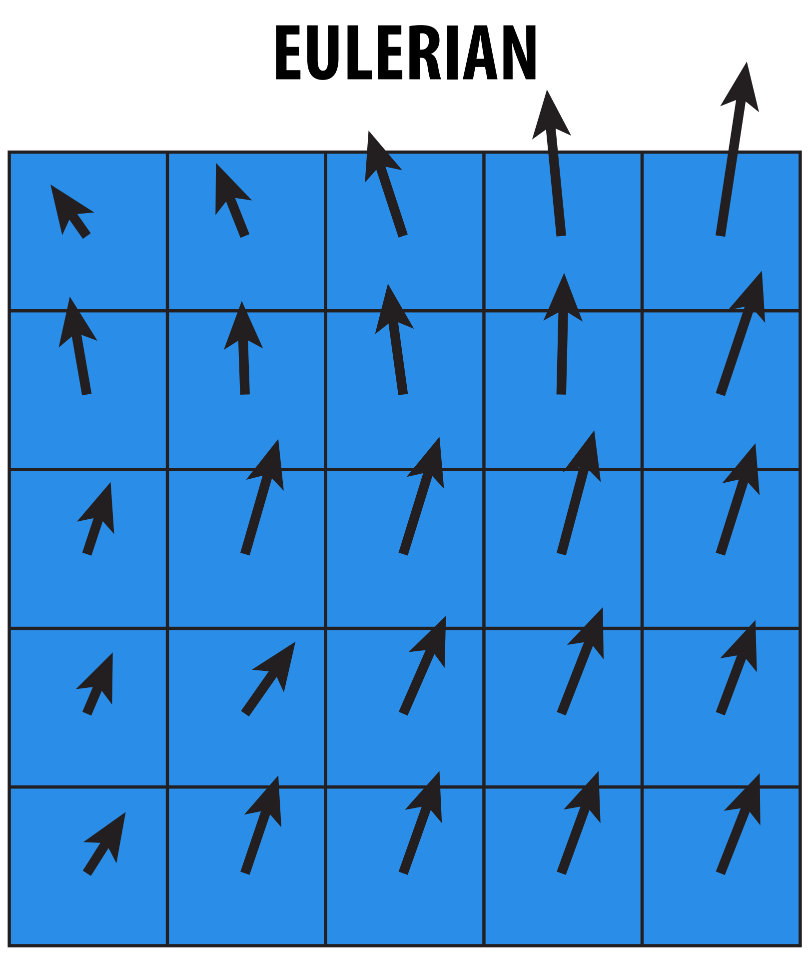
https://www.ibiblio.org/e-notes/webgl/gpu/fluid.htm

Computing Advection

- Numerically, how do we advect a given quantity φ (color, temperature, ...) in a given vector field u?
- Many options—recall discussion of *Eulerian* vs. *Lagrangian*
  - Eulerian (upwinding)
  - Lagrangian (PIC/FLIP)
  - semi-Lagrangian ("stable fluids")
- Each has pros & cons...



track position & velocity of moving particles



track velocity (or *flux*) at fixed grid locations

Upwinding (1D)

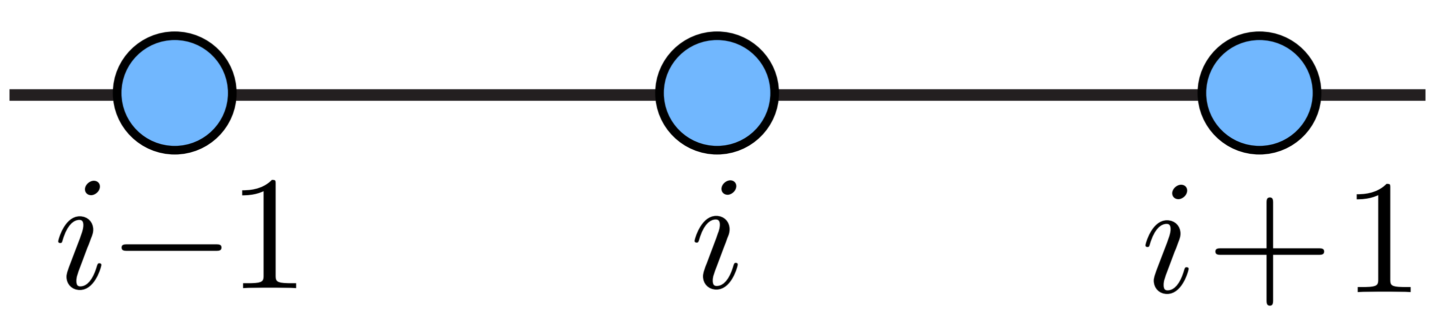
1D advection

$$\dot{\varphi} = -u\frac{\partial \varphi}{\partial x}$$

- Use Eulerian discretization (grid)
- Consider 1D advection equation
- Approximate time derivative w/ finite difference:

$$\dot{\varphi} \approx \frac{\varphi_i^{t+1} - \varphi_i^t}{\tau}$$

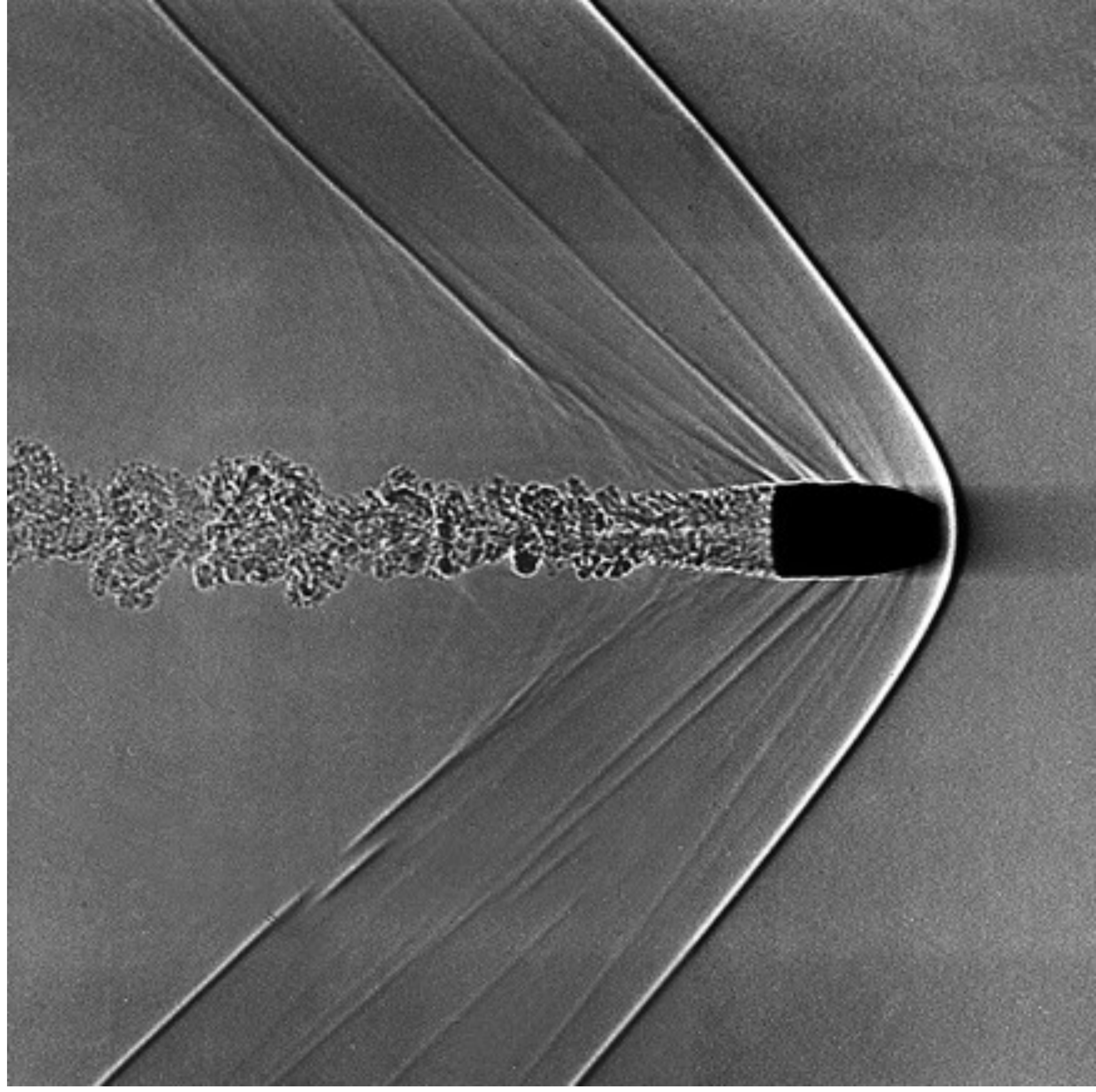
- Same deal w/ space derivative… but which direction?
- Pick based on sign of velocity ("upwind" direction):



$$\begin{array}{ll} u_i(\varphi_i^t - \varphi_{i-1}^t)/h, & u_i > 0 \\ u_i(\varphi_{i+1}^t - \varphi_i^t)/h, & u_i < 0 \end{array}$$

- Similar story in 2D, 3D (though many different options…)

Upwinding—Pros & Cons



- **Similar tradeoffs to many Eulerian methods**
- **PROS:**
  - **easy to compute (local stencil)**
  - **can generalize to very accurate schemes**
  - **helps capture strongly directional behavior like shockwaves**
- **CONS:**
  - **numerical diffusion (“blurring”) due to grid resolution**
  - **time step limited by grid size (*“CFL condition”*)**

Hence, computation gets (even) more expensive as resolution/velocity increases; not ideal for real-time (games, etc.)

