Abstraction and Composition in Modeling and Simulation

SIAM CSE 2023 Minisymposium on DEC and FEEC

Luke Morris, Andrew Baas, Jesus Arias, Maia Gaitlin, and James Fairbanks

$HOME=/UF/HWCOE/CISE

Spectrum of Scientific Computing Technology

Programs that have explicit representations of the model

Arbitrary Code

Domain Specific Languages

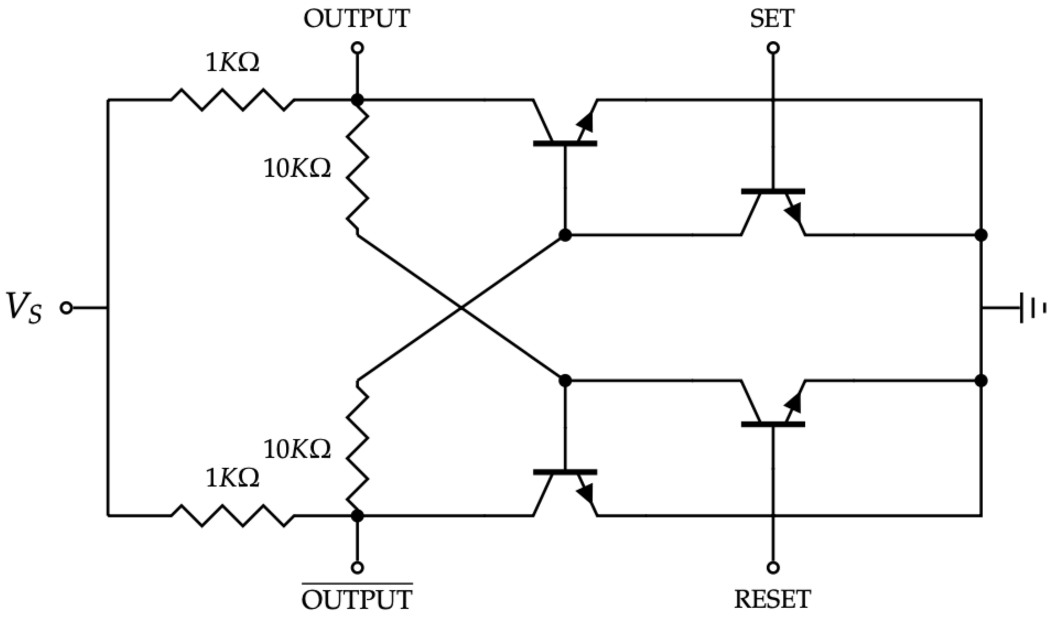
Modeling Frameworks

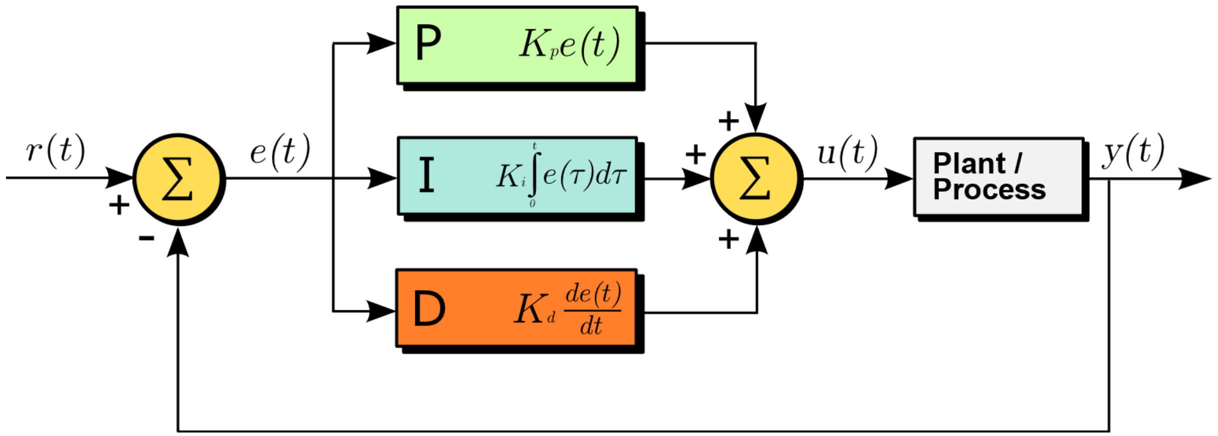
Computer Algebra Systems

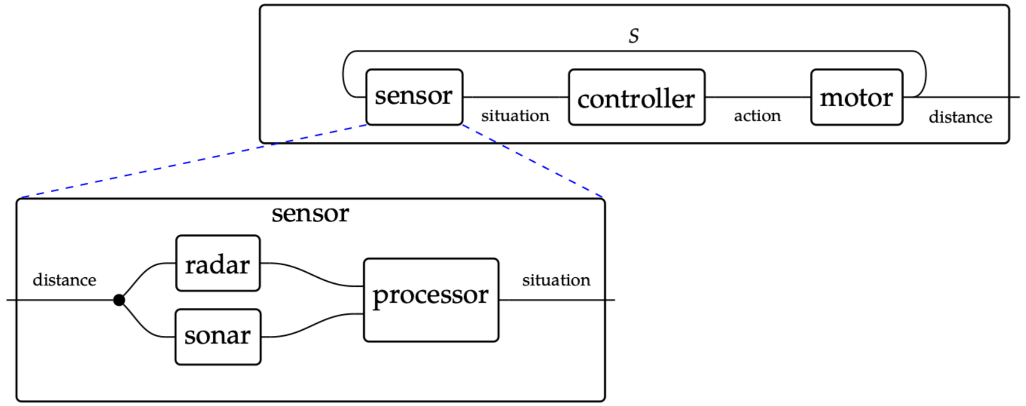
Symbolic Math Toolbox
Perform symbolic math computations

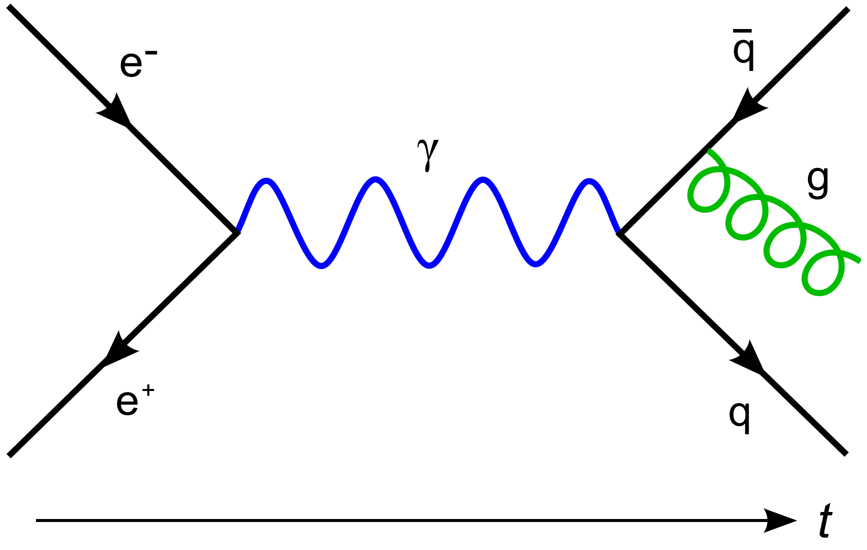
SymPy

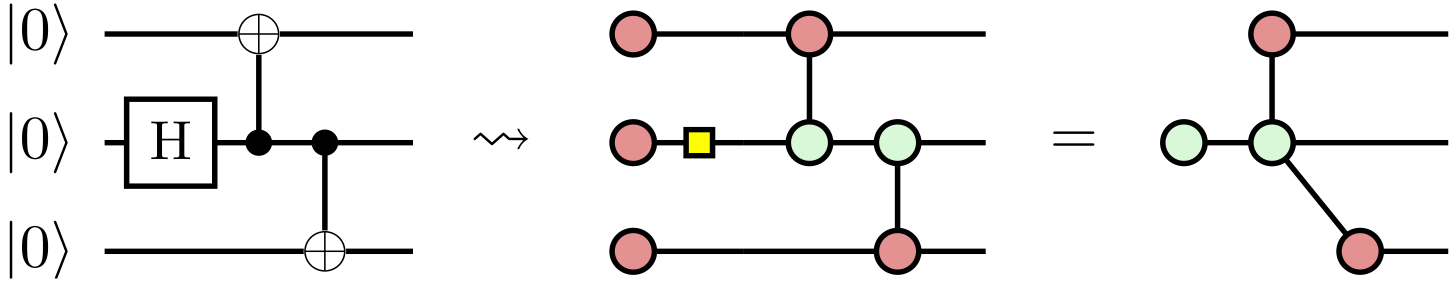
Formal Scientific Diagrams



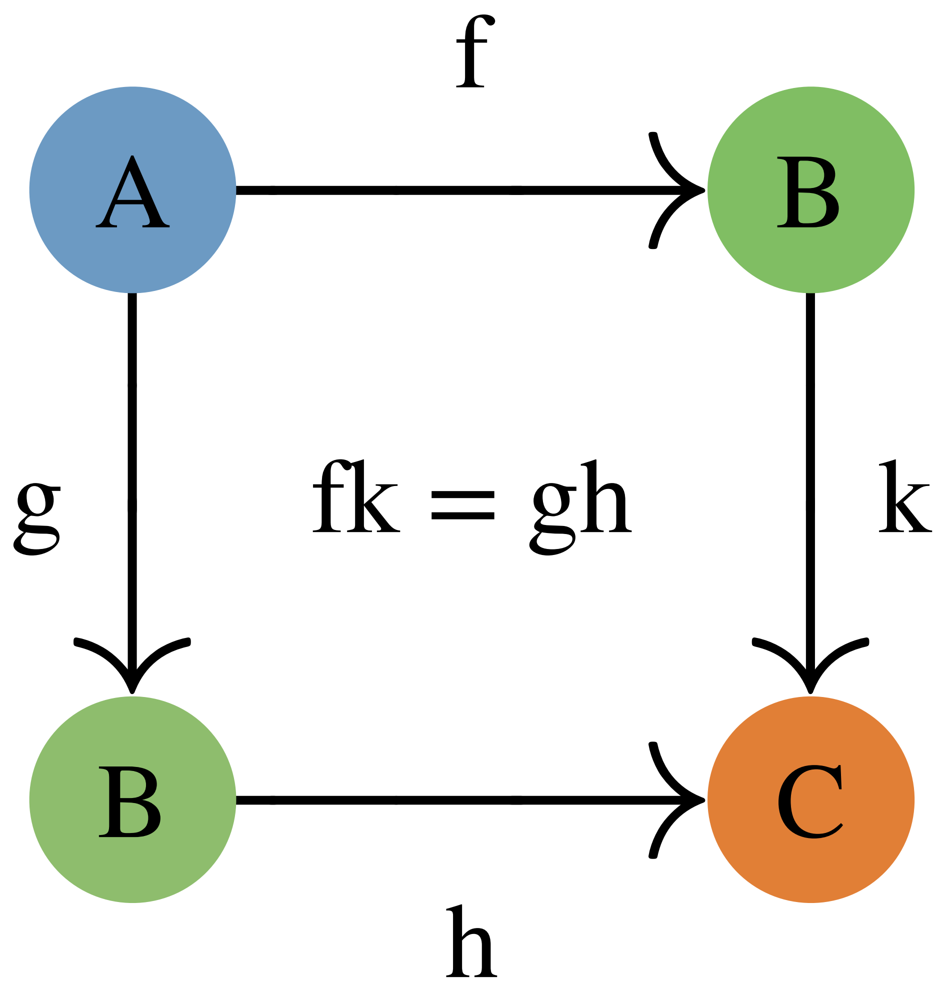








Category Theory

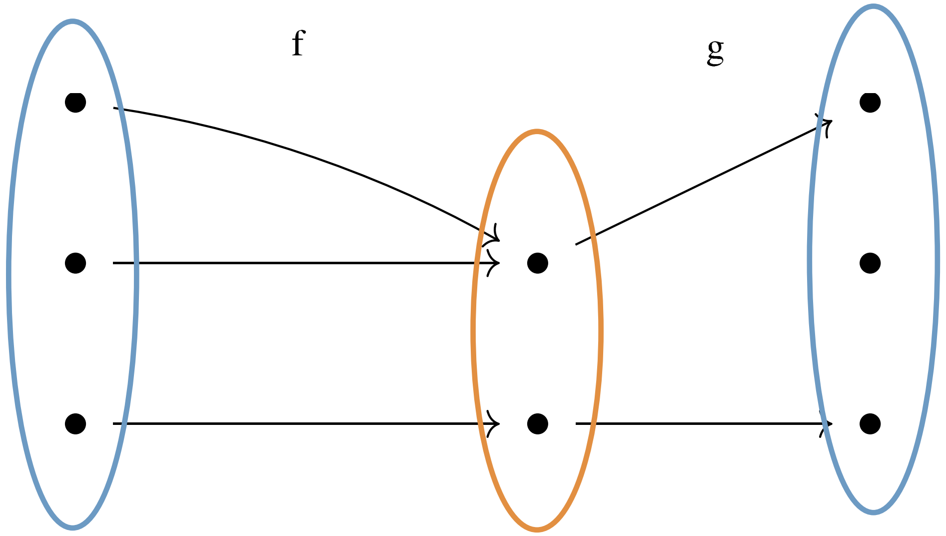


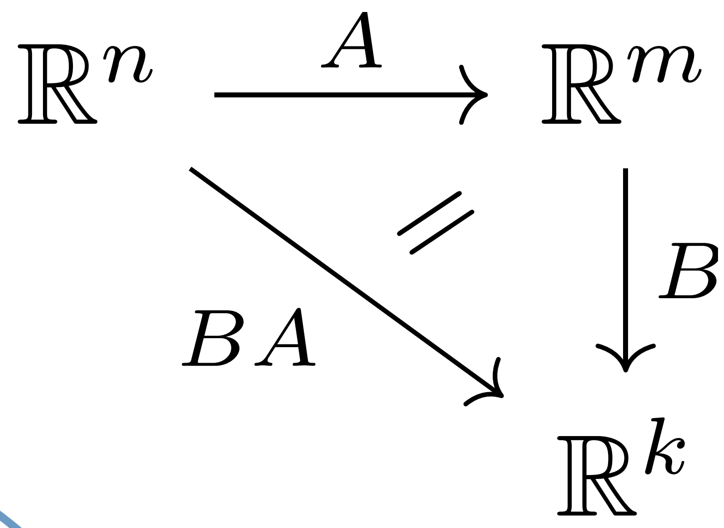
$$C = (Ob, Hom)$$

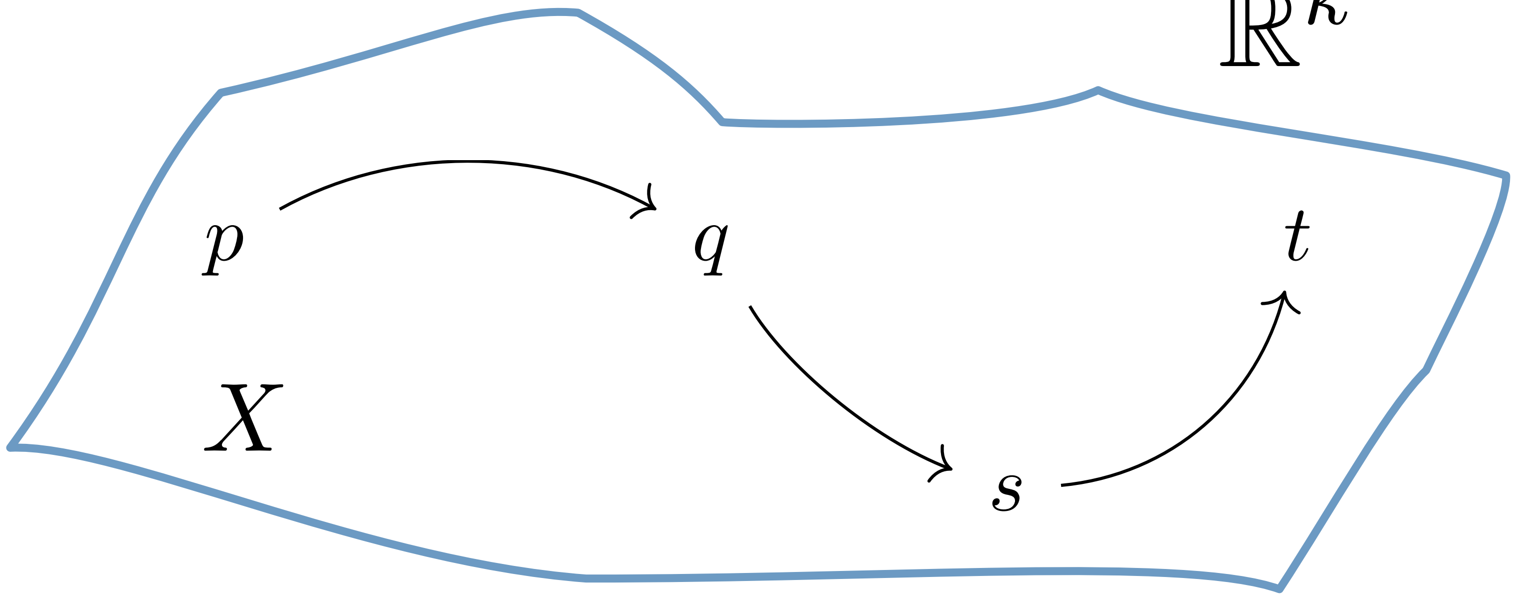
$$Ob : Set$$
$$\forall A, B : Ob \vdash Hom(A, B) : Set$$
$$\forall A : Ob \vdash id(A) : Hom(A, A)$$
$$\forall A, B, C : Ob \vdash \circ_{A,B,C} : Hom(A, B) \times Hom(B, C) \to Hom(A, C)$$

Some Example Categories

- Sets and Functions
- Finite Sets and Functions
- Vector Spaces and Linear Maps (Matrices)
- Topological Spaces and Continuous Functions
- Convex Spaces and Convex Functions
- Points in a Space and Paths in that Space
- Dynamical Systems and Changes of Coordinates



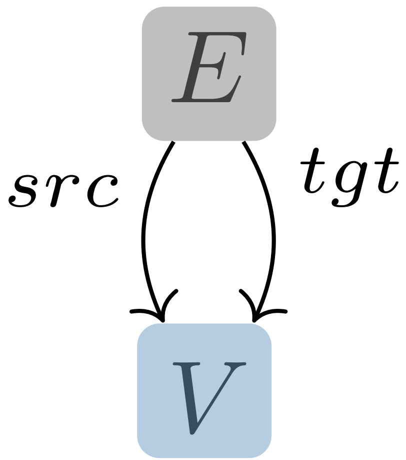


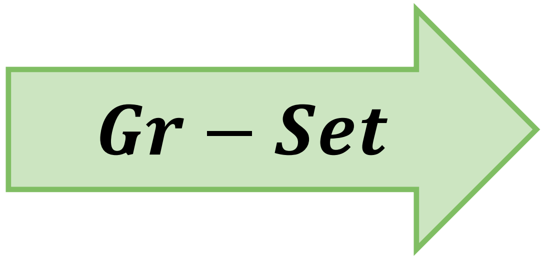


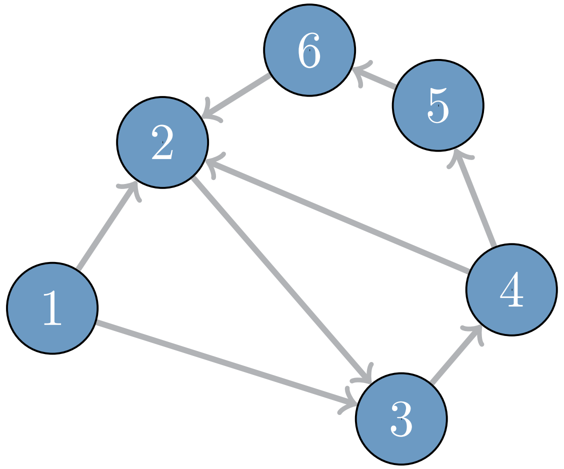
$\mathcal{C}$-Sets: Categorical Data Structures