time step

dimension

velocity

$$\left|\frac{u\tau^d}{h}\right| \leq 1$$

grid spacing

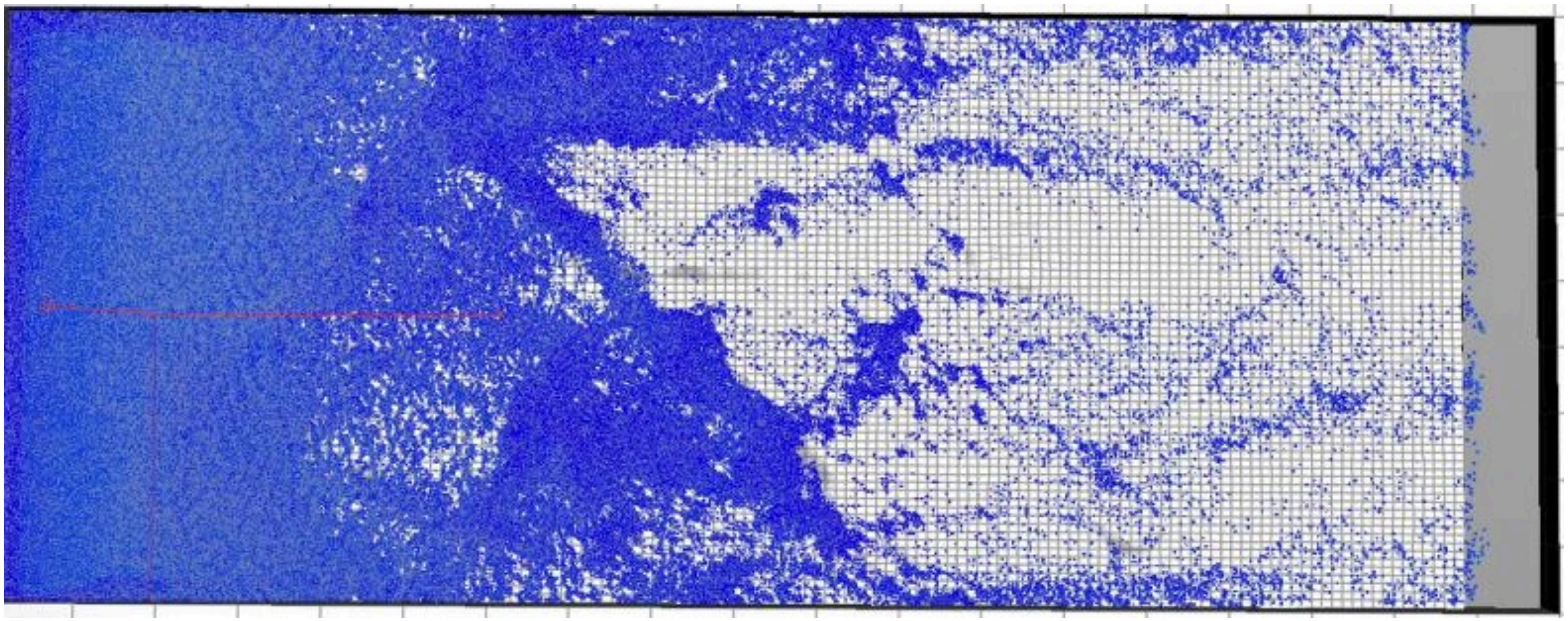
Numerical Diffusion

Unwanted "blurring" issue in all numerical advection schemes (sampling error!)

Crane, Llamas, Tariq, *"Real Time Simulation and Rendering of 3D Fluids"*

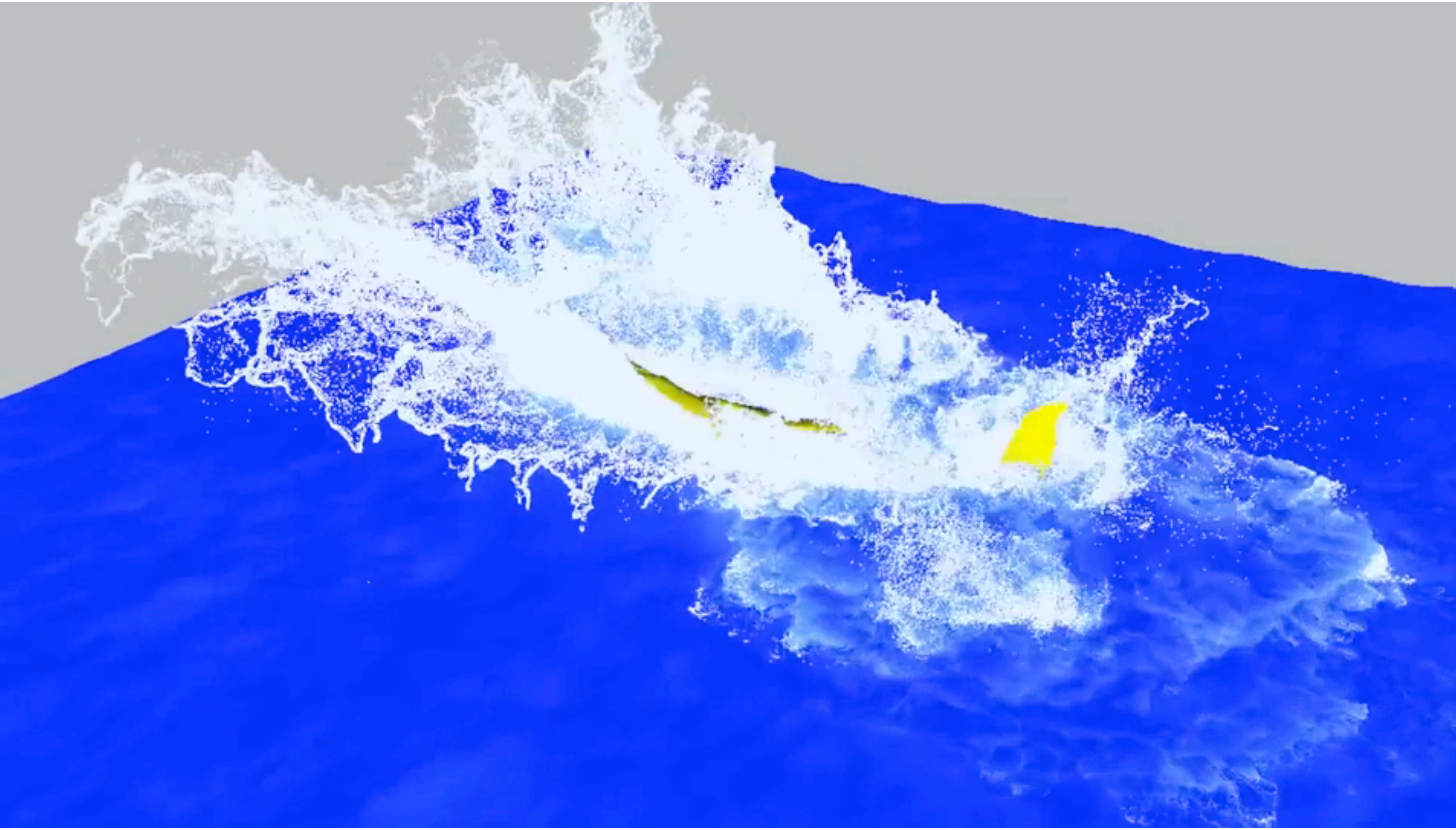
Lagrangian Advection

- Instead of grid, use particles to track velocity
- Still use background grid for other aspects of simulation



- E.g., PIC (*"particle-in-cell"*) or FLIP (*"fluid-implicit-particle"*)
- Far less numerical diffusion, but... (drawbacks?)
- Need *many* particles; sampling is hard; incoherent memory access

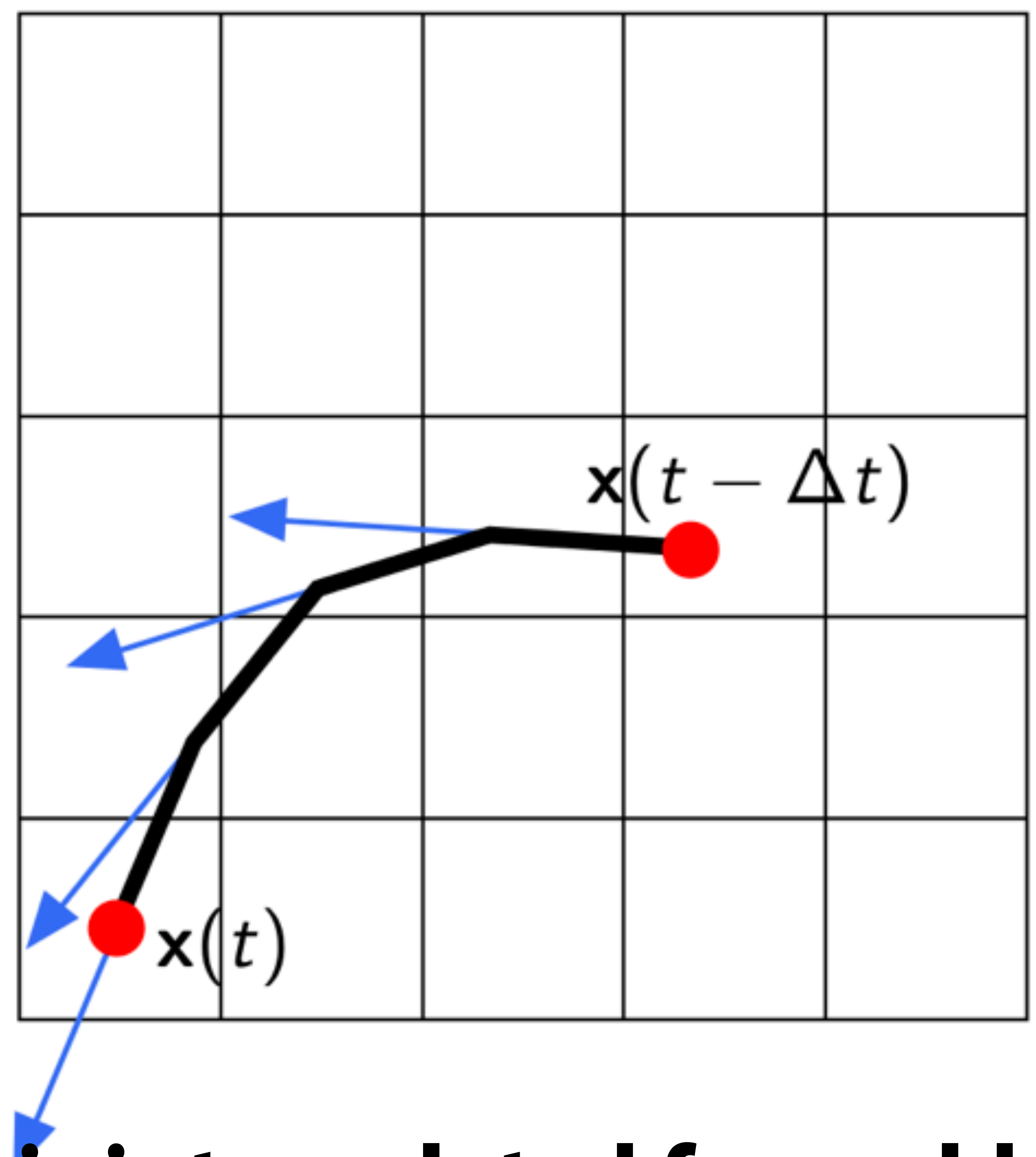
Particle Advection—Example



(produced using *"Naiad,"* now *"Bïfrost"*)

Semi-Lagrangian Advection

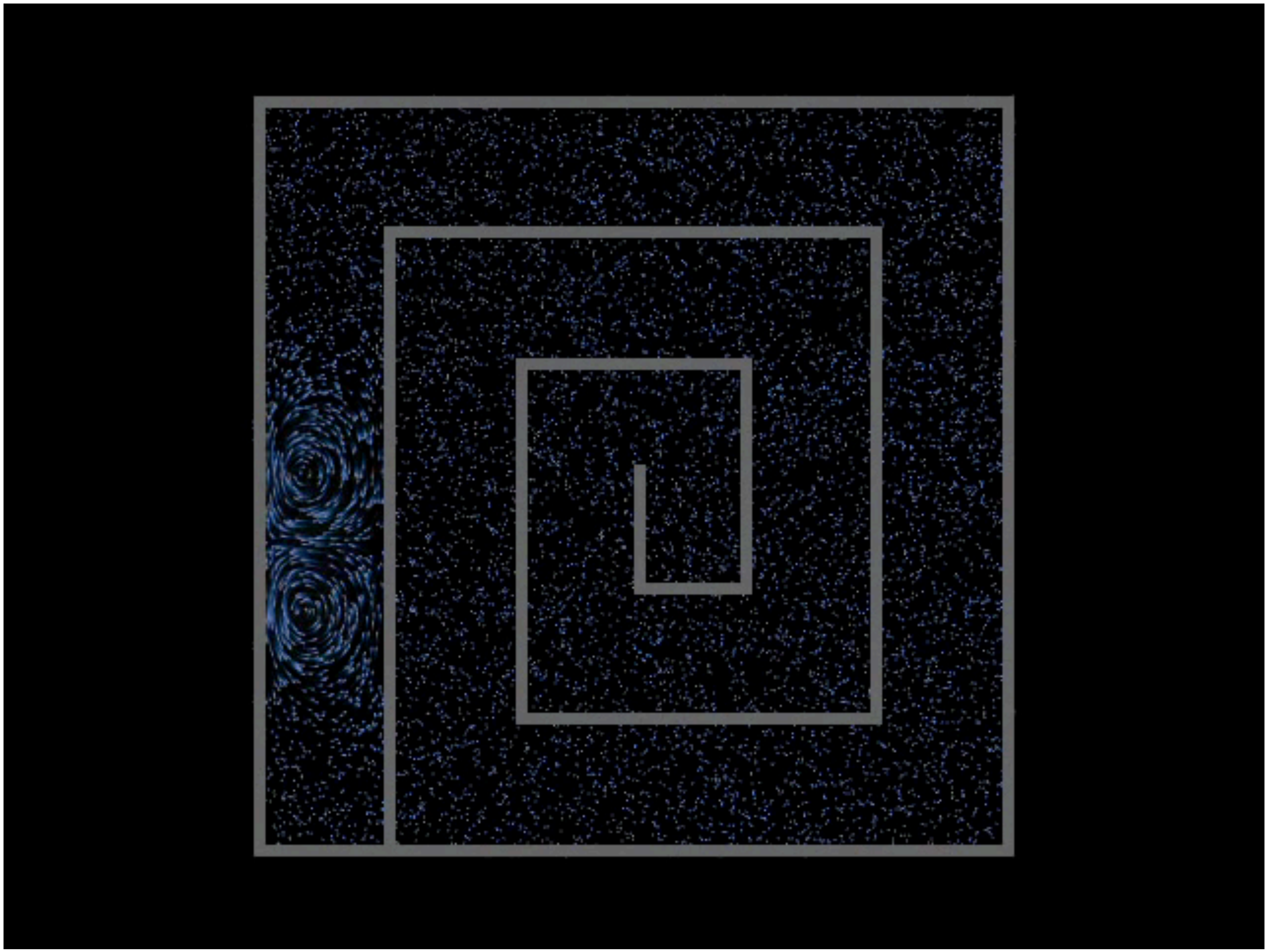
- Technique from atmospheric science (weather prediction!)
- Popular in graphics because *unconditionally stable*
- Velocities on grid; to advect, trace backwards and "copy":



- Since new velocity is interpolated from old ones, can never increase. However, can still be *inaccurate, damped*...

Numerical Energy Dissipation

- **Another common problem: loss of energy/momentum**



Mullen, Crane, Pavlov, Tong, Desbrun, *"Energy-Preserving Integrators for Fluid Animation"*

Advection—Summary

- Basic idea: push quantity forward by velocity
- Many different ways to compute:
  - Eulerian (grid)
  - Lagrangian (particles)
  - semi-Lagrangian (trace characteristics)
- Common challenges faces by *all* schemes:
  - stability (*"does it blow up for large time steps?"*)
  - numerical diffusion (*"does it get blurred out?"*)
  - numerical dissipation (*"does it lose energy?"*)
  - convergence/consistency (*"do we get the correct answer?"*)

Ok, we've now seen the first couple terms in Navier-Stokes (advection).

$$\dot{u} = -u \cdot \nabla u - \frac{1}{\rho}\nabla p + \nu \nabla^2 u + g$$

...Let's keep going.

Incompressibility

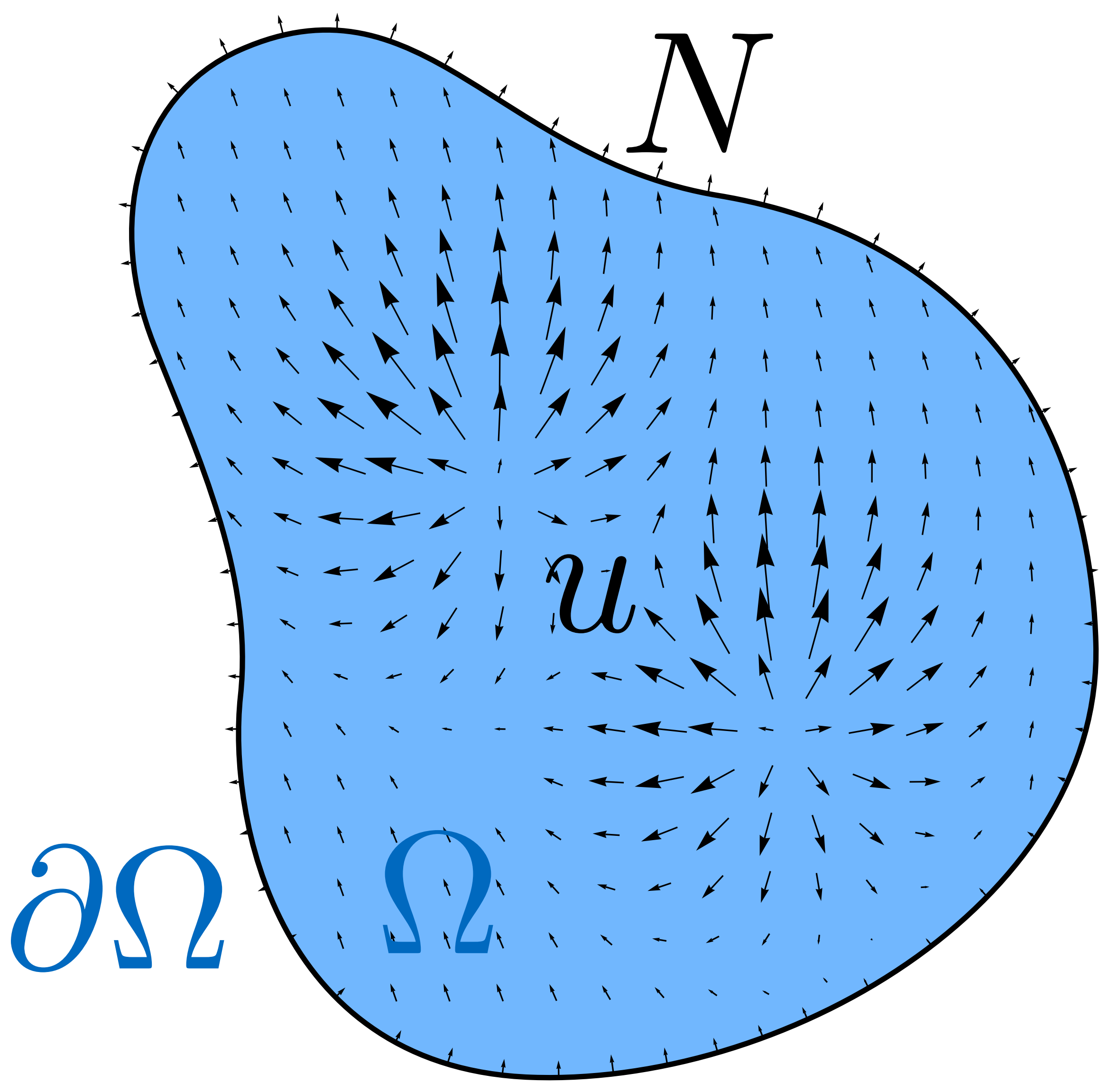
- **Many common fluids—like water!—are *incompressible,* i.e., no change in density (for all practical purposes...)**
- **Suppose the fluid has constant mass density ρ. Since mass cannot be created nor destroyed, we have**

$$0 = \frac{d}{dt}\int_{\Omega} \rho \; dx = \int_{\partial\Omega} \rho u \cdot N \; dx$$