Graphs are ubiquitous because they a simple & useful structure

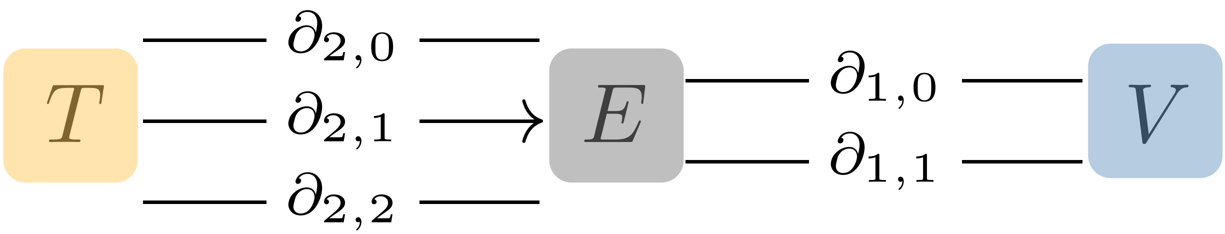
Gr

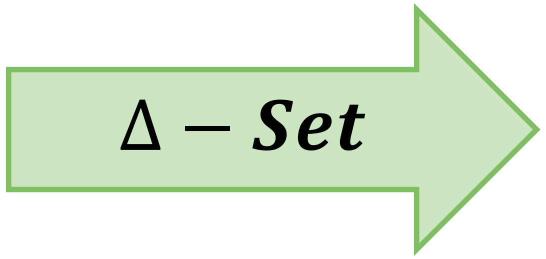


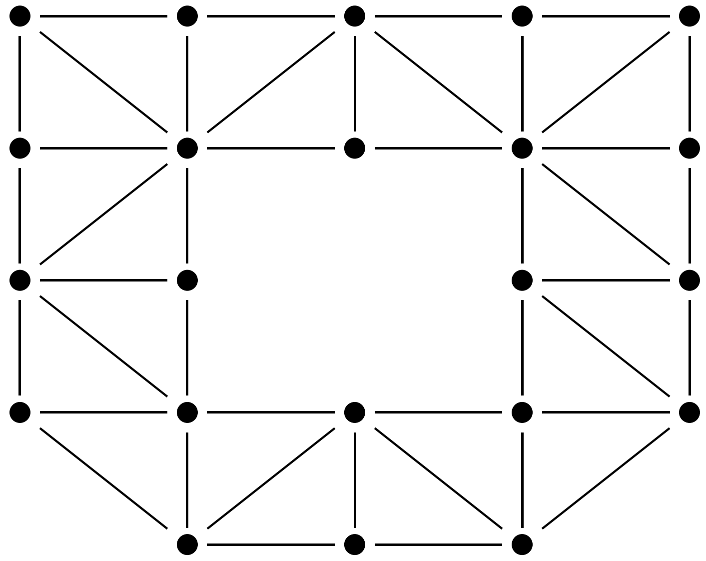




Δ







State of the Art circa 2020

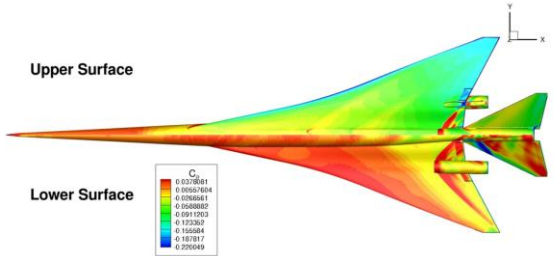
Multiphysics Simulation

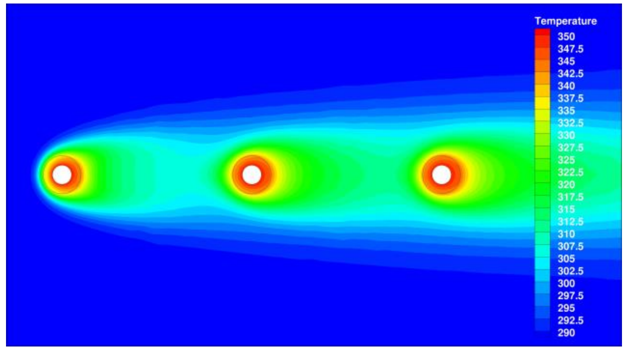
SU2: An Open-Source Suite for Multiphysics Simulation and Design

Thomas D. Economon*
Stanford University, Stanford, California 94305
Francisco Palacios†
The Boeing Company, Long Beach, California 90808
and
Sean R. Copeland,‡ Trent W. Lukaczyk,‡ and Juan J. Alonso§
Stanford University, Stanford, California 94305

DOI: 10.2514/1.J053813

2015





Discrete Exterior Calculus

NUMERICAL METHOD FOR DARCY FLOW DERIVED USING DISCRETE EXTERIOR CALCULUS

ANIL N. HIRANI, KALYANA B. NAKSHATRALA, AND JEHANZEB H. CHAUDHRY

2008

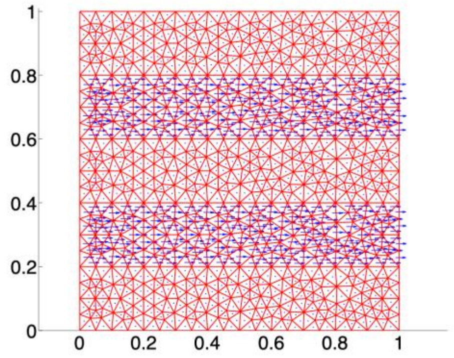


Figure 8: Layered medium with 2 different permeability patterns. The domain has 5 layers with alternating permeability. In the top figure the permeability k, from bottom layer to top is 5, 10, 5, 10 and 5. In the bottom figure the permeability k is 1, 10, 1, 10 and 1. The computed flux is visualized as a vector field.

Discrete exterior calculus discretization of incompressible Navier-Stokes equations over surface simplicial meshes

2016

Mamdouh S. Mohamed[a,1,*], Anil N. Hirani[b], Ravi Samtaney[a]

[a]*Mechanical Engineering, Physical Sciences and Engineering Division, KAUST, Jeddah, KSA*
[b]*Department of Mathematics, University of Illinois at Urbana-Champaign, IL, USA*

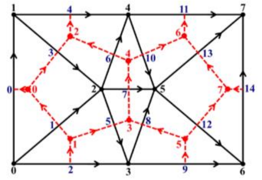
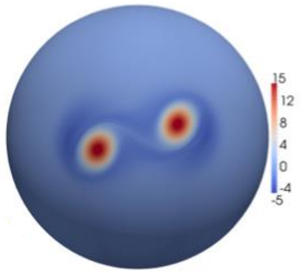


Figure 1: A sample simplicial mesh in 2D showing the primal simplices (in black color) and their dual cells (in red color). The positive orientation of the primal 2-simplices and dual 2-cells is counterclockwise.



Category Theoretic Dynamical Systems

ALGEBRAS OF OPEN DYNAMICAL SYSTEMS ON THE OPERAD OF WIRING DIAGRAMS

DMITRY VAGNER, DAVID I. SPIVAK, AND EUGENE LERMAN

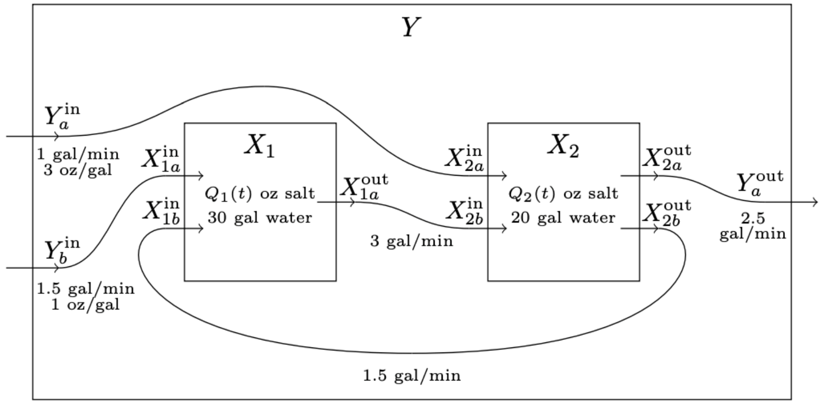
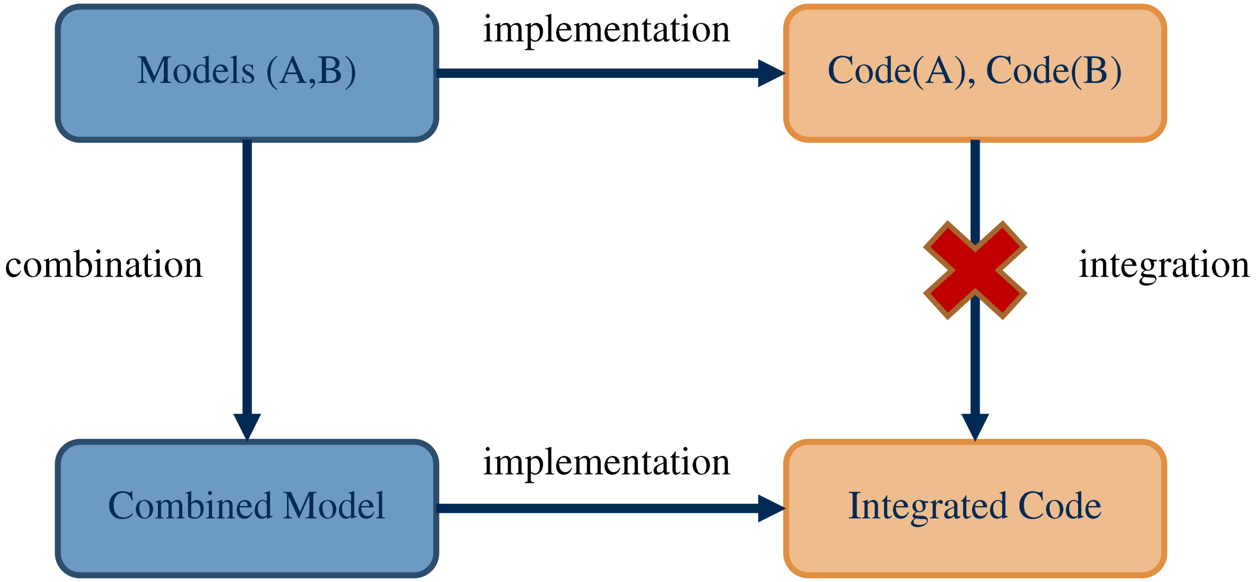


FIGURE 9. A dynamical system from Boyce and DiPrima interpreted over a wiring diagram $\Phi = (X_1, X_2; Y; \varphi)$ in $\mathcal{O}\mathbf{W}$.

2018

Abstract mathematics only
No software or simulations

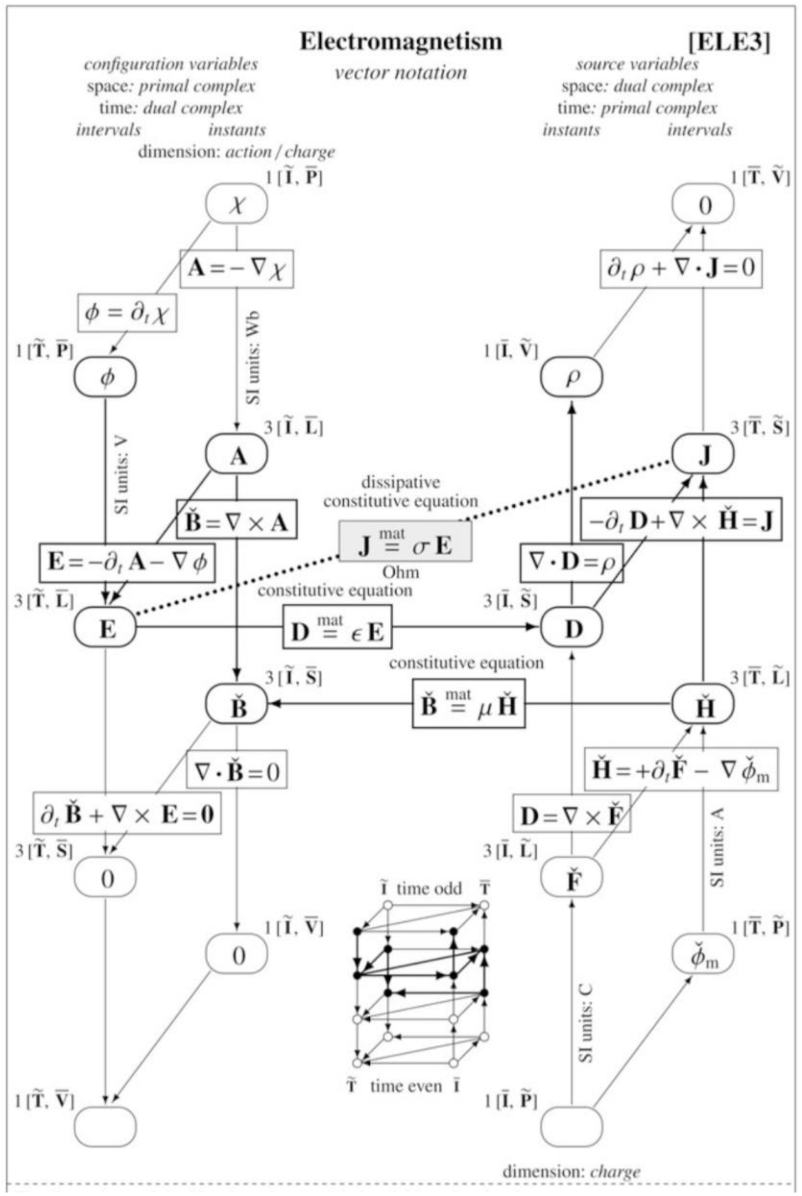
Automated Construction of Complex Simulators



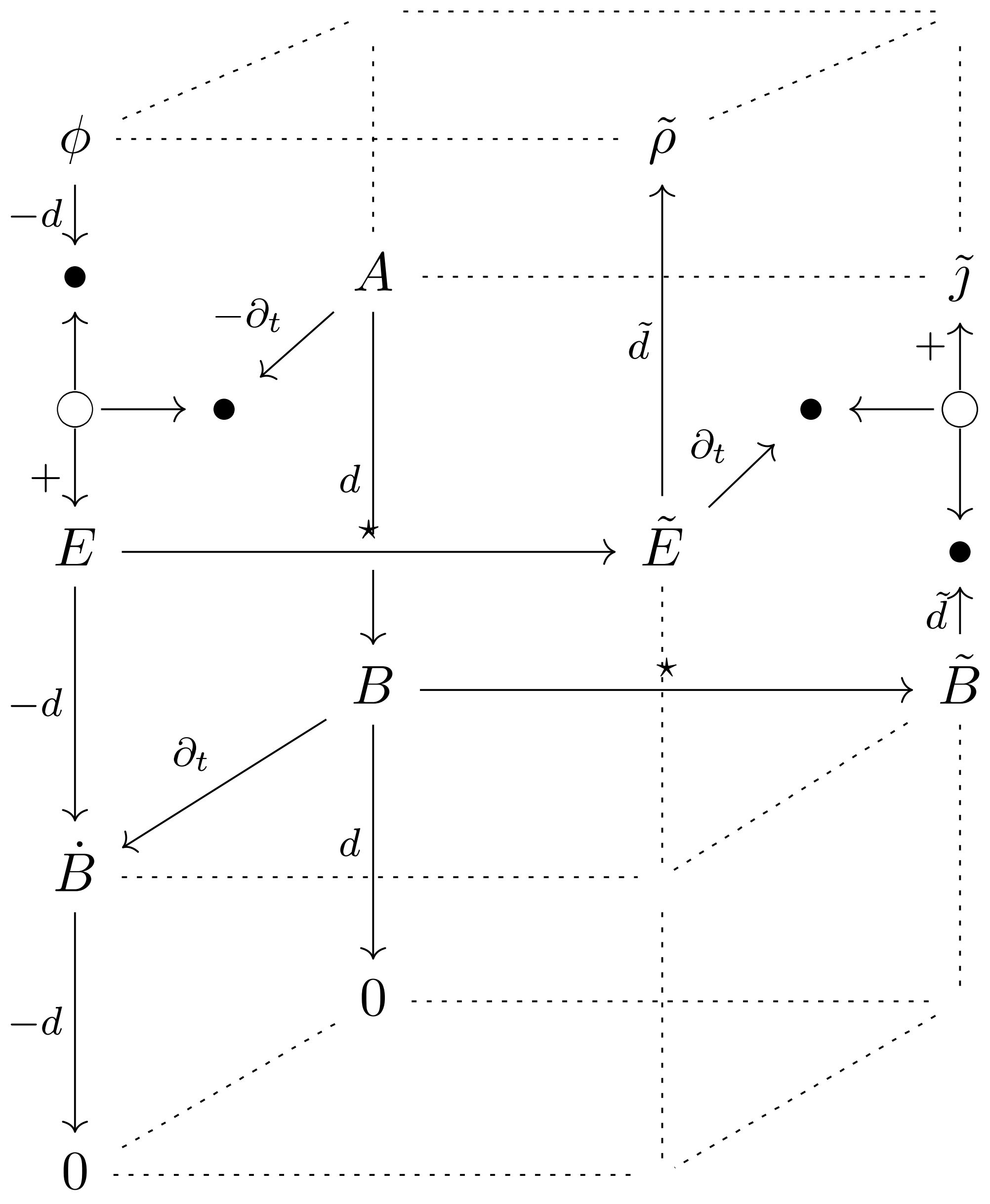
- Plan to get high performance by fusing models at the math level
- Code generation is automated so we can exceed human capability

Motivation: Tonti Diagrams

Tonti Diagram for E&M

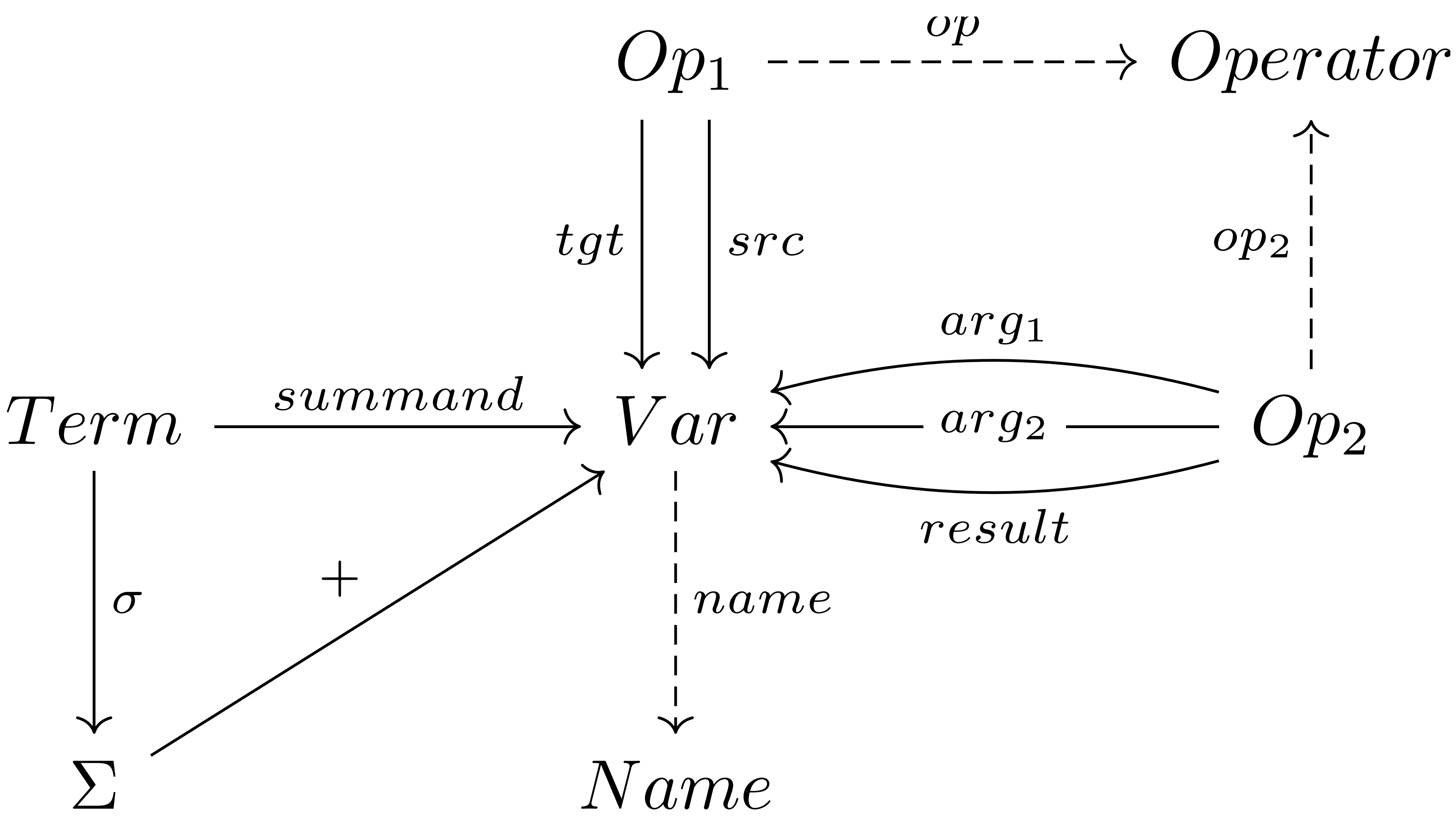


DECAPODE for Electricity and Magnetism



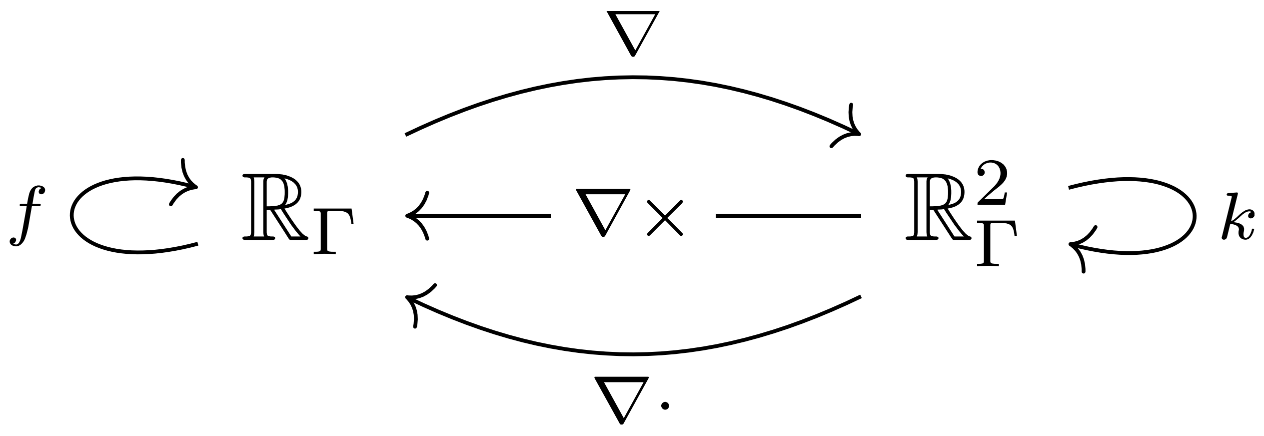
DECAPODES are C-Sets

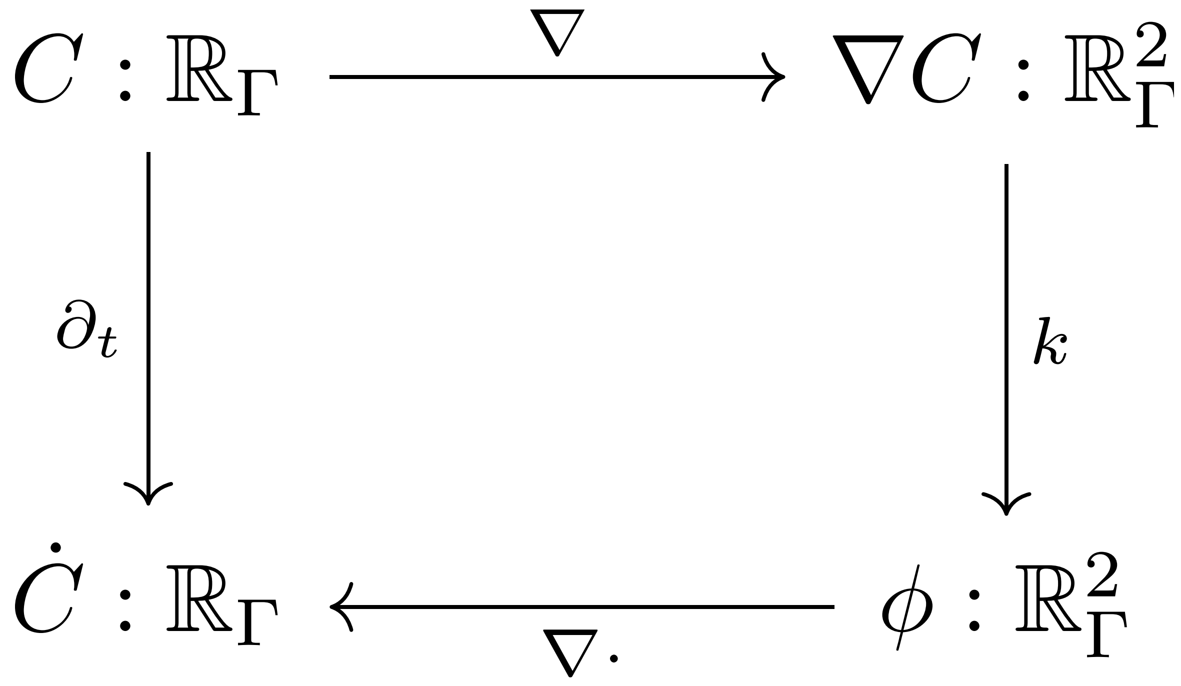
- The data of the equational presentation can be expressed as an ACSet over this schema
- Dashed arrows are Data Attributes
- Slice Construction gives a category of Typed Decapodes.



Design Based on Hyp-Sigma approach in
Filippo Bonchi and Fabio Gadducci and Aleks Kissinger and Pawel Sobocinski and Fabio Zanasi, "String Diagram Rewrite Theory I: Rewriting with Frobenius Structure", In J. ACM, vol. 69, no. 2, pp. 14:1–14:58, 2022.

Vector Calculus Diagrams

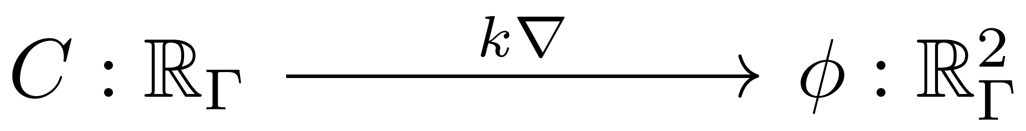




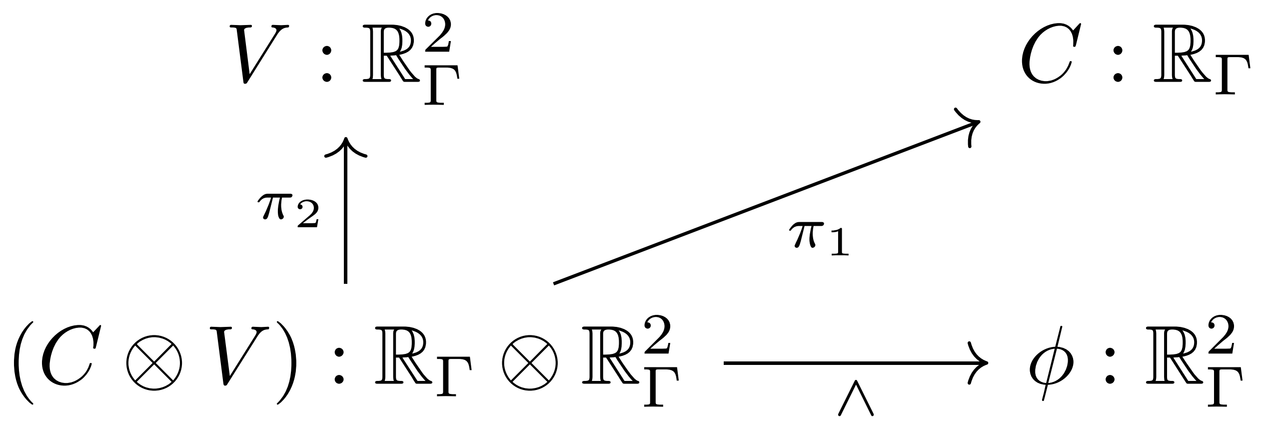
- Scalar and vector fields are the objects
- Differential operators Div, Grad, Curl are arrows
- Allow functions (f/k) on scalar/vector fields too

Physical Principles in Vector Calculus

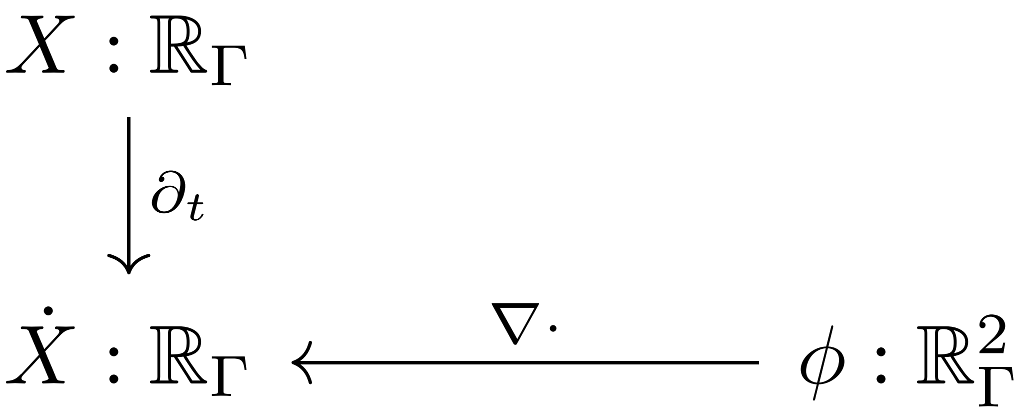
Fick's Law of Diffusion



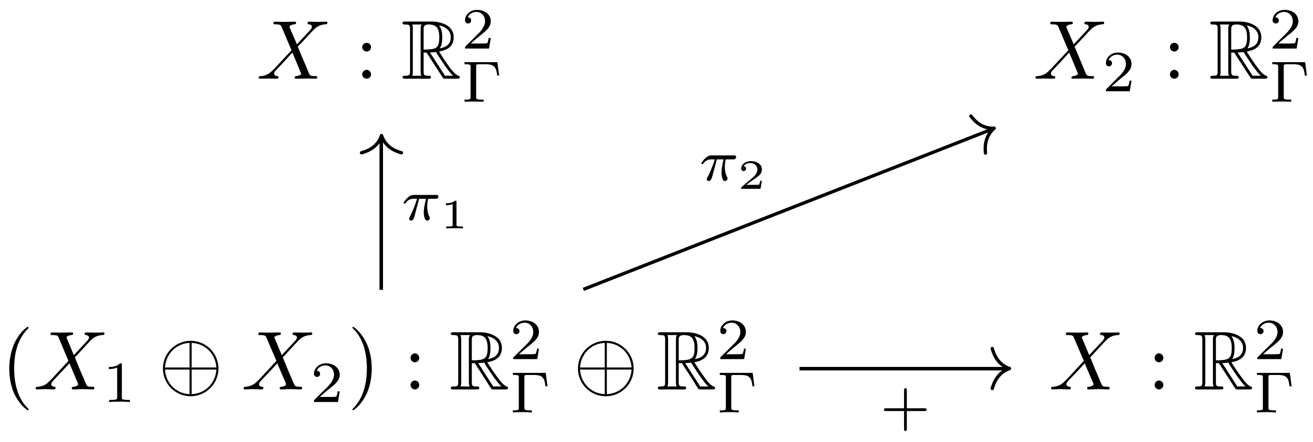
Scalar Transport by Advection



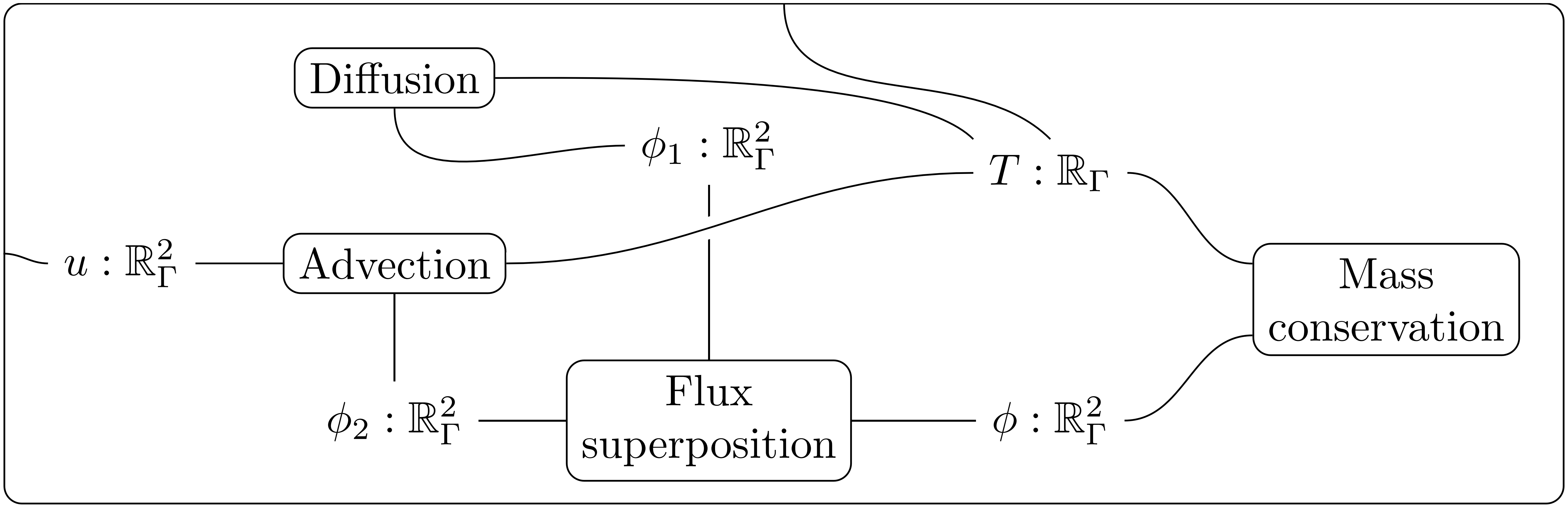
Conservation of Mass



Superposition of Flux

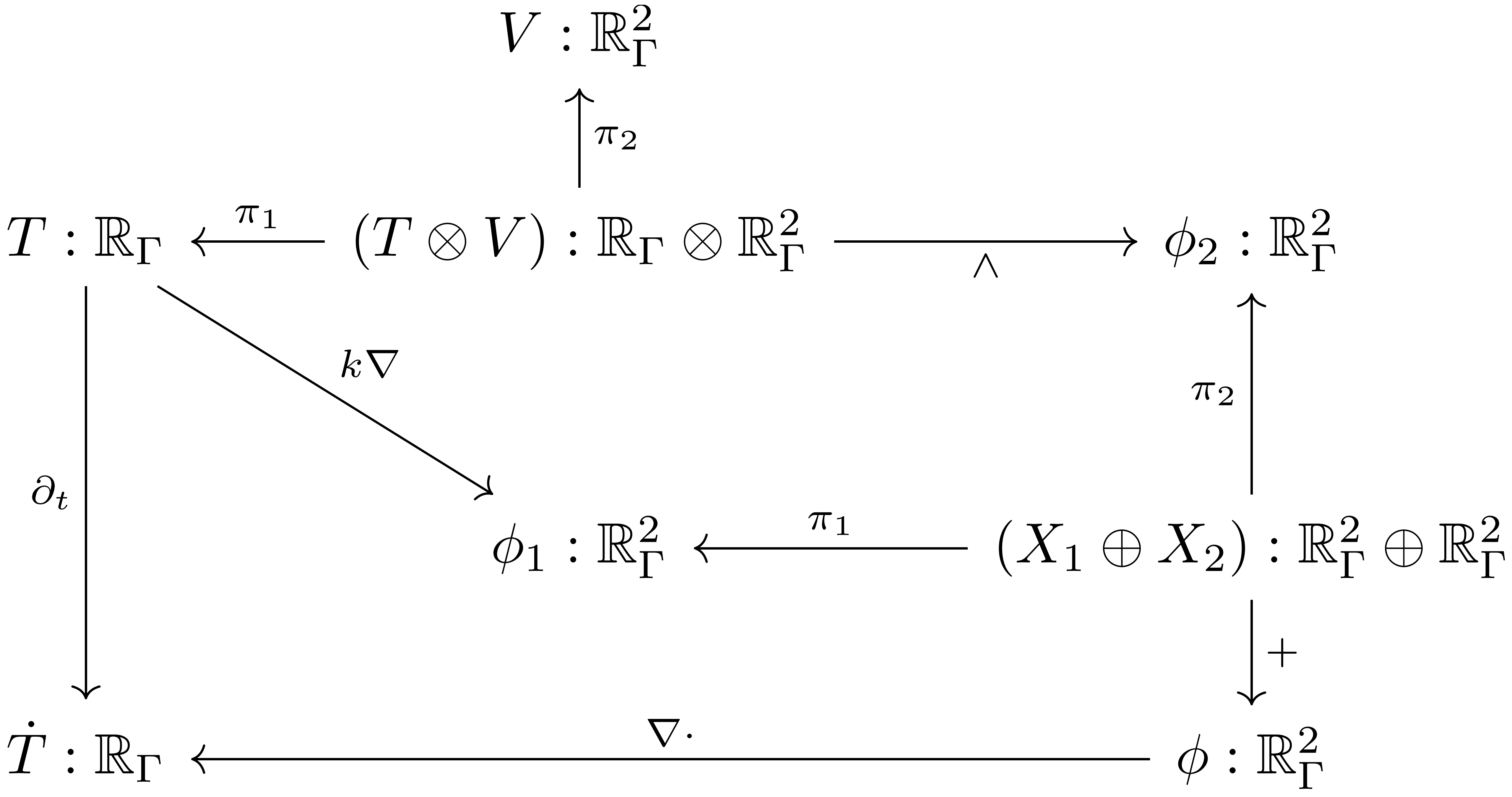


Composing Multiphysics with Wiring Diagrams



- Junctions are Variable : Domain
- Boxes are subsystem (component physics)
- Ports are exposed variables in the subsystems
- Wires connect ports to junctions with matching domains
- Outer Ports allow hierarchy by exposing variables of the composite system to next level of hierarchy