no change in mass

total mass

total change in mass is the same as flux through the boundary



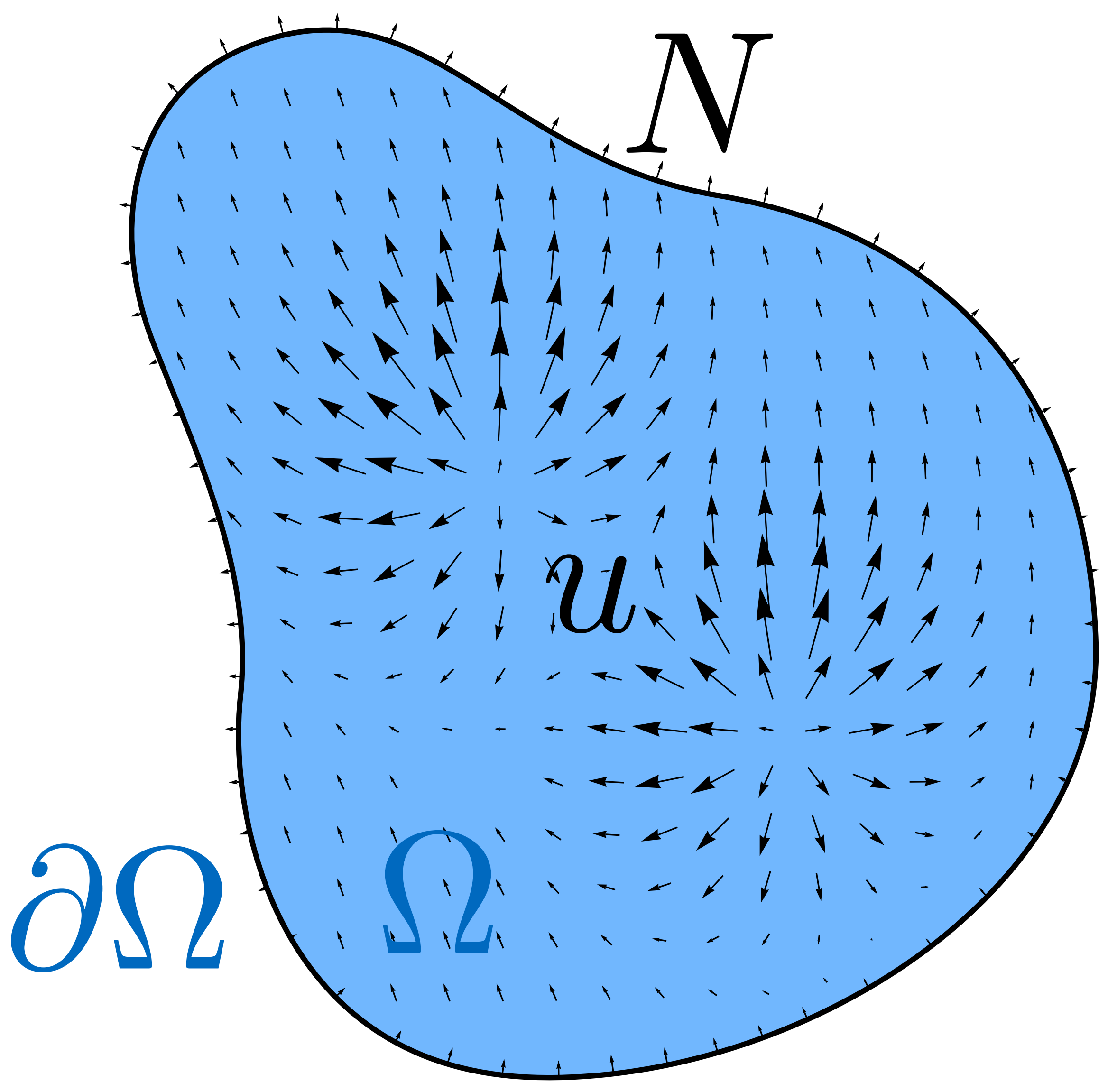
Incompressibility

- From here, apply *divergence theorem*

$$\int_{\partial\Omega} \rho u \cdot N \; dx = \int_{\Omega} \nabla \cdot (\rho u) \; dx$$

- Since this quantity is zero for every possible region Ω, it must in fact be zero pointwise:

$$\nabla \cdot (\rho u) = 0$$



- Since density is constant, we get our *incompressibility constraint*

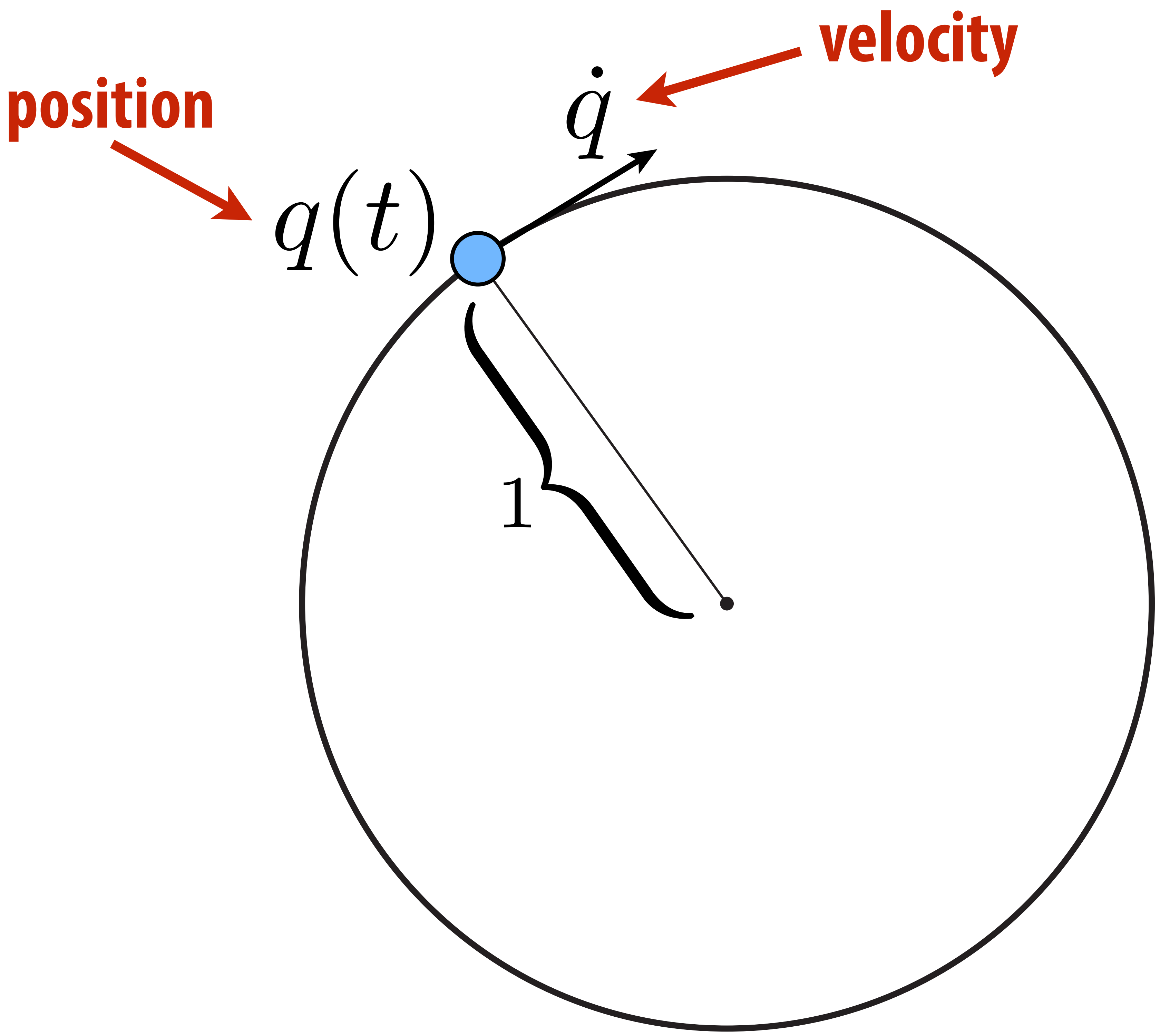
$$\nabla \cdot u = 0$$

"no divergence," i.e., *no sources or sinks allowed!*
(otherwise, fluid will COMPRESS!)

How do constraints enter into our equations of motion?

Simple Example: Bead on a Hoop

- **Consider a bead constrained to slide around on a hoop:**



$$g(q) := |q| - 1 = 0$$

constraint function

- **For simplicity, let's assume no external forces (gravity, etc.)**
- **How do we get the equations of motion?**

Constrained Lagrangian Mechanics

- **Return to Lagrangian picture, with a twist:**

Lagrangian

"Lagrange multiplier"

$$\mathcal{L} := K - U - \lambda g$$

kinetic energy

potential energy

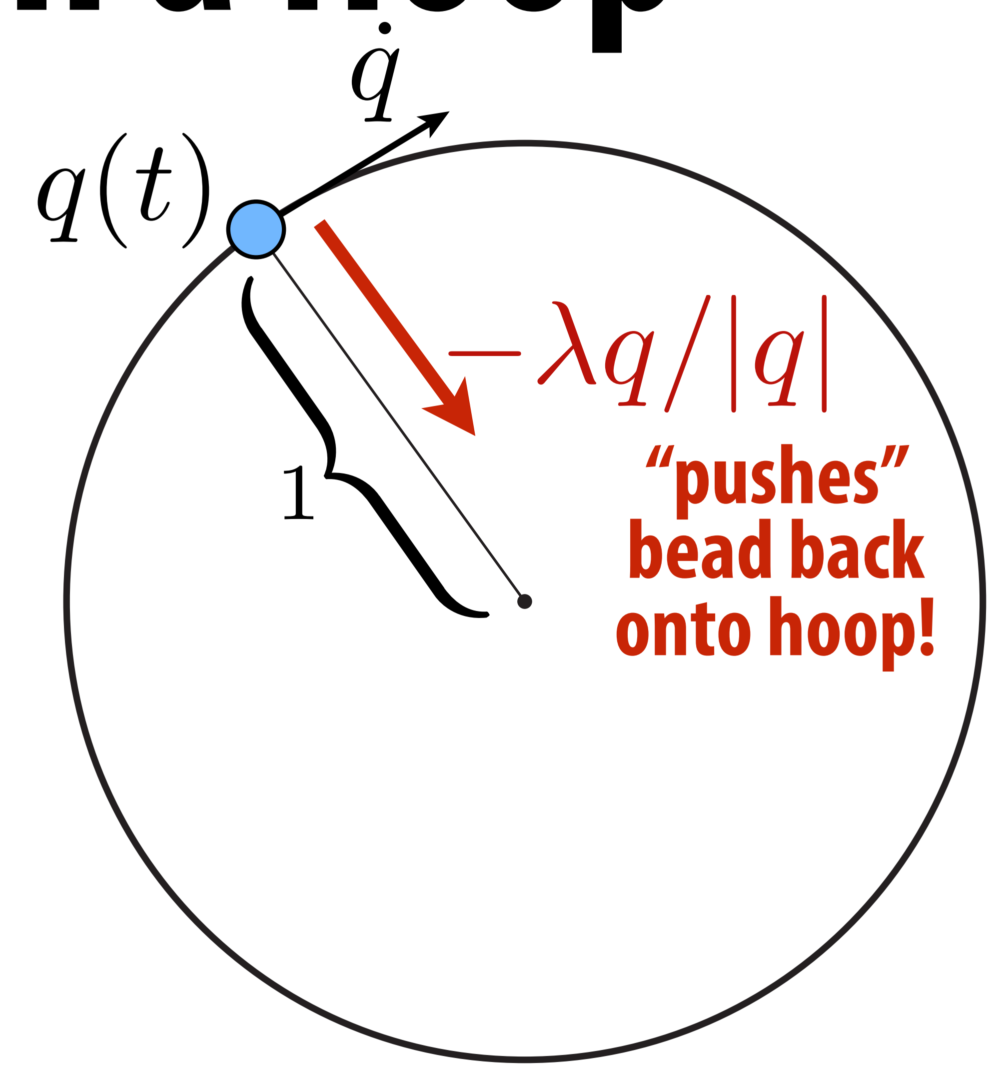
constraint function

- **As before, equations of motion are given by Euler-Lagrange:**

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}} = \frac{\partial \mathcal{L}}{\partial q}$$