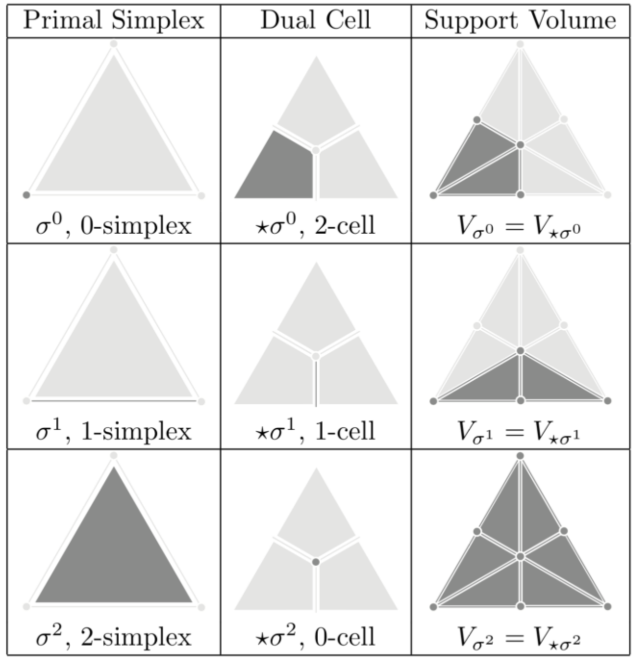
Advection Diffusion Multiphysics Model



de Rahm Complex

$$
\begin{array}{ccccccc}
\Omega_0 & \xrightarrow{d} & \Omega_1 & \xrightarrow{d} & \Omega_2 & \xrightarrow{d} & \Omega_3 \\
\star\downarrow\uparrow\star^{-1} & & \star\downarrow\uparrow\star^{-1} & & \star\downarrow\uparrow\star^{-1} & & \star\downarrow\uparrow\star^{-1} \\
\tilde{\Omega}_n & \xleftarrow[\tilde{d}]{} & \tilde{\Omega}_{n-1} & \xleftarrow[\tilde{d}]{} & \tilde{\Omega}_{n-2} & \xleftarrow[\tilde{d}]{} & \tilde{\Omega}_{n-3}
\end{array}
$$

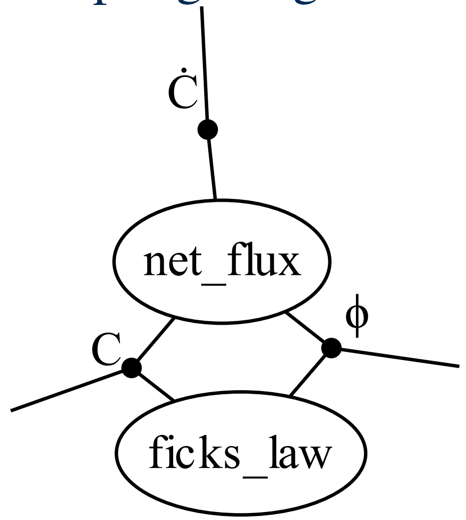
- Forms (cochains) over a manifold or simplicial complex
- Need to care about primal/dual for discretization
- Duality explains some properties of FEM



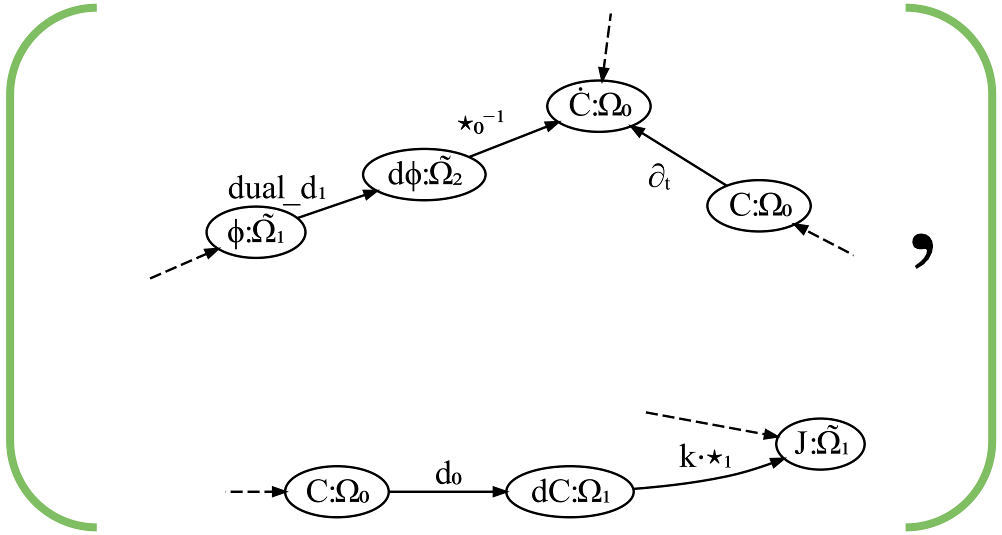
Multiphysics!

Graphical Language for Hierarchically Formulating Multiphysics Models

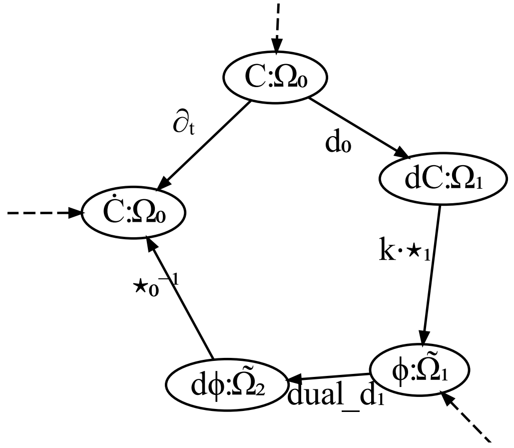
Coupling Diagram



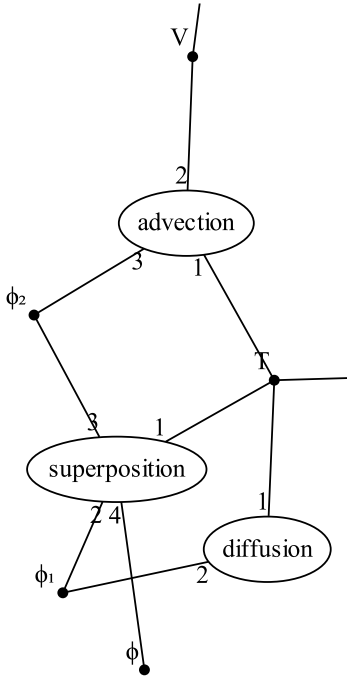
Component Laws



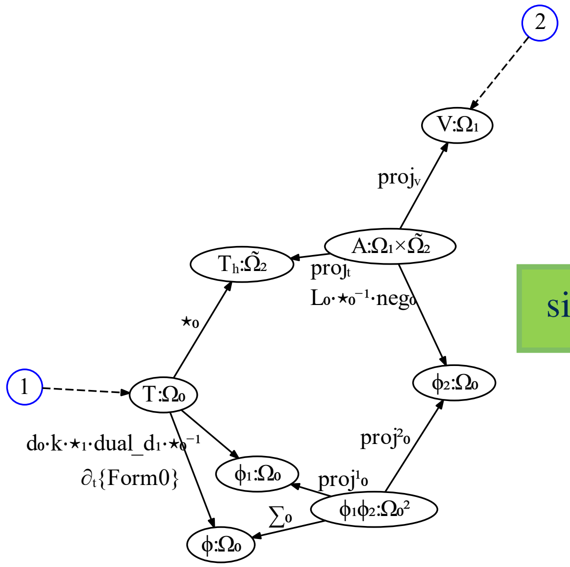
Composite Physics



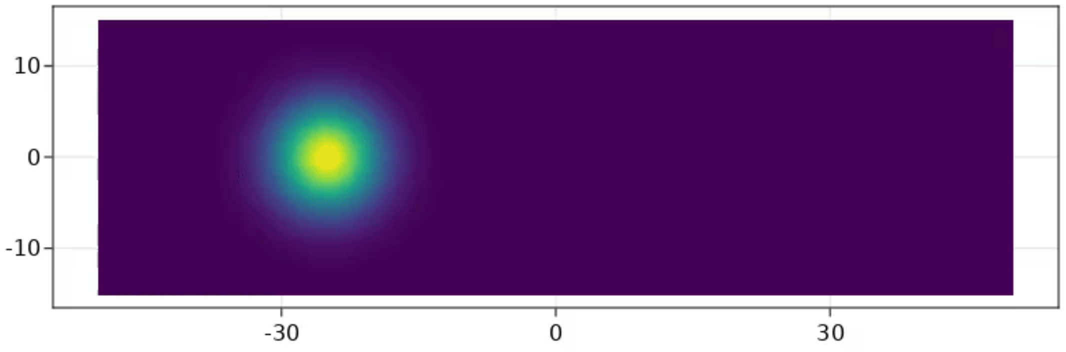
Automatic Simulator Compiler and Runtime



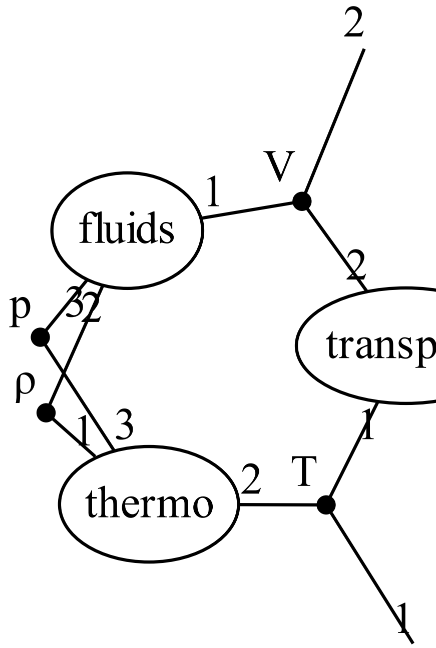
formulate

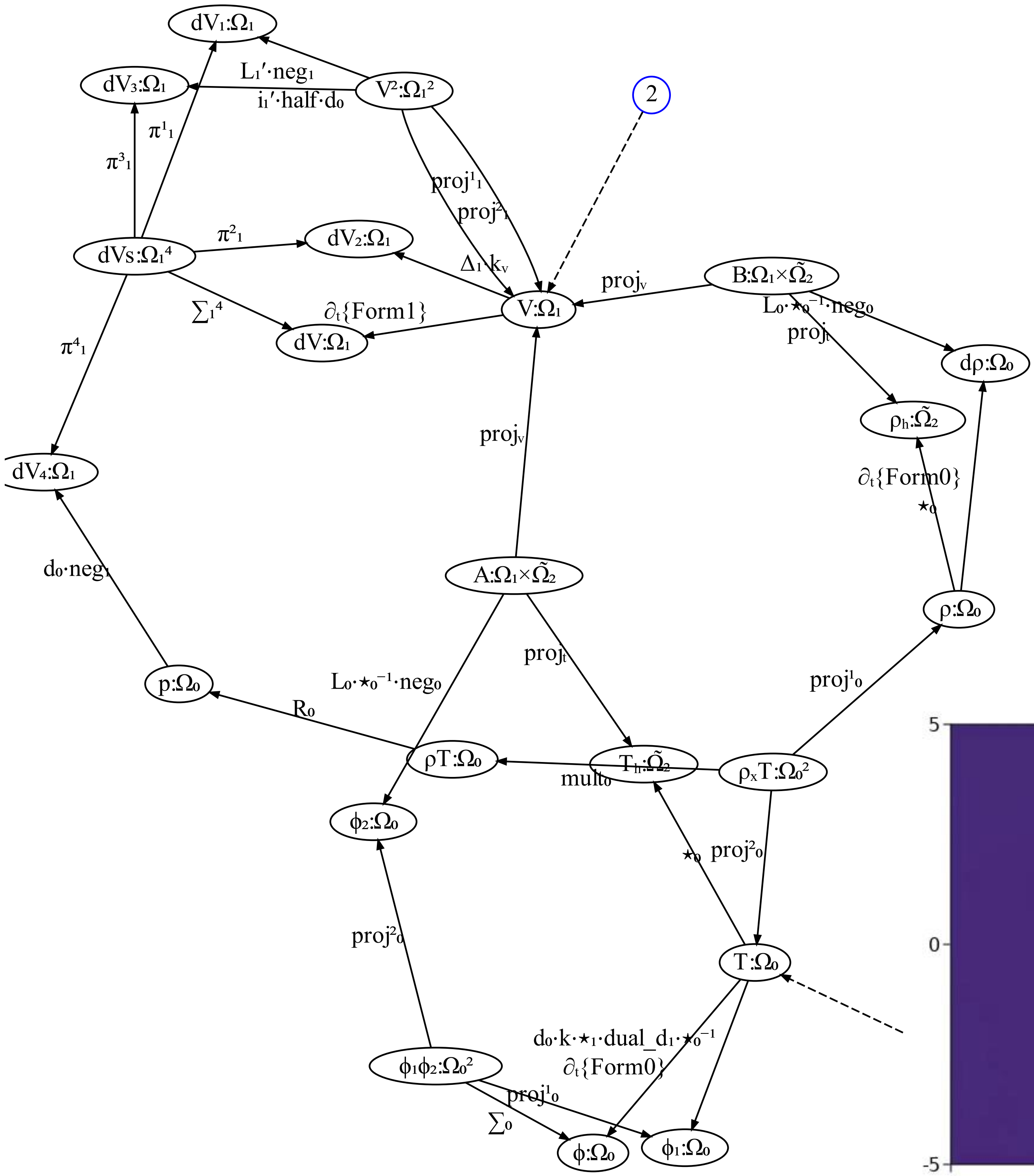


simulate

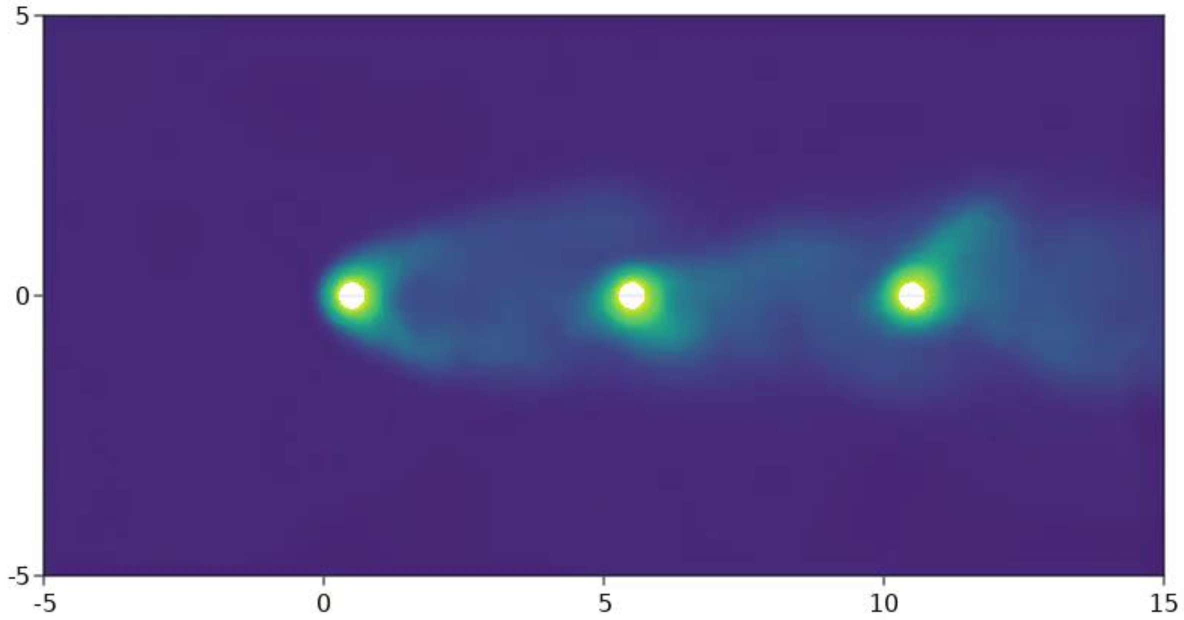


Conjugate Heat Transfer = Heat + Navier-Stokes





We have the expressive power to formulate CHT



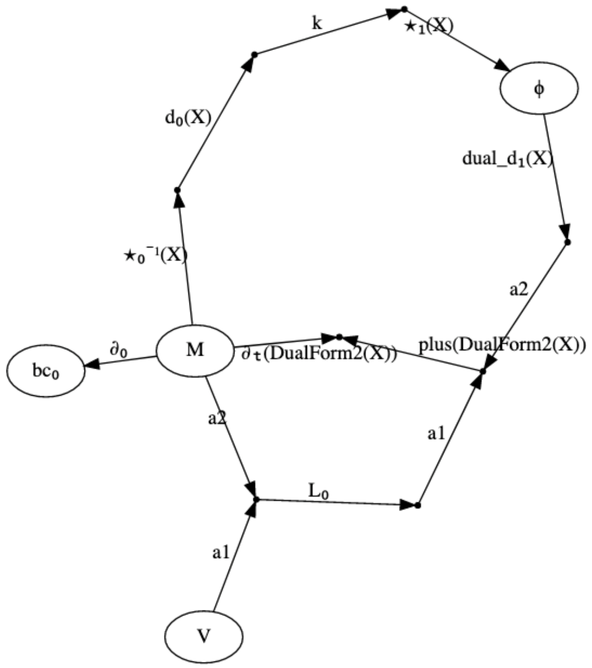
Graphical Equations to Simulation Pipeline

DEC Equations

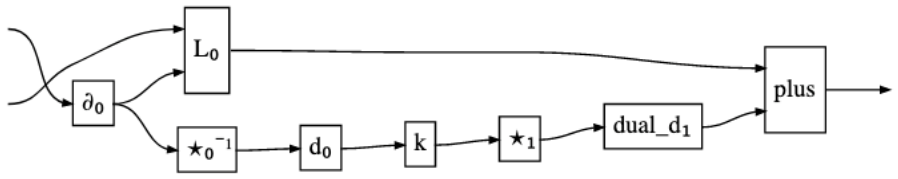
```
@present Flow2DQuantities(FreeExtCalc2D) begin
  X::Space
  M::Hom(munit(), DualForm2(X))  # Mass per dual-cell
  dM::Hom(munit(), DualForm2(X)) # change in mass
  V::Hom(munit(), Form1(X))      # Flow field
  ϕ::Hom(munit(), DualForm1(X))  # negative diffusion flux
  k::Hom(Form1(X), Form1(X))     # diffusivity (usually scalar multiplication)
  L₀::Hom(Form1(X)⊗DualForm2(X), DualForm2(X))
  ∂₀::Hom(DualForm2(X), DualForm2(X))
  bc₀::Hom(munit(), DualForm2(X))
end

@present Adv2D <: Flow2DQuantities begin
  # Fick's first law
  ϕ == M ⋅ ⋆₀⁻¹(X) ⋅ d₀(X) ⋅ k ⋅ ⋆₁(X)
  # Diffusion/advection equation
  M ⋅ ∂ₜ(DualForm2(X)) == (V⊗M) ⋅ L₀ + ϕ ⋅ dual_d₁(X)
  # Boundary condition
  M ⋅ ∂₀ == bc₀
end
```

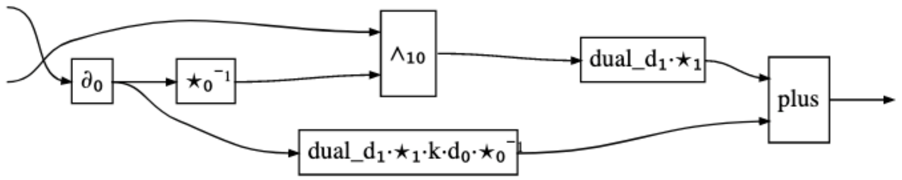
Graphical Equations



Computation Graph



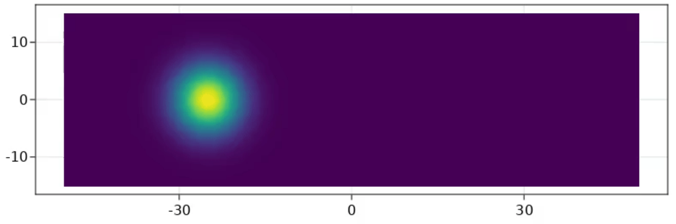
Optimized Computation Graph



Julia Code

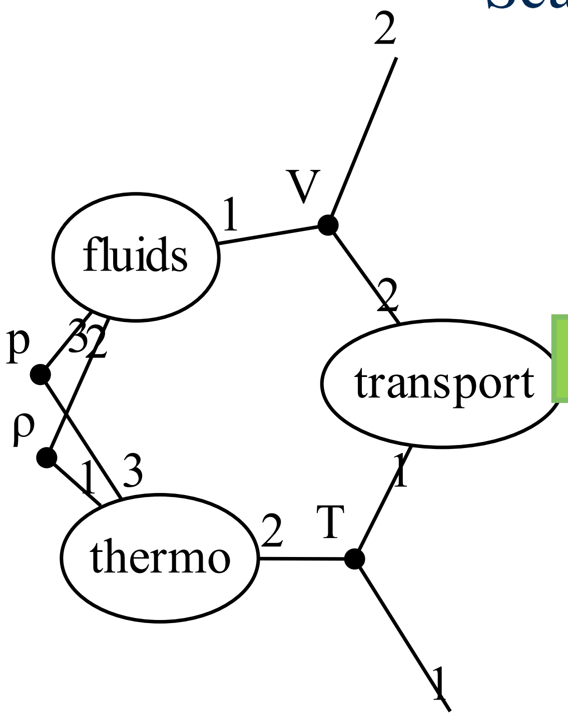
```
:(var"##745"(du, u, p, t, mem, matrices, funcs) = begin
  begin
      (var"#83#84"())(mem[6], u[1:4835])
      mem[2] .= matrices[2] * mem[6]
      (Decapods.Examples.var"#2#6")(mem[3], u[4836:19077], mem[2])
      mem[4] .= matrices[13] * mem[3]
      mem[1] .= matrices[12] * mem[6]
      begin
          for i = eachindex(mem[4])
              (mem[5])[i] = (funcs[2])((mem[4])[i], (mem[1])[i])
          end
      end
      du[1:4835] .= mem[5]
  end
end)
```

Simulation Results

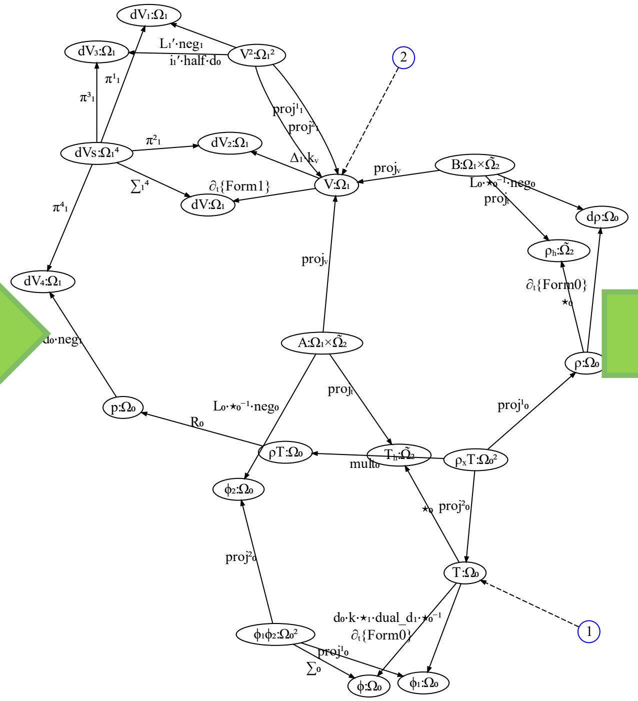


Comparing to SU2

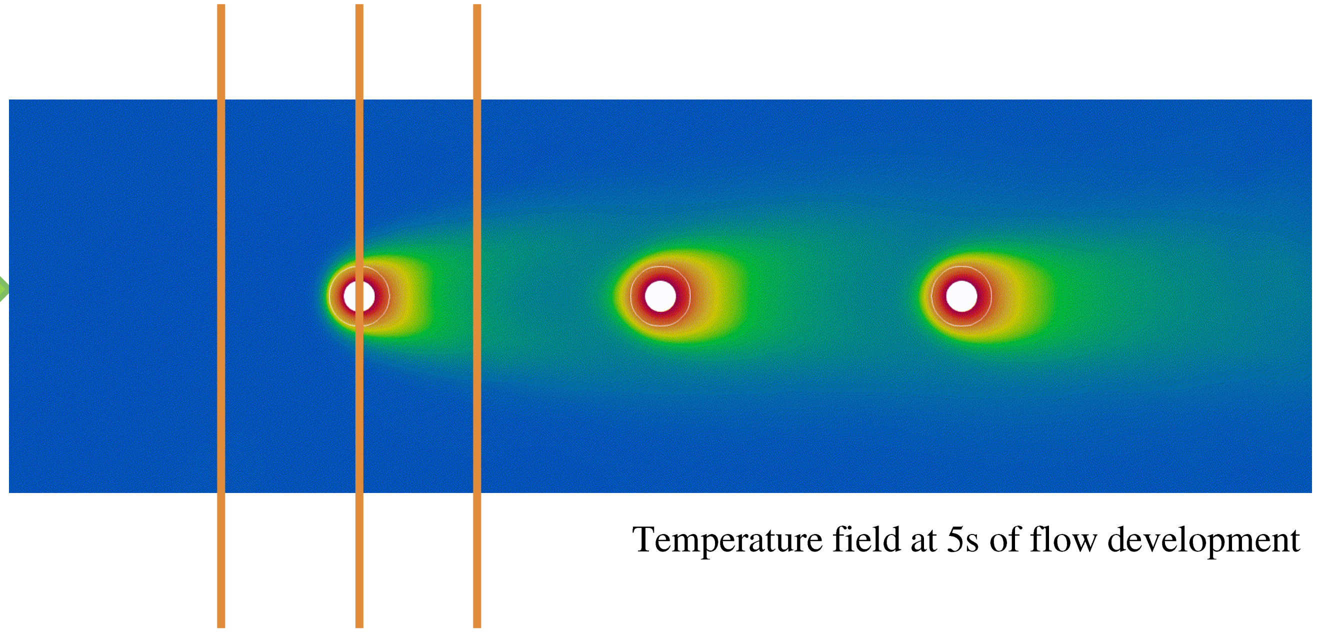
Scaling up to Conjugate Heat Transfer with Navier Stokes of Viscous Fluids



formulate

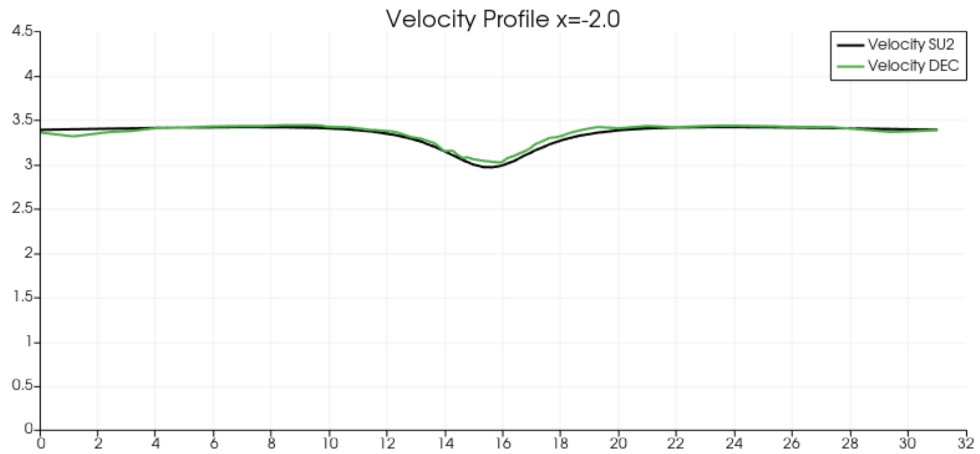


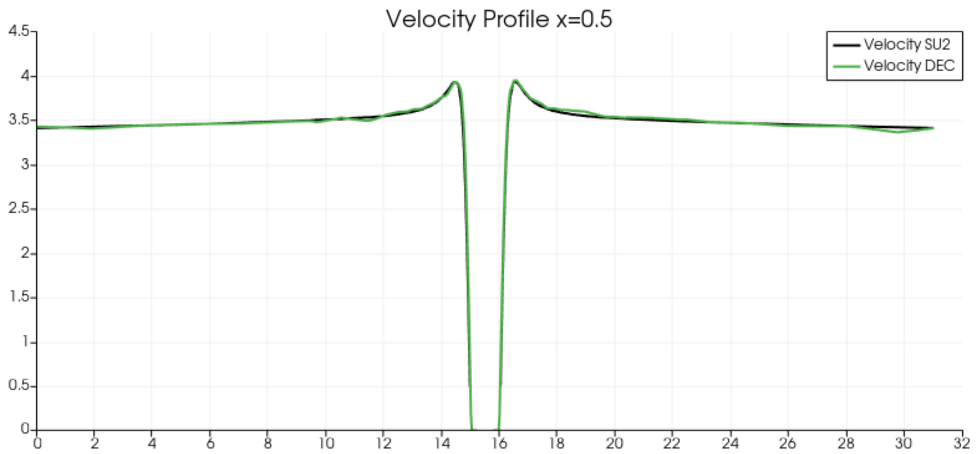
simulate

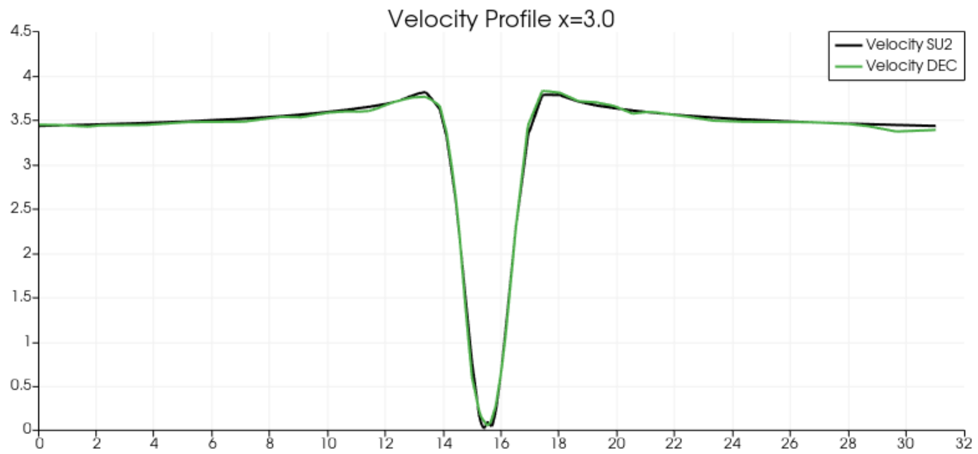


Temperature field at 5s of flow development

Velocity Profiles

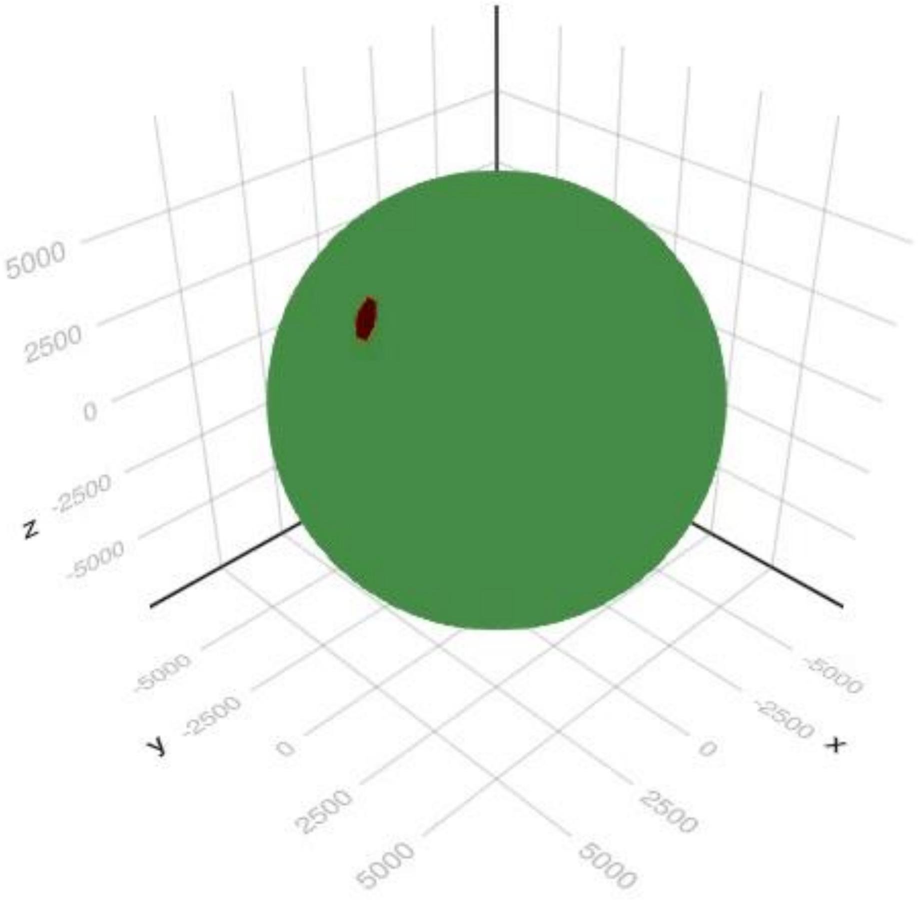




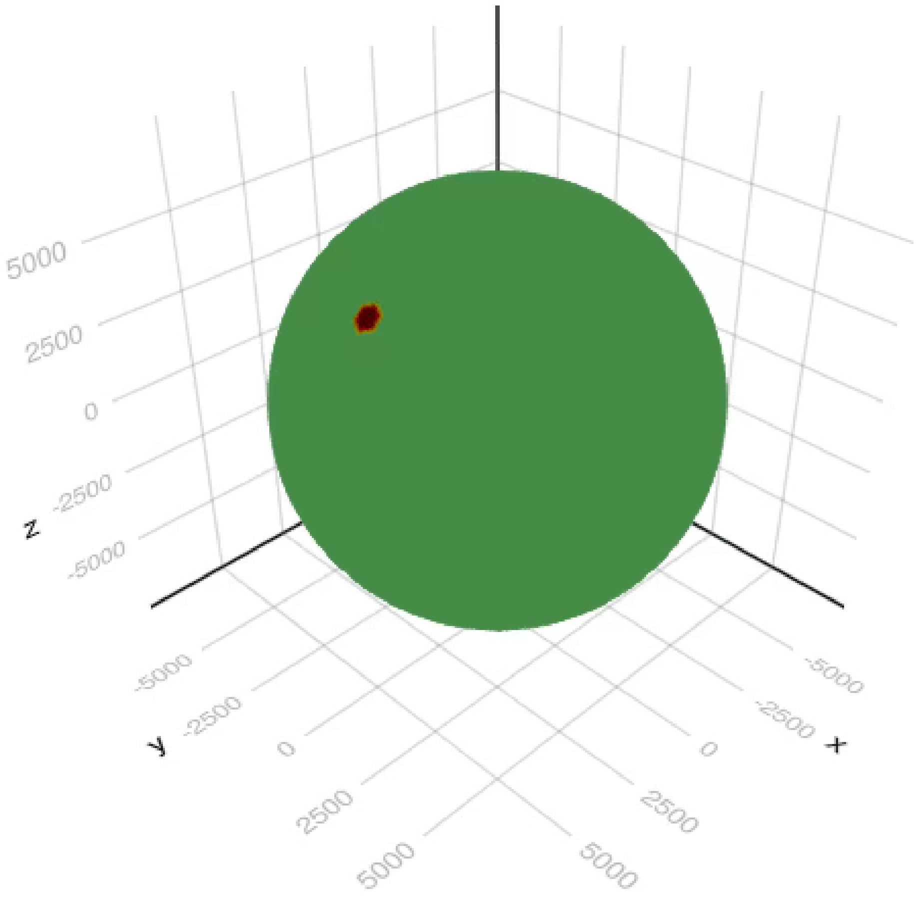


Software Decoupling Improves V&V

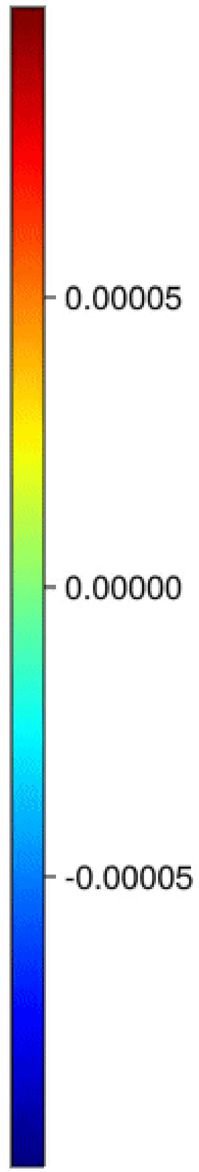
Meshing is decoupled from formulation so you can easily compare implementations