- **Don't have to reason about force diagrams, etc.—just have to write down energy, constraints, and then "plug and chug"...**

Equations of Motion for a Bead on a Hoop



- **Let's try it for our bead on a hoop:**

KINETIC $$K = \frac{1}{2} m |\dot{q}|^2$$

POTENTIAL $$U = 0$$

CONSTRAINT $$g = |q| - 1$$

LAGRANGIAN $$\mathcal{L} = \frac{1}{2} m |\dot{q}|^2 - \lambda(|q| - 1)$$

$$\frac{\partial \mathcal{L}}{\partial q} = -\lambda q/|q|$$

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}} = m\ddot{q}$$

EQUATIONS OF MOTION

constraint force

$$\boxed{\begin{array}{rcl} m\ddot{q} & = & -\lambda q/|q| \\ |q| & = & 1 \end{array}}$$

Fluid Pressure

- So what's the force due to incompressibility in our fluid?
- Constraint was a *divergence-free* constraint on velocity:

$$\nabla \cdot u = 0$$

- Skipping the rest of the Lagrangian, the constraint term is

L2 inner product

Stokes' theorem

$$\langle\langle \lambda, \nabla \cdot u \rangle\rangle = -\langle\langle \nabla \lambda, u \rangle\rangle$$

Lagrange multiplier is now a *function!*

- Derivative with respect to velocity u? (Any guesses?)

$$-\nabla \lambda$$

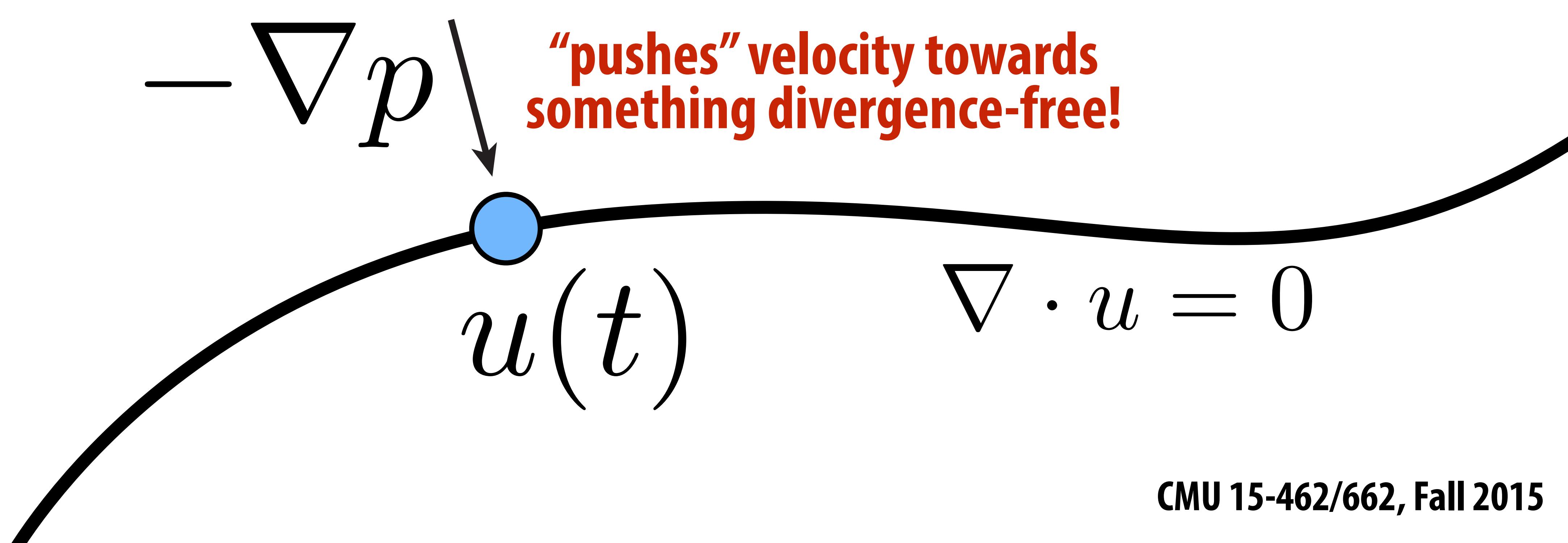
constraint force

Fluid Pressure

- **...Ok, what the heck does this force mean?**
- **For fluids, typically use "p" rather than "λ" to denote Lagrange multiplier:**

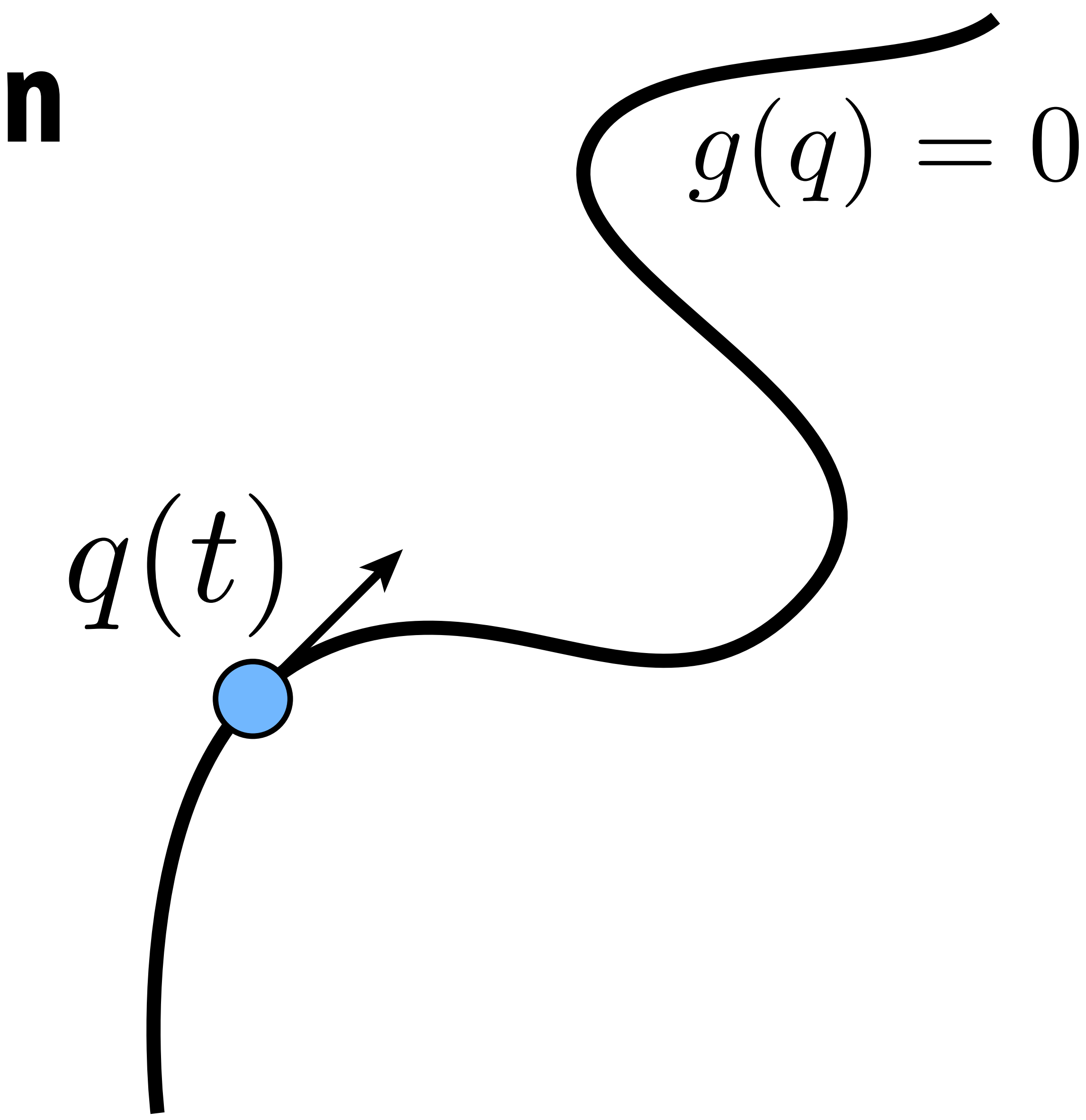
$$-\nabla p$$

- **Physically, corresponds to *pressure*, i.e., the force per unit area required to keep the fluid from compressing.**
- **Intuitively, prevents "particles from bumping into each-other"**
- **Geometrically, pressure is force pushing the velocity back onto the space of divergence-free vector fields (just like our bead...)**