Navier Stokes on UV-Sphere

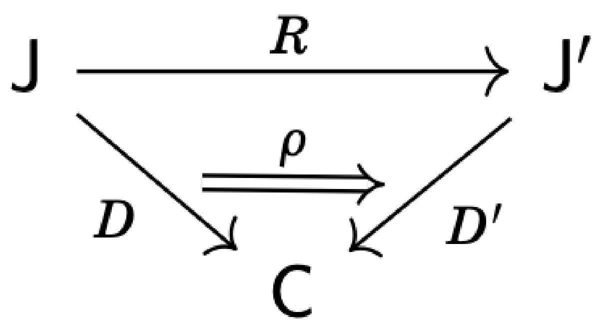


Navier Stokes on Icosphere

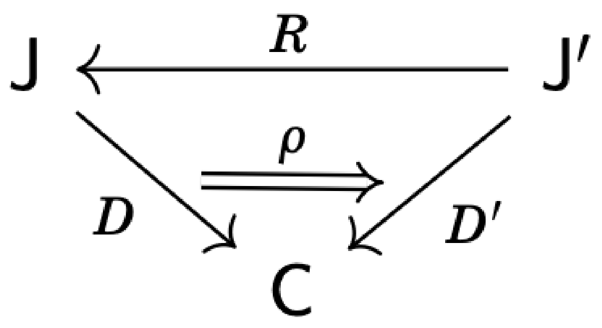


Morphisms of Diagrams

Forward Direction



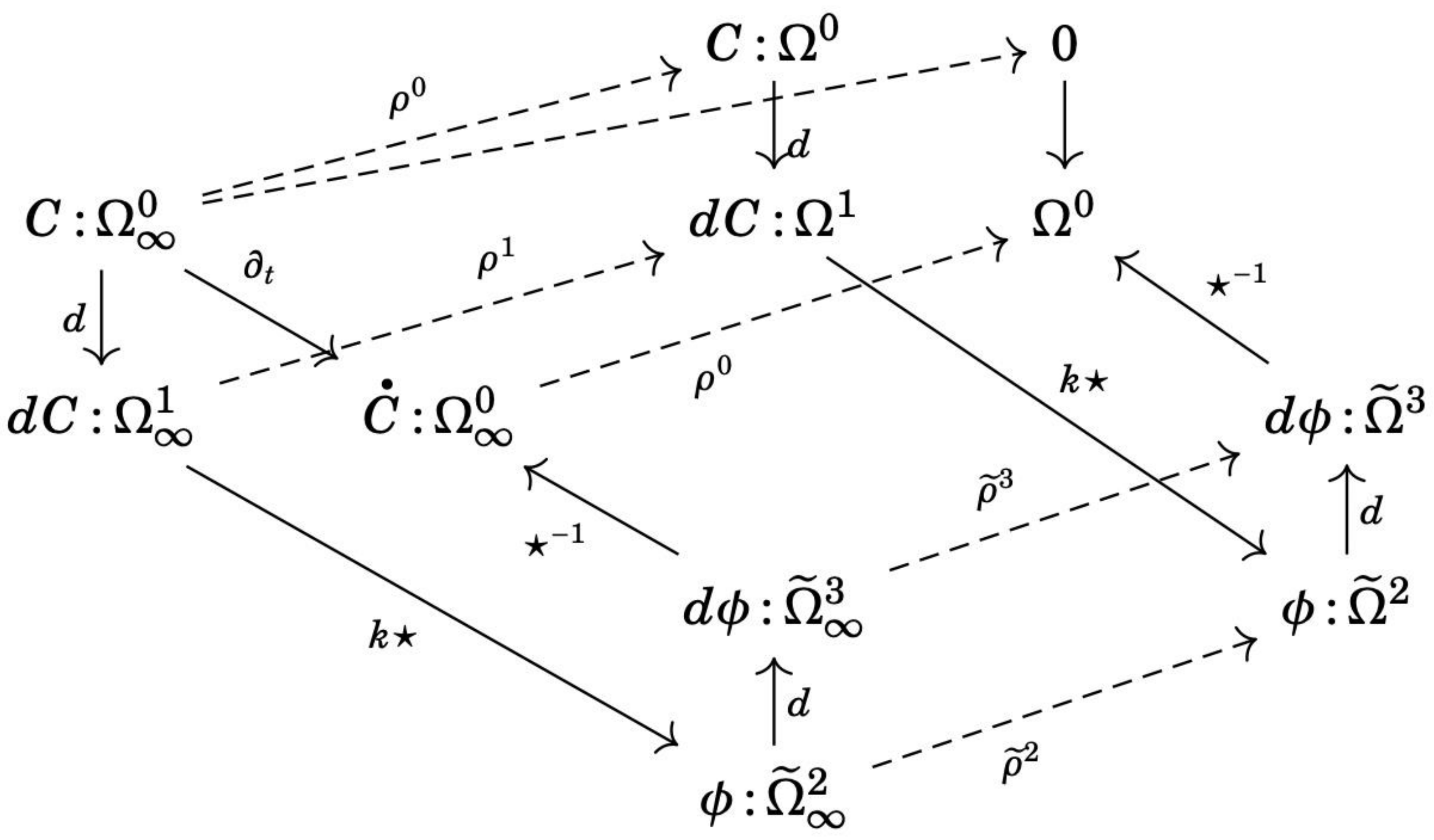
Reverse Direction



- When do two equations specify *the same physics*?
- Maps between diagrams encode relationships between physical theories
- Functors between shapes relate structural similarity
- *Natural transformations* relate data
- **R** specifies relationships between syntactic variables and operators
- $\boldsymbol{\rho}$ specifies the relationships between the numerical data

Steady States of a System Expressed Diagrammatically

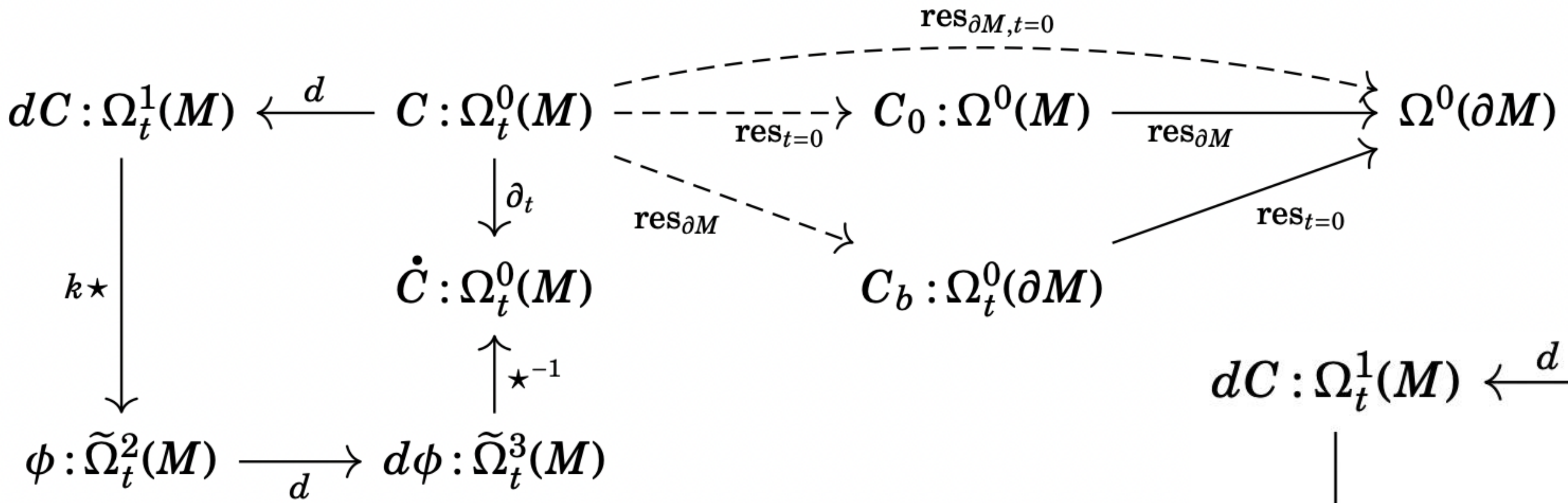
A morphism (R, ρ) from the first diagram to the second can be depicted as



- Maps between diagrams encode relationships between physical theories
- ie. the steady state heat equation is the limit of the dynamic heat eqn as time goes to infinity

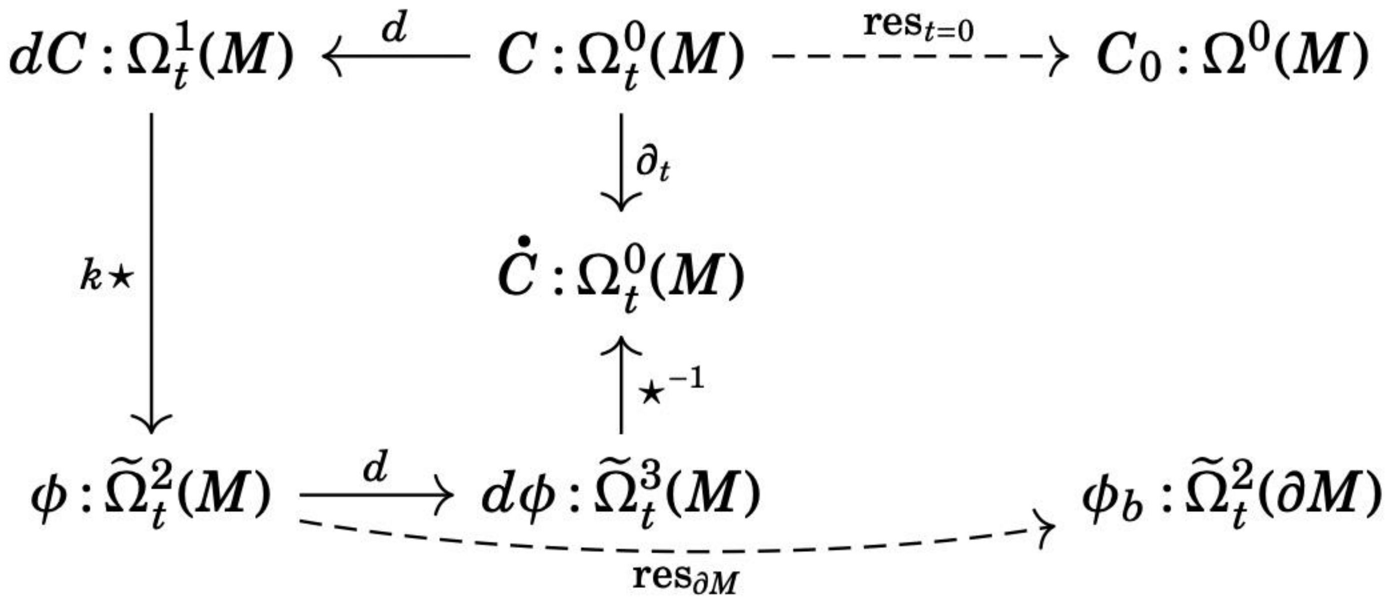
Note: The “3D” layout here is for illustrative purposes.

Boundary Conditions



Neumann

Dirichlet



- Morphisms between manifolds (like the boundary operator) induce relationships between diagrams.
- Can enforce boundary and initial conditions in the same framework

Conclusions

- Applying Category Theory to study abstractions in mathematics leads to better software
- ACT software can address physics and mechanical engineering problems
- We can push out the Pareto frontier of usability, generality, and performance in scientific software

Publications and Software

1. Operadic Modeling of Dynamical Systems: Mathematics and Computation, S. Libkind, A. Baas, E. Patterson, J. Fairbanks, ACT 2021 Proceedings
2. Categorical Data Structures for Technical Computing, E. Patterson, O. Lynch, J. Fairbanks, *Compositionality*, Accepted Feb 2022
3. An Algebraic Framework for Structured Epidemic Modeling, S. Libkind, A. Baas, M. Halter, E. Patterson, and J. Fairbanks, accepted Mar 2022 with *The Royal Society Phil. Trans. A*
4. Computational category-theoretic rewriting, K. Brown, T. Hanks, E. Patterson, J. Fairbanks, accepted to *International Conference on Graph Transformation*, July 2022
5. A Diagrammatic View of Differential Equations in Physics, E. Patterson, A. Baas, T. Hosgood, J. Fairbanks, accepted May 2022 to *Mathematics in Engineering*
6. Compositional Exploration of Combinatorial Scientific Models. K. Brown, T. Hanks, and J. Fairbanks, Submitted May 2022 *Proceedings of the Conference on Applied Category Theory*
7. Diagrammatic Equations for Multiphysics Modeling and Simulation, A. Baas, K. Brown, J. Arias, M. Gaitlin, E. Patterson, J. Fairbanks in preparation

Package	Description
Catlab.jl	A framework for applied category theory in the Julia language
Semagrams.jl	A framework for developing GUIs for studying presheaf categories. Used to build HMIs for modeling tools.
AlgebraicPetri.jl	Build and analyze petri net based models compositionally. Supports both ODE and Stochastic execution
Individuals.jl	Agent Based Modeling with Rewriting Rules
AlgebraicDynamics.jl	Build and Simulate dynamical systems compositionally
StockFlow.jl (funded by AFOSR)	A Systems Dynamics approach to modeling based on Stock and Flow Diagrams
CombinatorialSpaces.jl	Simplicial sets and other combinatorial models of geometric spaces. A space to build geometric and PDE simulators
Decapodes.jl	A PDE solver based on the Method of Lines for Discrete Exterior Calculus. Supports compositional description of multiphysics.
AlgebraicRelations.jl	Integration with Relational Database Technology. Including building SQL categorically
AlgebraicWorkflows.jl	A system for representing analysis workflows based on monoidal categories. Tracks scientific data processing pipelines with provenenance
Hydrologics.jl (prerelease)	Hydrology modeling with operads of river systems
AlgebraicNeurons.jl (prerelease)	Deep Learning with complex architectures constructed hierarchically

The Team


Dr. James Fairbanks


Dr. Kris Brown

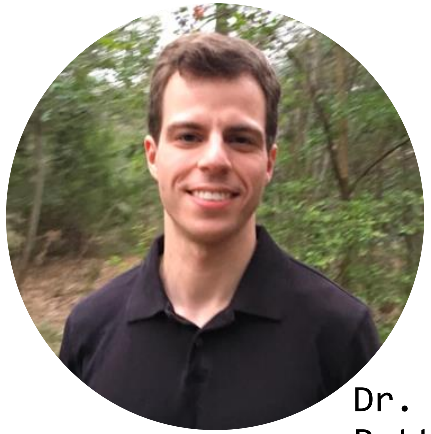

Tyler Hanks


Luke Morris


George Rauta


Sophie Libkind


Dr. Tim Hosgood


Dr. Evan Patterson


Owen Lynch


Andrew Baas

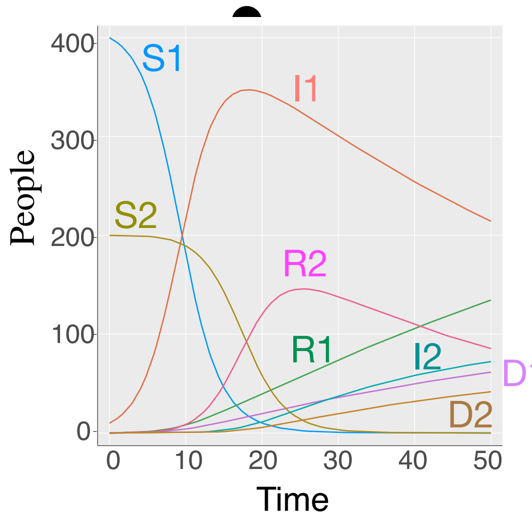

Maia Gatlin

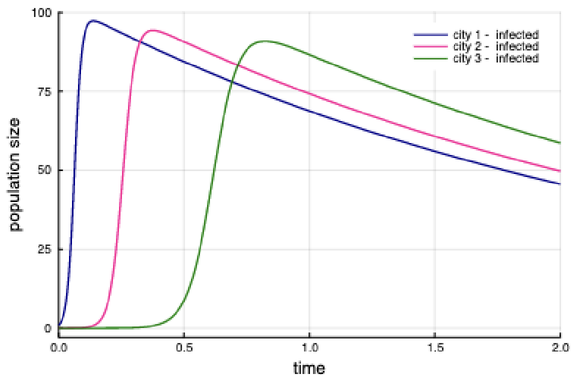

Jesus Arias


Dr. Clayton Kerce

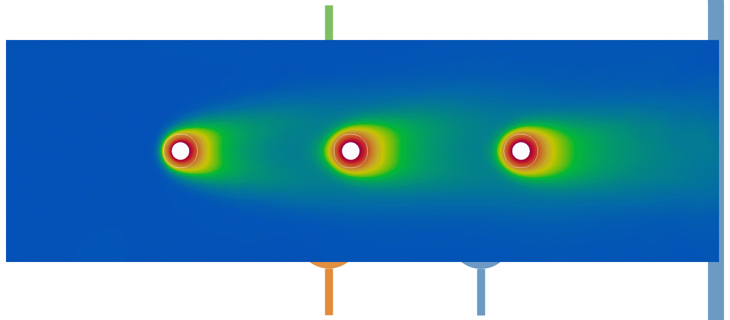
Mathematical Programming Languages Compile to ODES

Languages for
Hierarchical System
Descriptions





(a)



Mathematical Interpreters

Sharing
(Biology)

Signal Flow
(Electronics / Controls)

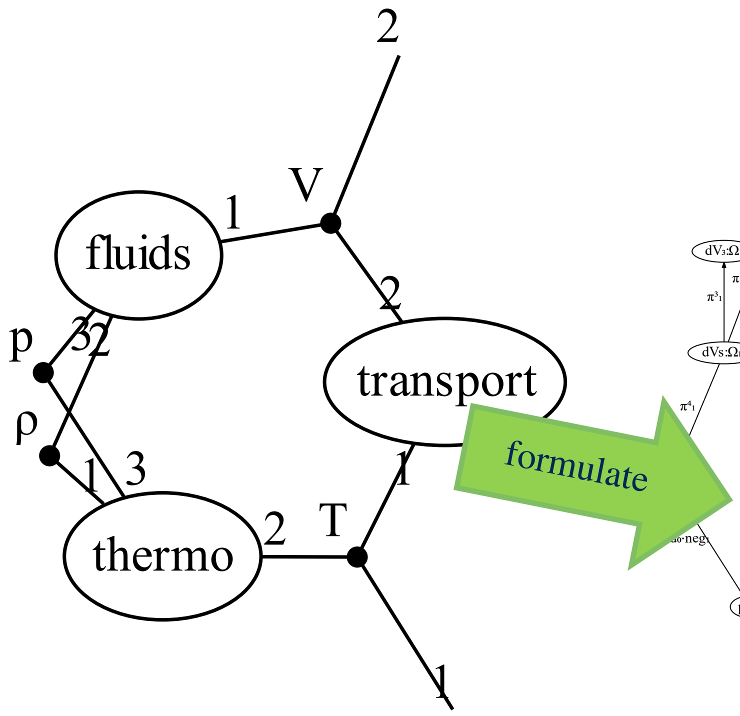
FEM/Stencils
(Physics)

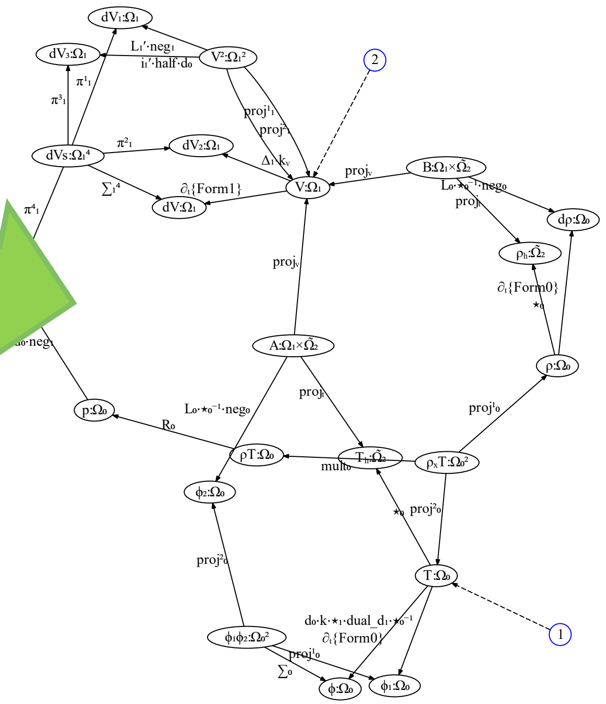
Common Dynamical Systems Semantics
Support simulation with existing solvers

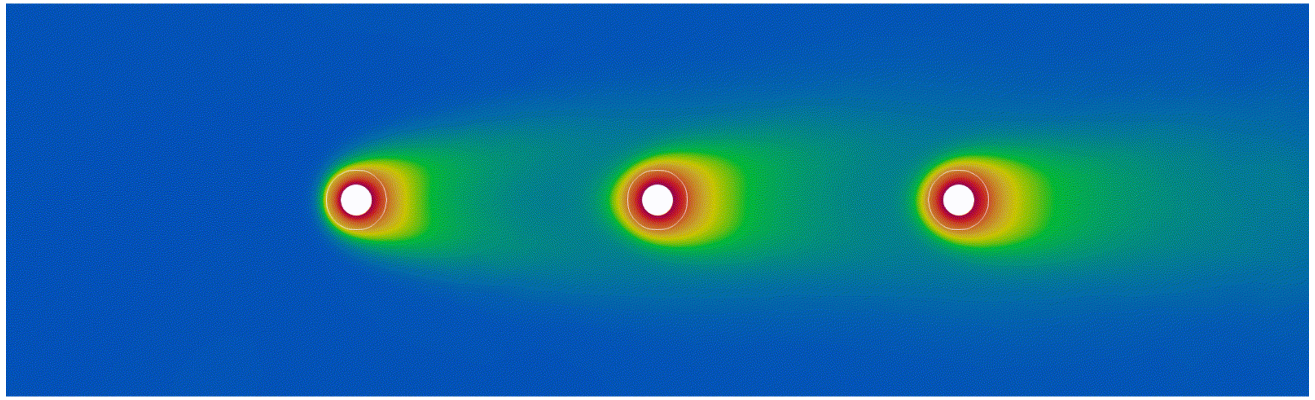
$$\begin{bmatrix}\dot{u}_1\\ \dot{u}_2\\ \dot{u}_3\end{bmatrix} = f\left(\begin{bmatrix}u_1\\ u_2\\ u_3\end{bmatrix}\right)$$

Operadic Modeling of Dynamical Systems: Mathematics and Computation, S. Libkind, A. Baas, E. Patterson, J. Fairbanks, *Applied Category Theory* Proceedings 2021

Level of Effort Breakdown

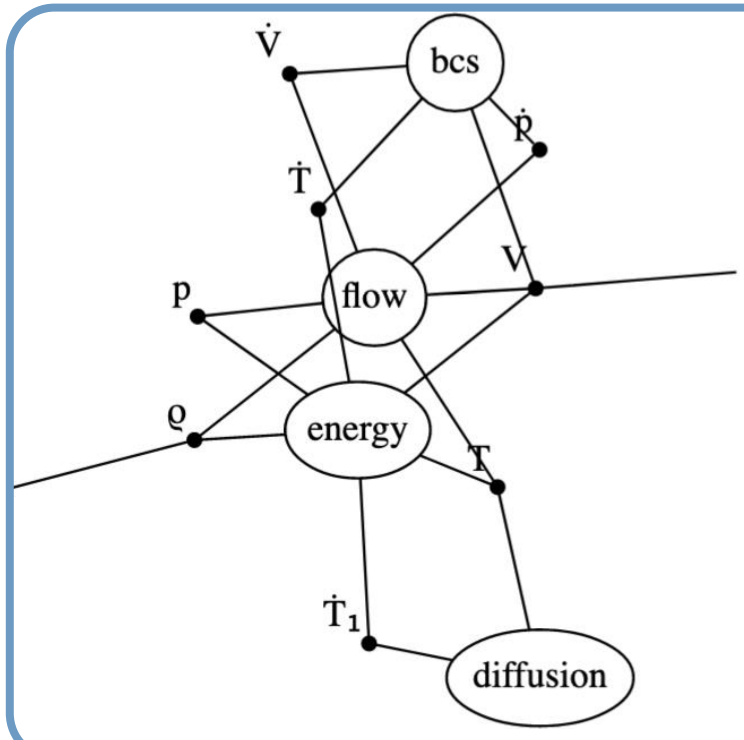




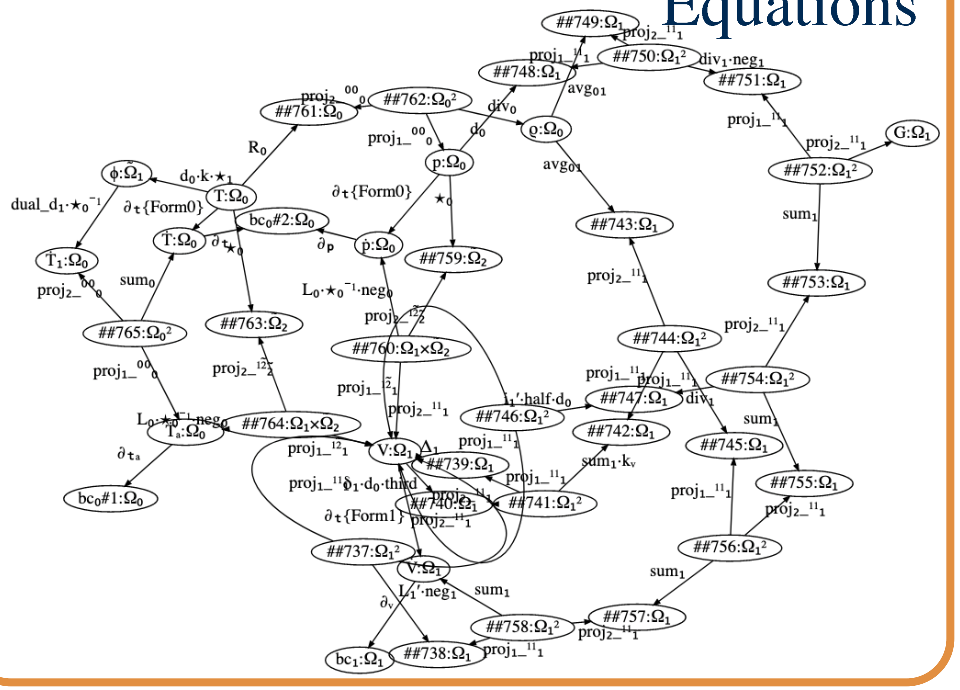


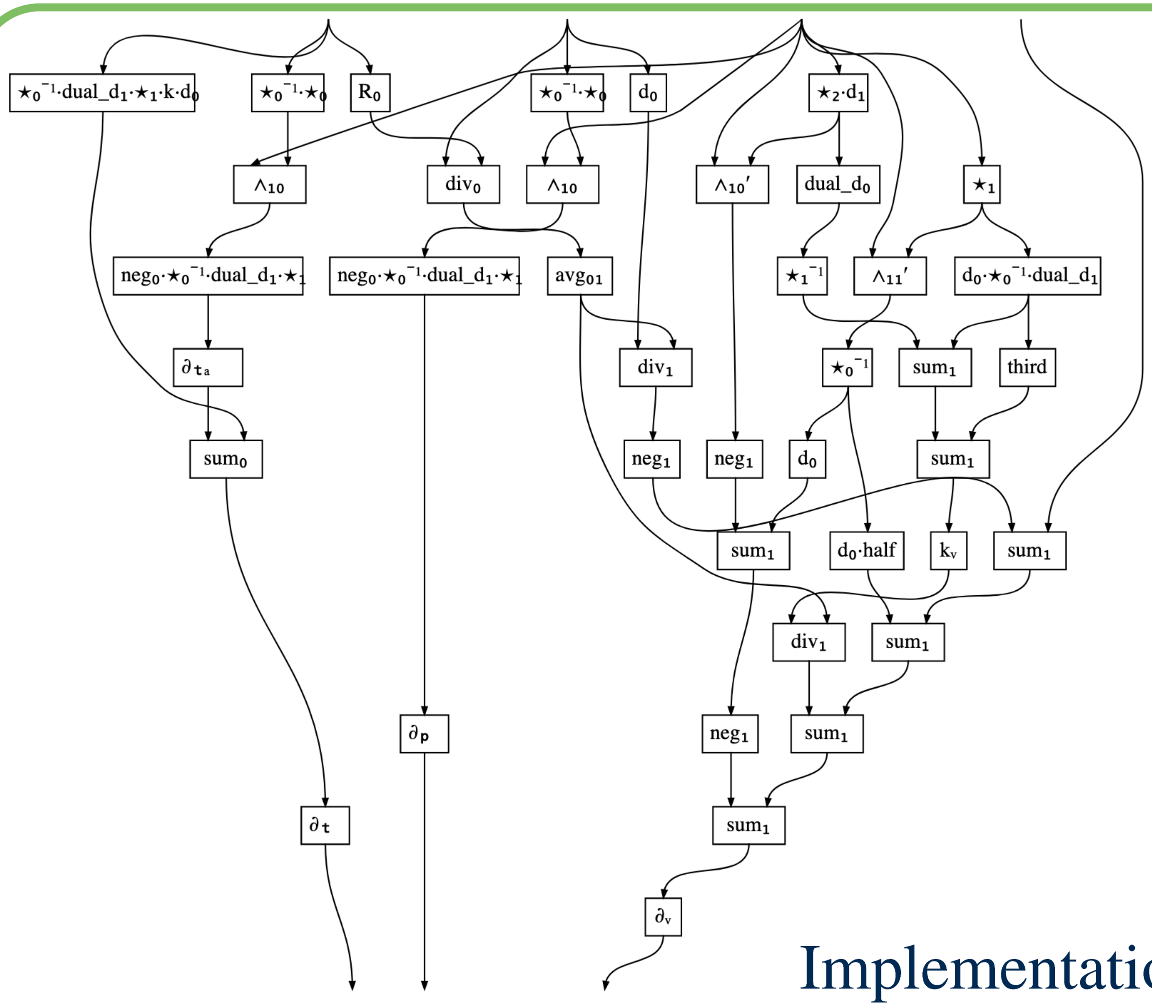
Code Stage	Lines of Code	%
Imports	26	10%
Define Physics	84	31%
Generating Method from Physics	9	3%
Mesh Loading and Boundary Conditions	89	33%
Initial Conditions	15	6%
Running Sim	5	2%
Results and Viz	41	15%

CHT Generated Diagrams

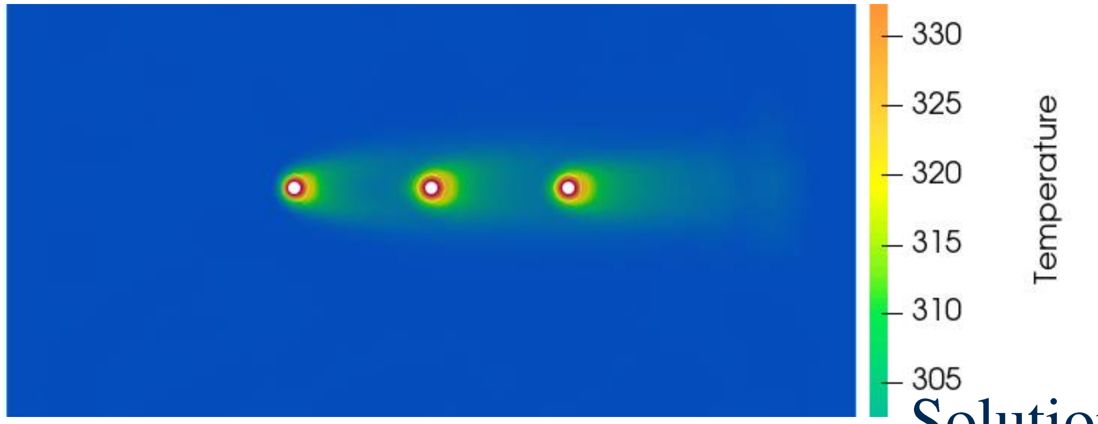


Theory


Equations



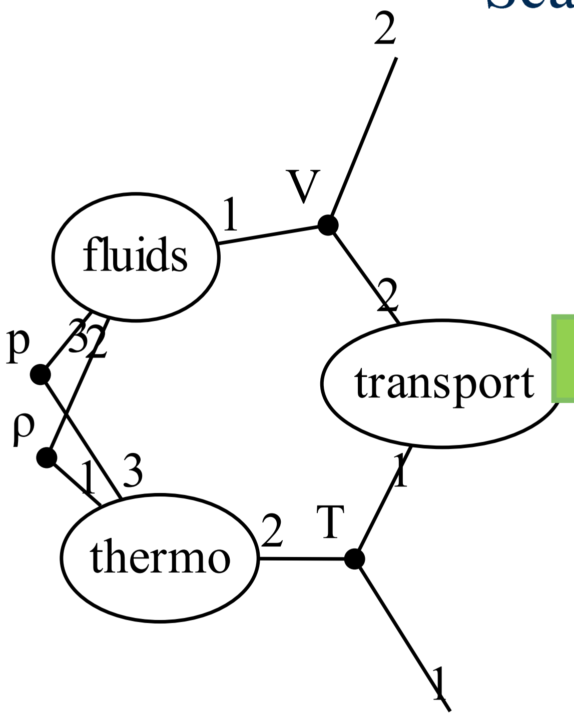
Implementation



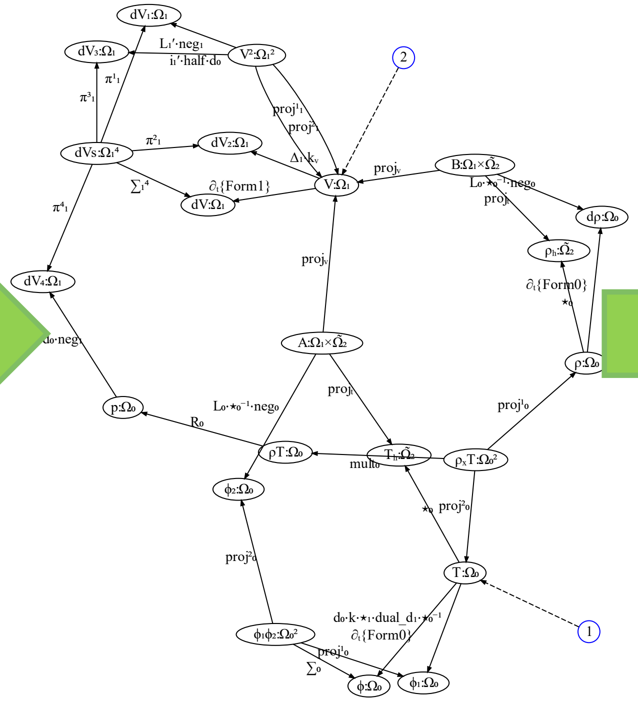
Solution

Comparing to SU2

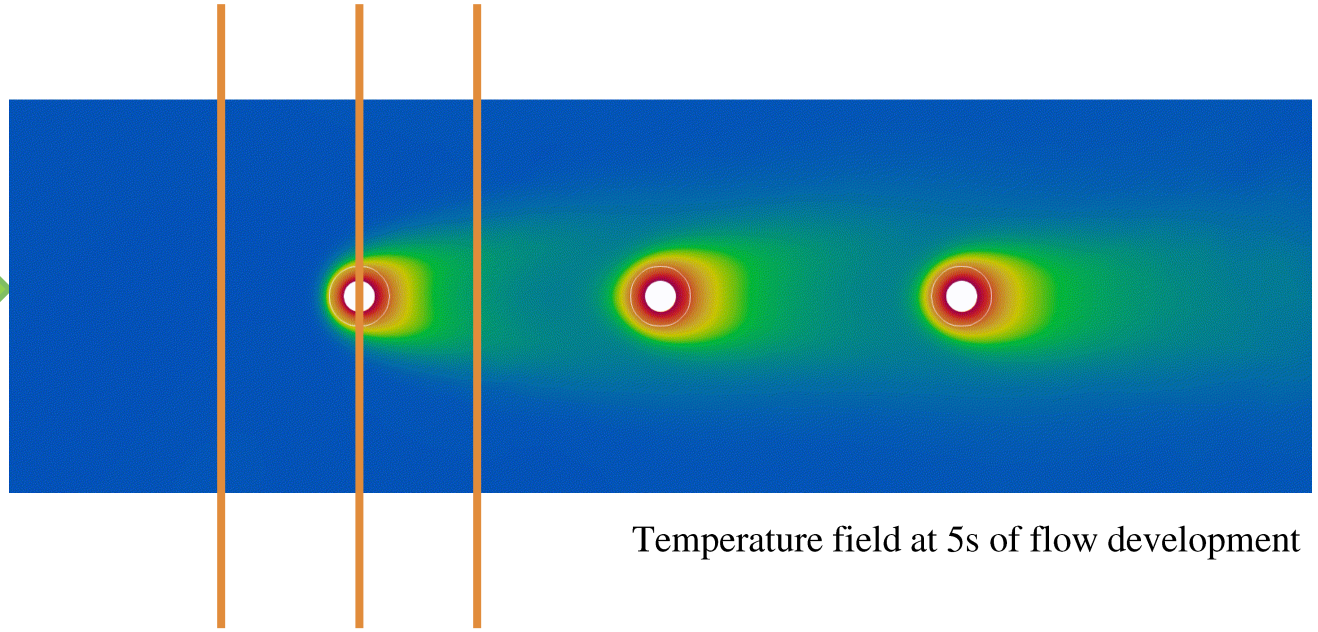
Scaling up to Conjugate Heat Transfer with Navier Stokes of Viscous Fluids