Constraint Enforcement

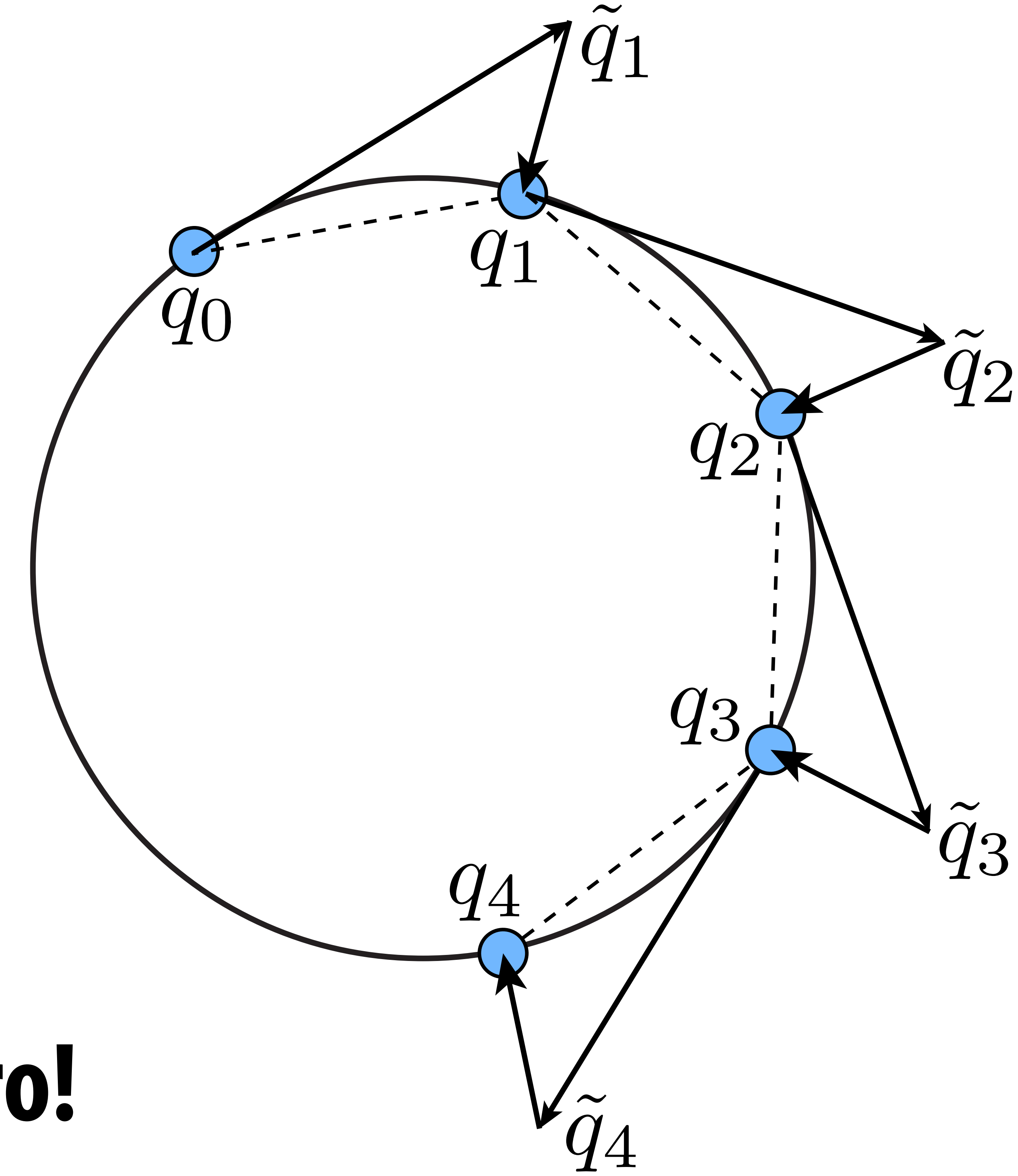
- Remember that equations of motion are *implicit*: they don't directly tell us the trajectory; they just say which equations any physical trajectory must satisfy.
- How do we integrate equations of motion w/ constraints?
- *Many* different techniques! One basic division
- REDUCED COORDINATES
  - use as few degrees of freedom as possible
  - e.g., angle of bead instead of (x,y)
  - not always possible/easy
- "FULL" COORDINATES*
  - use coords that are easy/intuitive to work with
  - somehow need to keep them on constraint curve...



*Point of philosophy: there is no canonical "full set of coordinates." Can always add more, cook up other descriptions, etc.

Constraint Projection

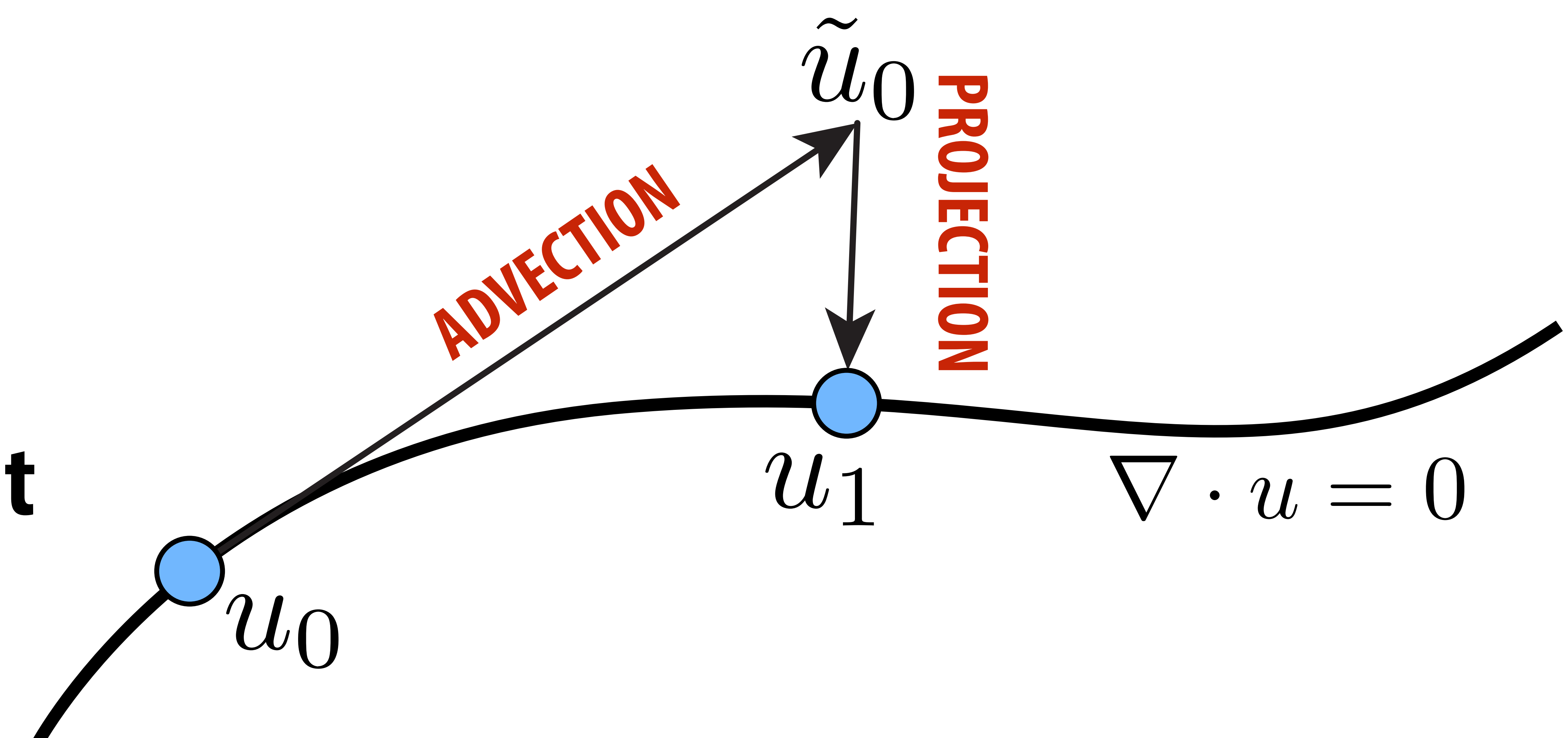
- Prototypical approach to constraints:
  1. Pretend there are no constraints; take a small step
  2. Project back onto constraint set
  3. Repeat
- E.g., bead on hoop:
  1. Move forward
  2. Normalize position
  3. Repeat
- Do we get the right dynamics?
- Yes—but only as time step goes to zero!
- For large time steps, we lose energy/momentum (just saw this for fluids!)



Constraint Projection for Fluids

- In fluids, analog of *"moving forward"* is advection
- Afterwards, velocity may no longer be divergence-free
- IDEA: Project onto nearest divergence-free field
- "Chorin's projection method"
- Same loop as before:
  1. Advect
  2. Project
  3. Repeat

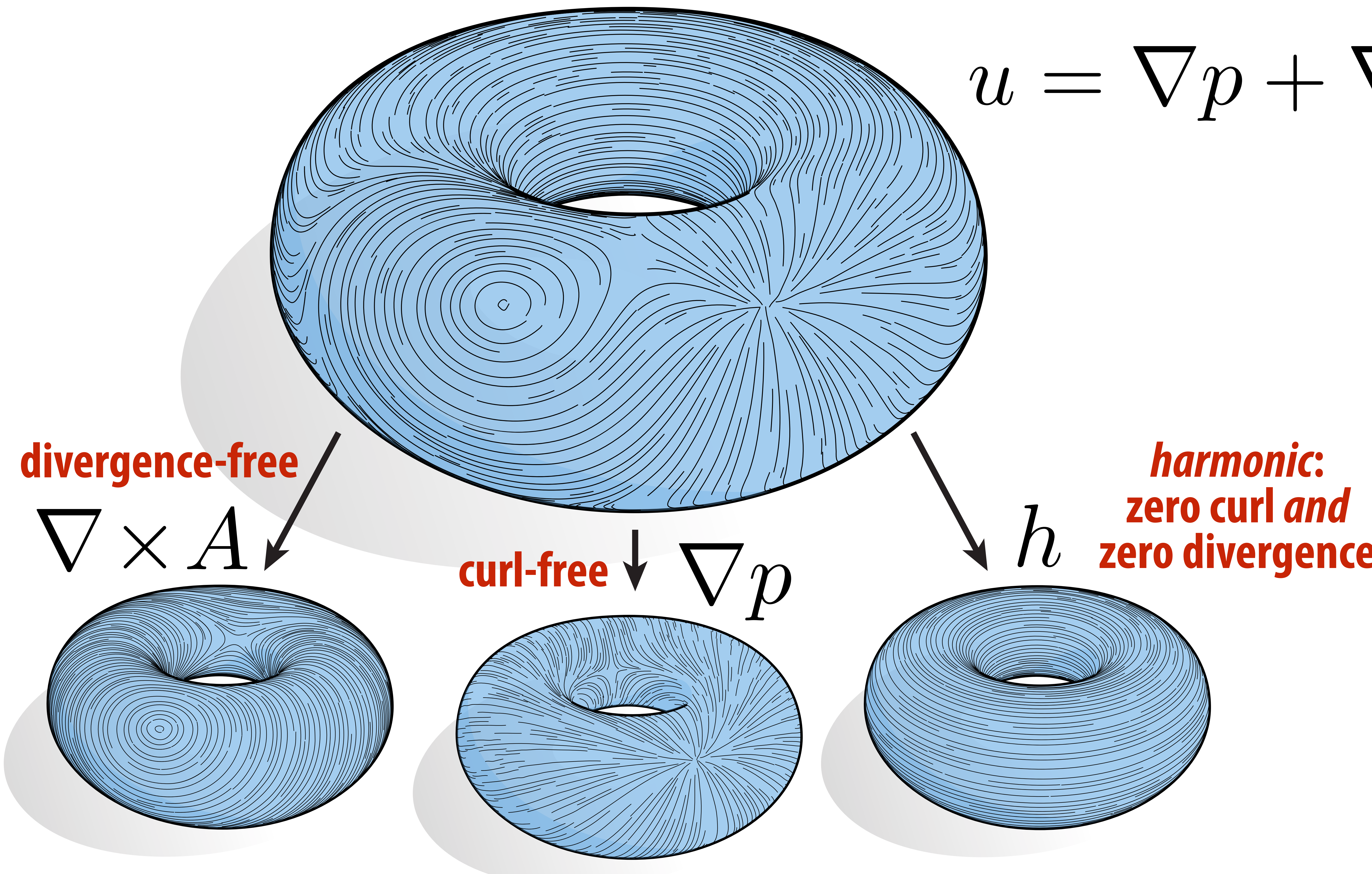
(Not the *only* method, but is the basis of many current techniques in graphics.)



Helmholtz-Hodge Decomposition

- **So, how do we find the closest divergence-free field?**
- **FACT: every vector field can be expressed as the sum of a divergence-free part, a curl-free part, and a harmonic part:**

$$u = \nabla p + \nabla \times A + h$$



Removing Divergence

- A divergence-free field should not have any divergence!
- So, seek and destroy: find divergence, then subtract it.
- Taking divergence of Helmholtz-Hodge decomposition yields

$$\nabla \cdot \tilde{u} = \underbrace{\nabla \cdot \nabla}_{\Delta} p + \cancelto{0}{\nabla \cdot (\nabla \times A)} + \cancelto{0}{\nabla \cdot h}$$

$$\Longleftrightarrow \quad \boxed{\Delta p = \nabla \cdot \tilde{u}}$$

Poisson equation!

- Overall, simple strategy for enforcing incompressibility:
  1. Solve (linear) Laplace equation
  2. Subtract pressure gradient to get div-free field

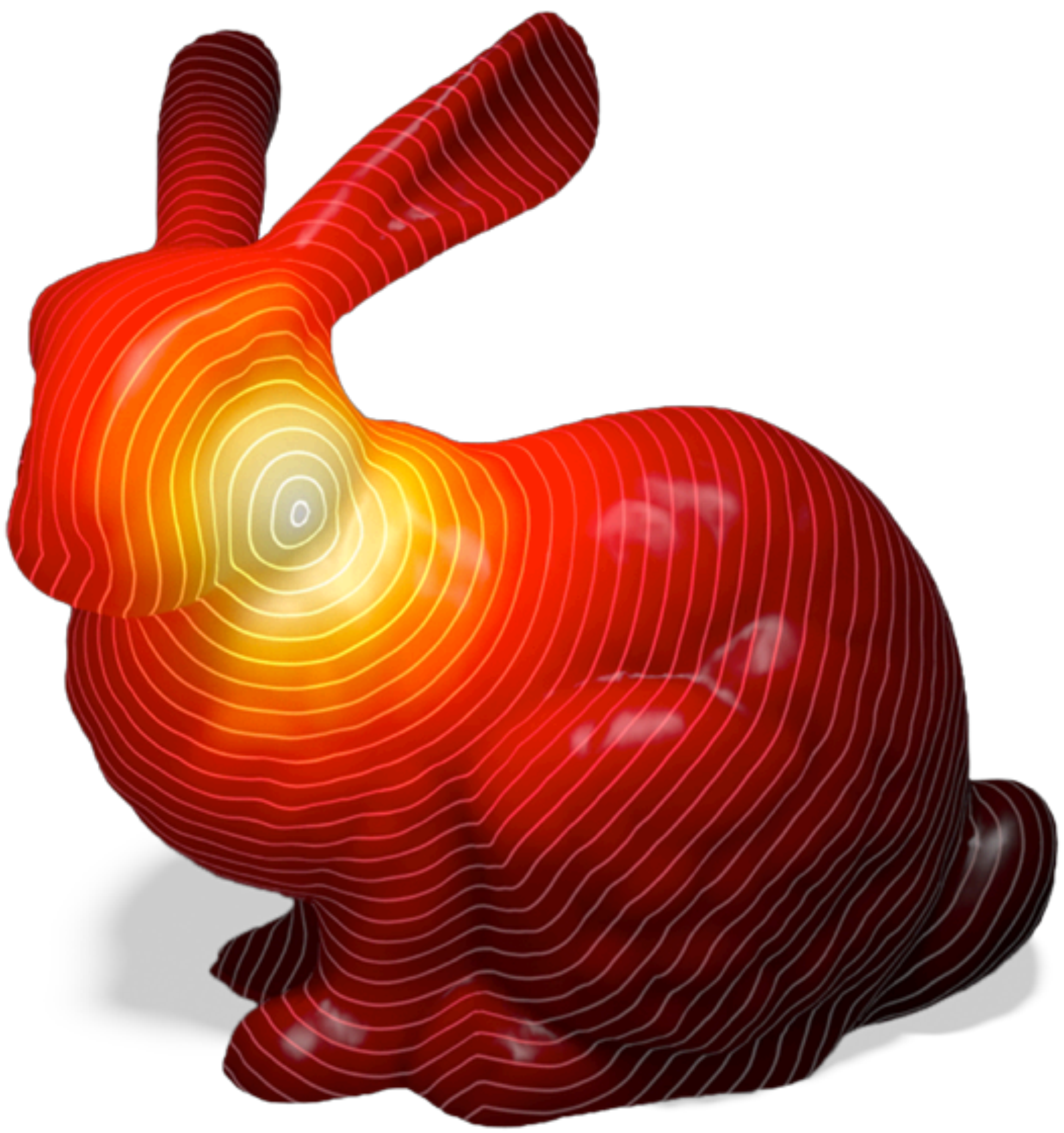
We now have the most important terms:

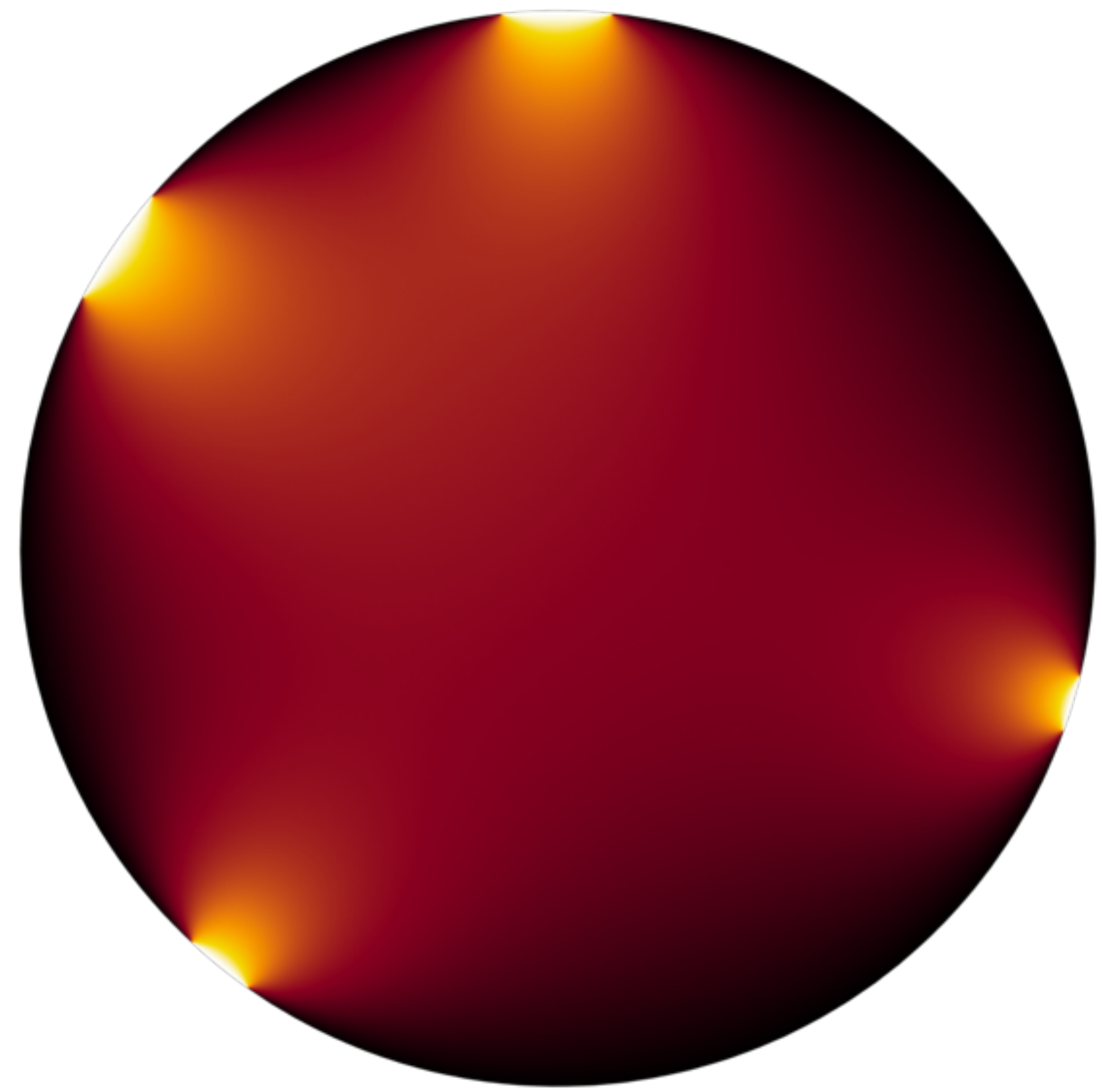
$$\dot{u} = -u \cdot \nabla u - \frac{1}{\rho}\nabla p + \nu \nabla^2 u + g$$

The rest is easy.

Reminder: Heat / Diffusion Equation

- "How does an initial distribution of heat spread out over time?"





- CLAIM: Models damping / viscosity in many physical systems

Fluid Viscosity

- For fluids, diffusion models "slowing down" of velocity due to internal "friction" (intermolecular forces)

kinematic viscosity

velocity

$$\nu \Delta u$$

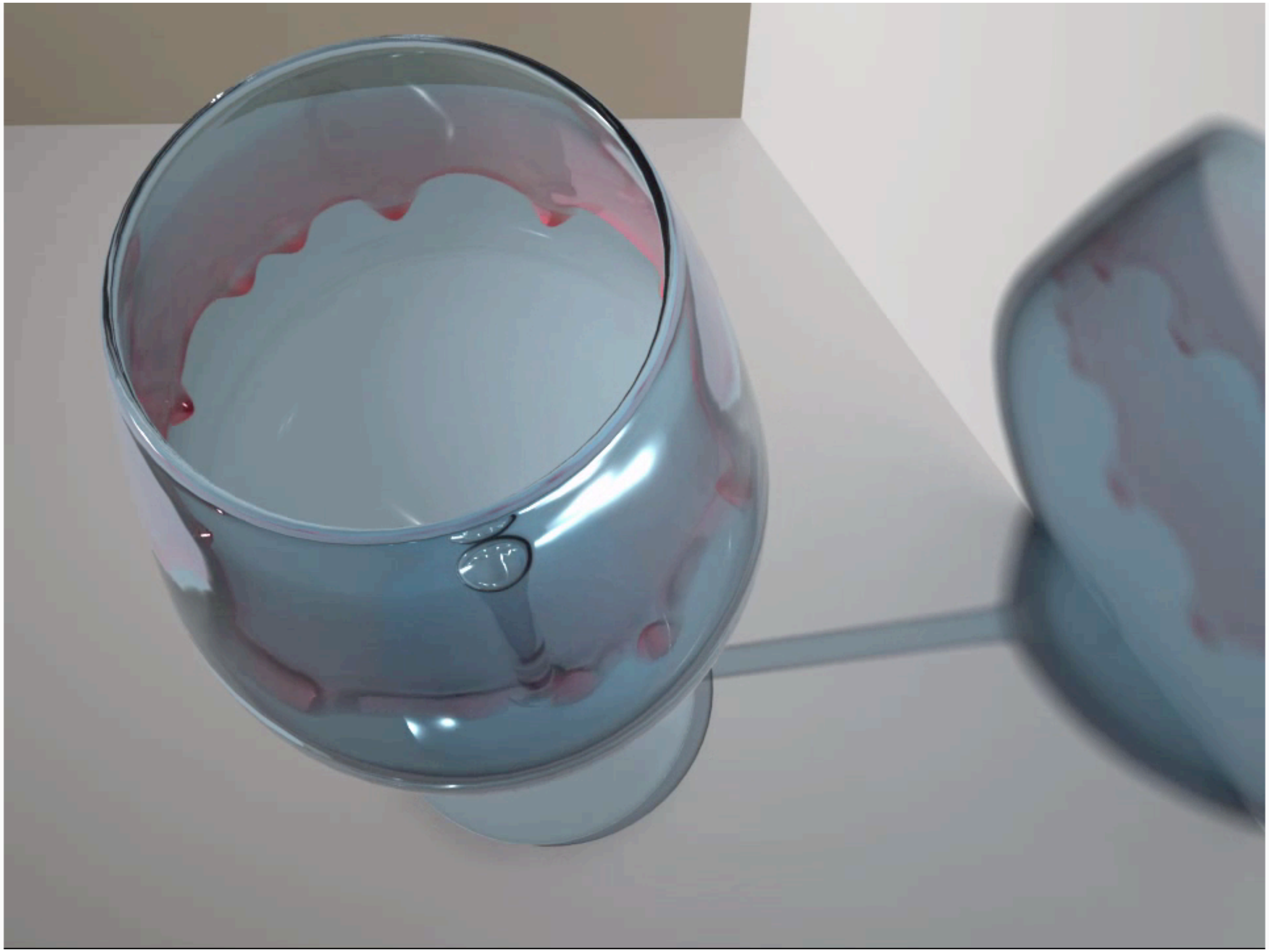
our old frenemy the Laplacian

- Viscosity much higher for e.g., honey; much lower for wine.
- Different regimes of viscosity lead to very different simulation techniques...

Honey Simulation

Bergou, Audoly, Vouga, Wardetzky, Grinspun, *"Discrete Viscous Threads"*

Wine Simulation



Azencot, Vantzos, Wardetzky, Rumpf, Ben-Chen, *"Functional Thin Films on Surfaces"*

And that's pretty much it.

$$\dot{u} = -u \cdot \nabla u - \frac{1}{\rho}\nabla p + \nu\nabla^2 u + g$$

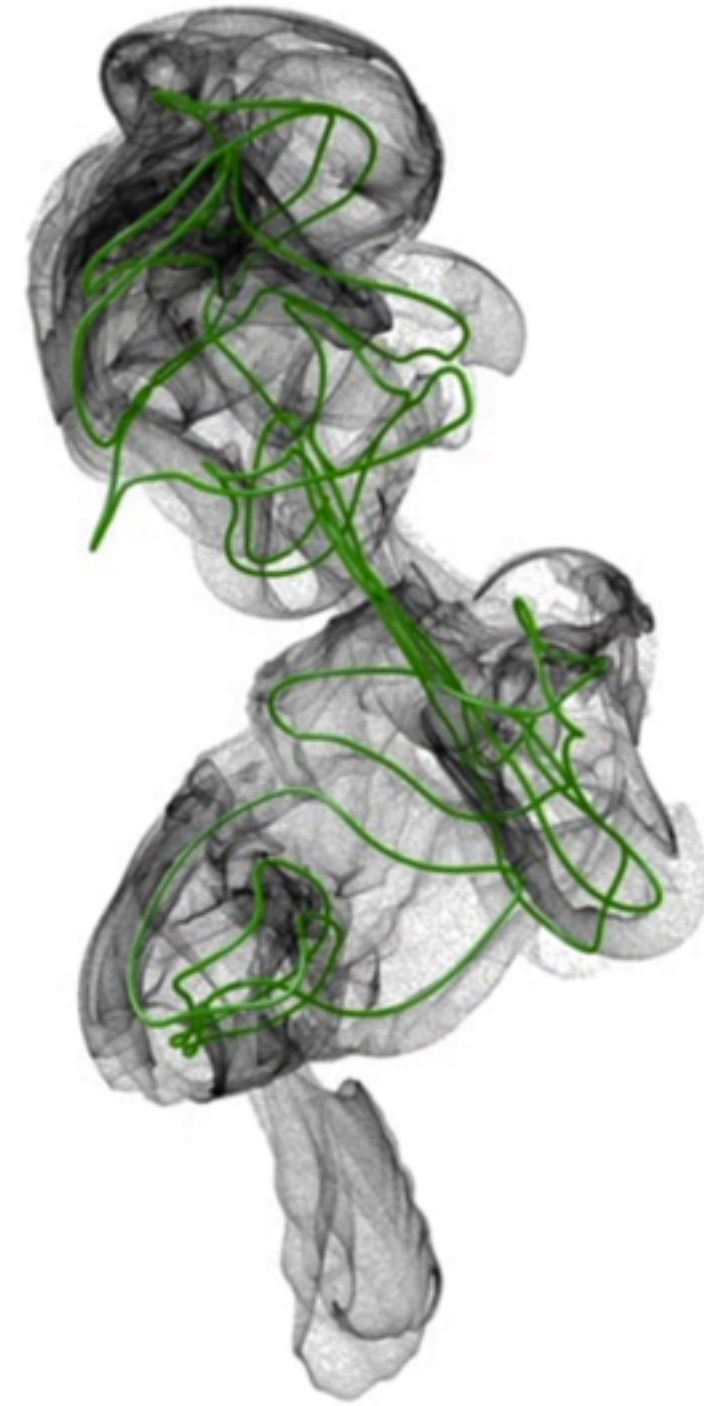
The rest is just "anything else" (like gravity).

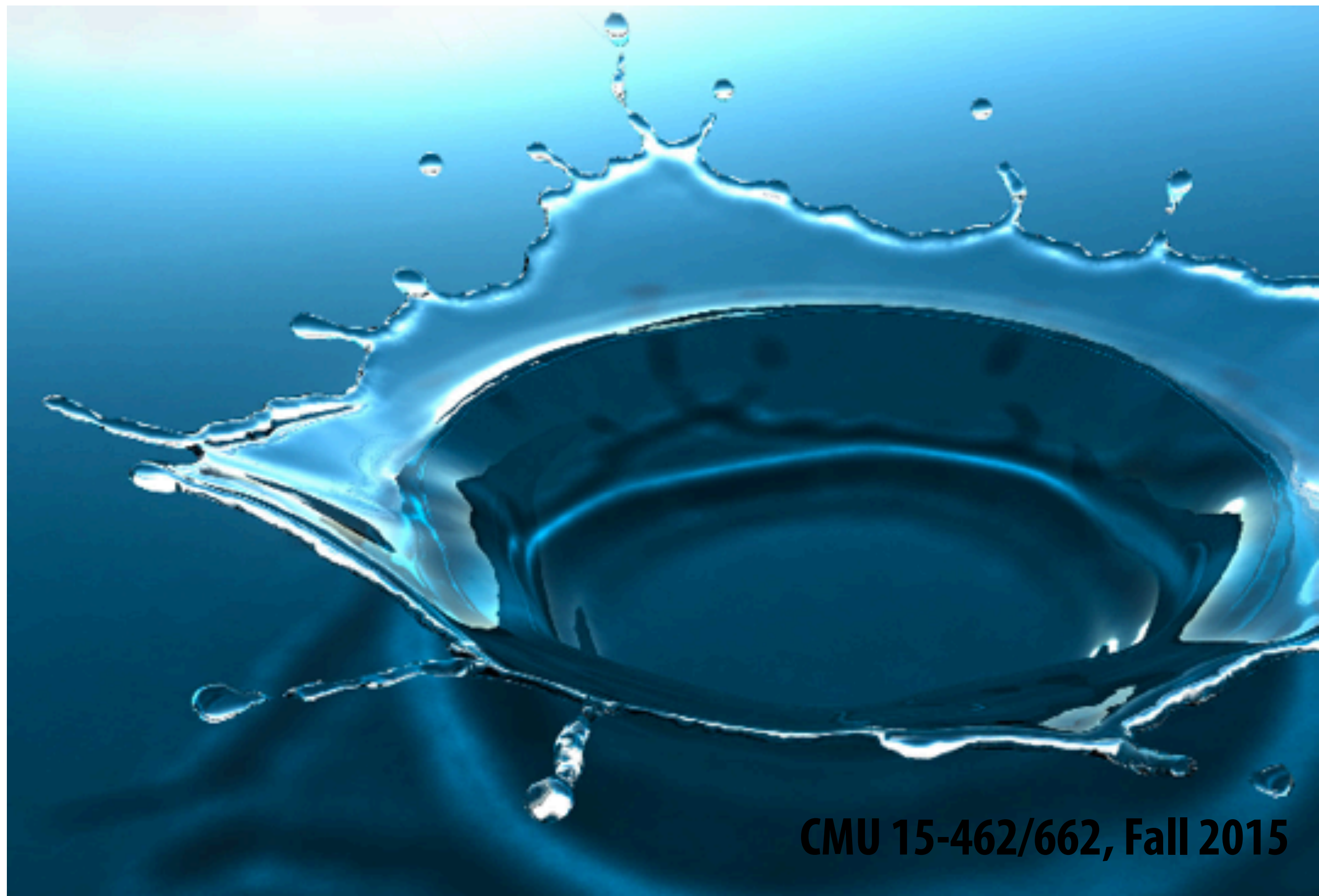
Notice that we used two of our model equations: the Poisson and heat equations.

(We weren't kidding when we said these things show up *everywhere* in simulation!)

An ocean of knowledge... *be a sponge!*

- *So* much more to say about fluids:
- Other forms of fluid equations
  - vorticity increasingly popular in graphics
  - leads, e.g., to nice 1D vortex filaments
- Free surface tracking
  - e.g., simulation of liquids
  - track surface with grid OR particles OR mesh
- Various fluid phenomena
  - fire, smoke, viscoelastic, ...
- Pure particle-based methods (SPH, MD)
- Go out there and make a splash!





Next up: Virtual Reality