formulate

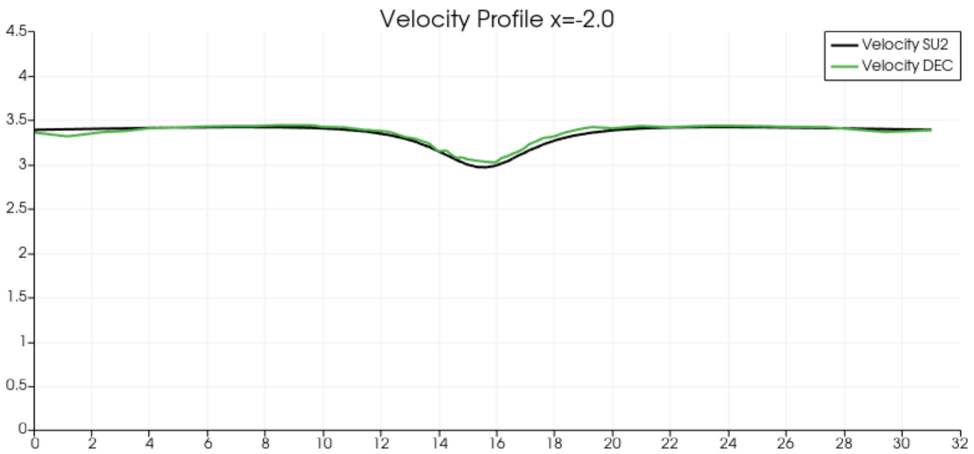


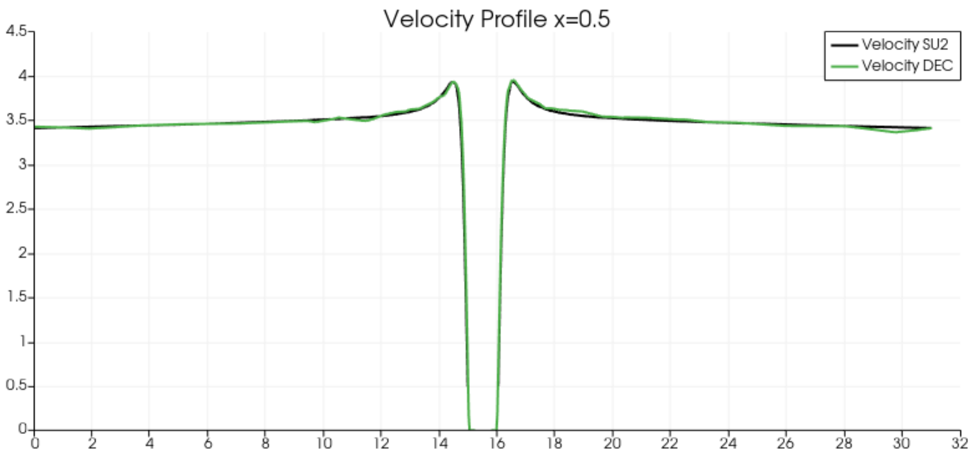
simulate

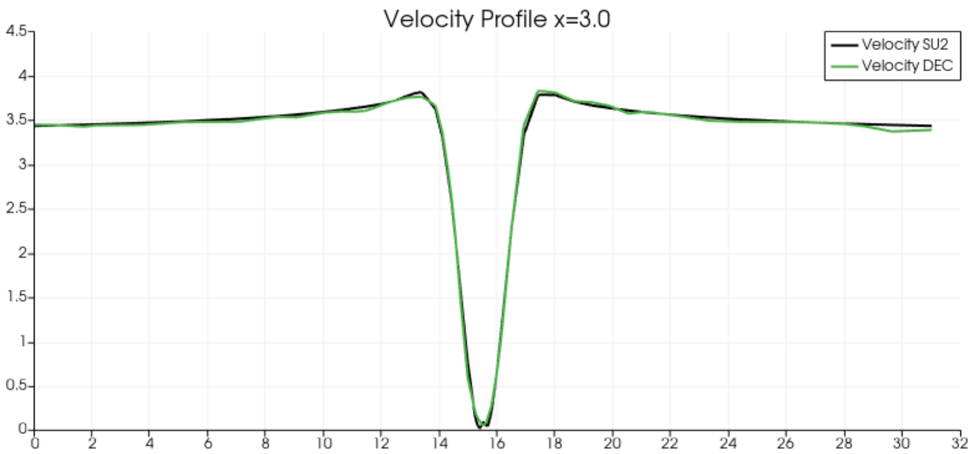


Temperature field at 5s of flow development

Velocity Profiles

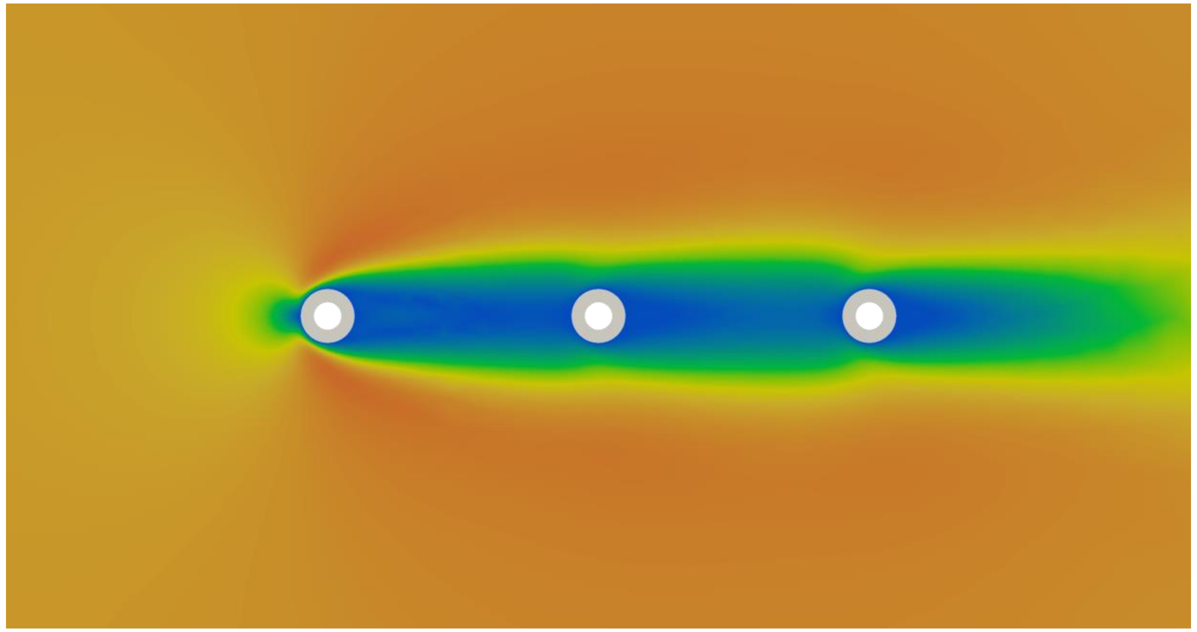


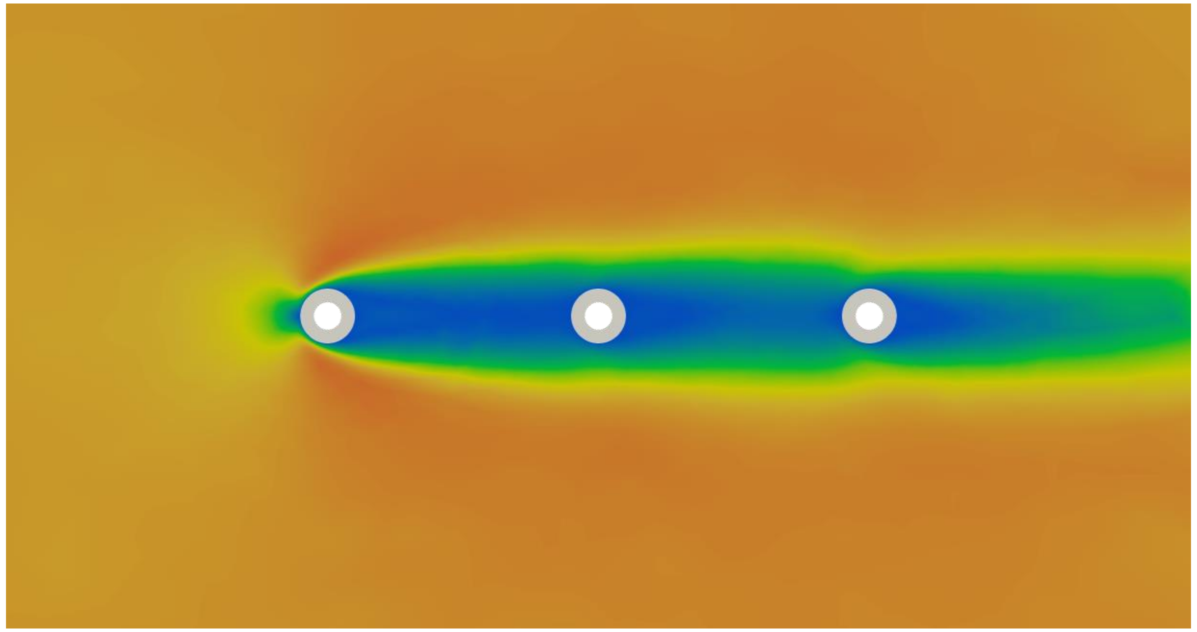




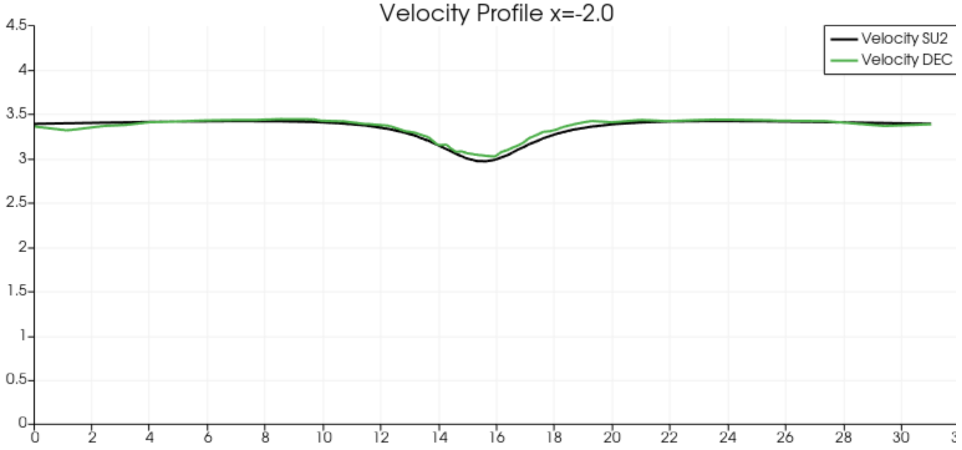
Comparing to SU2

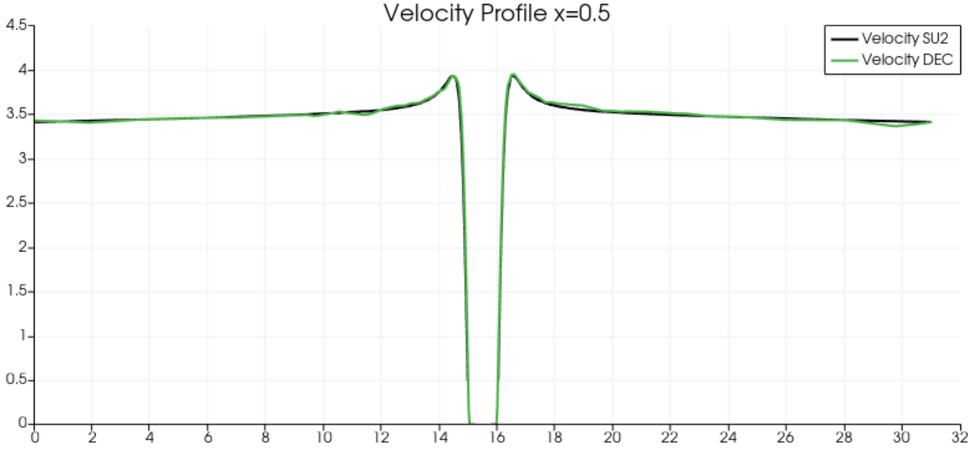
Velocity Field

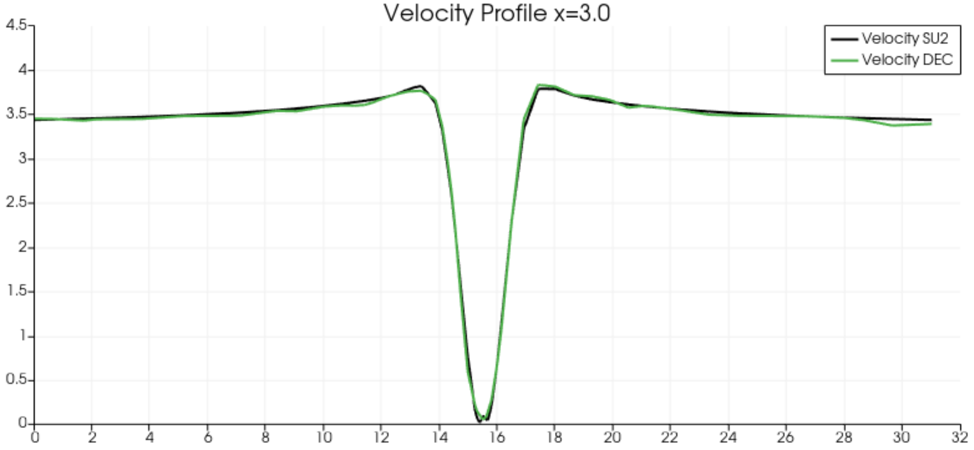




Velocity Profiles

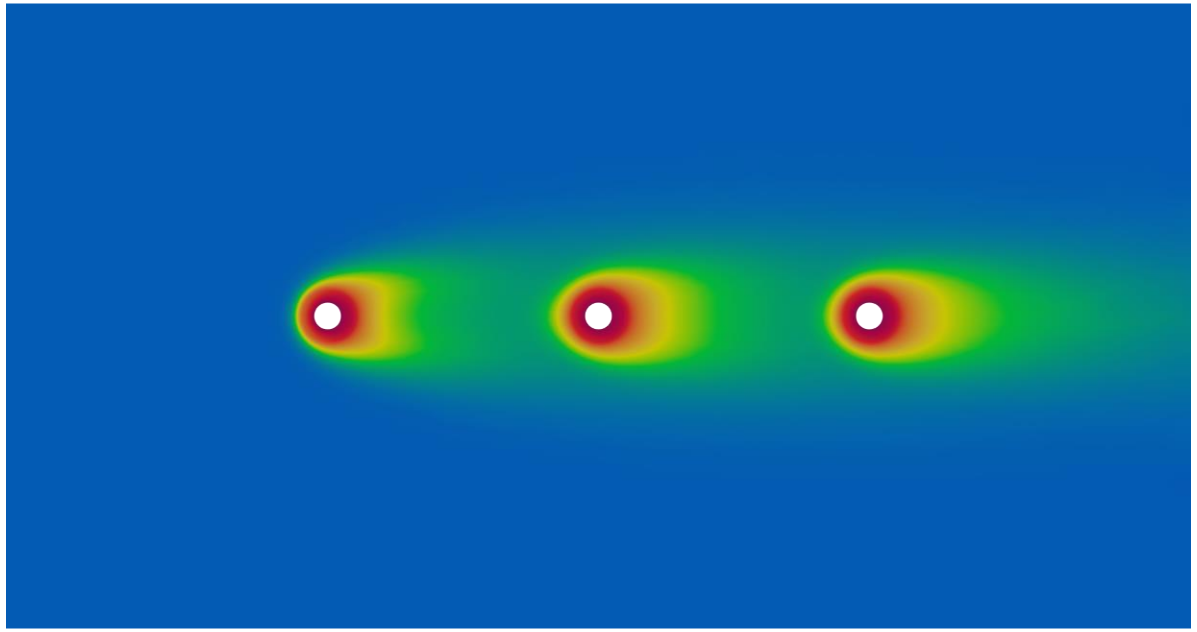


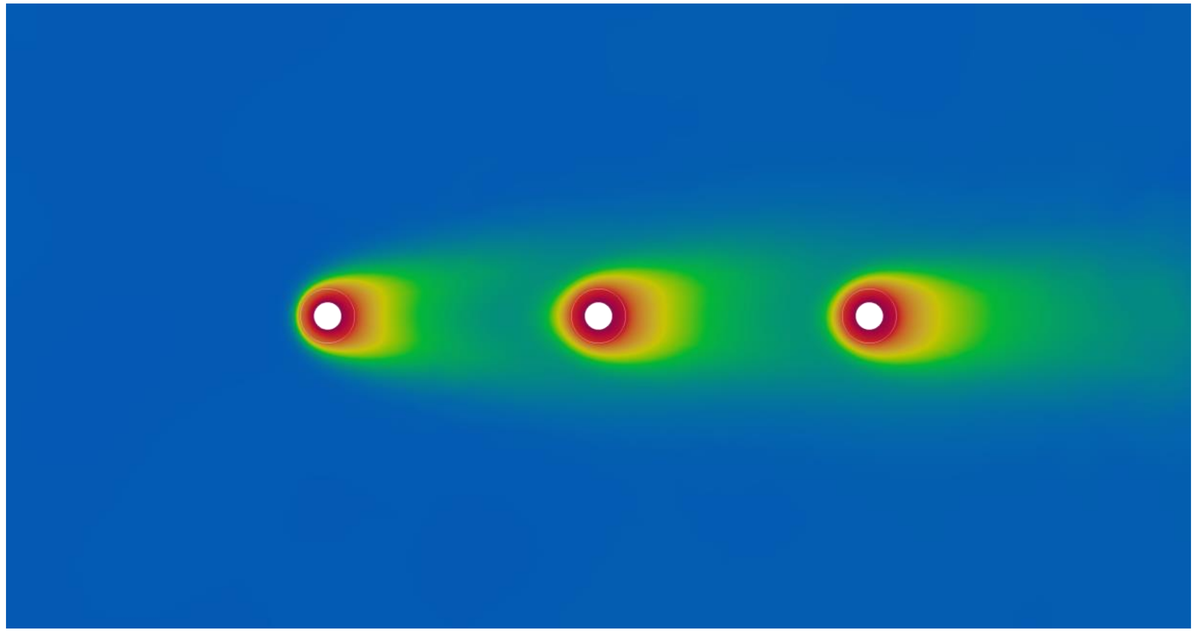




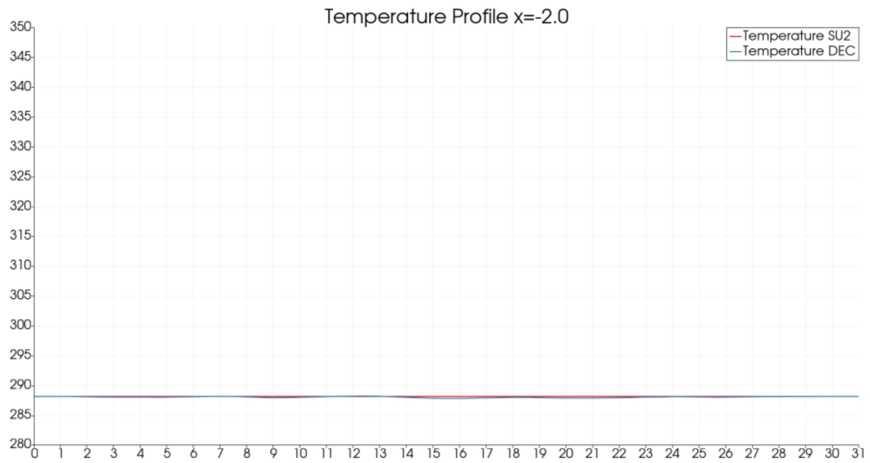
Comparing to SU2

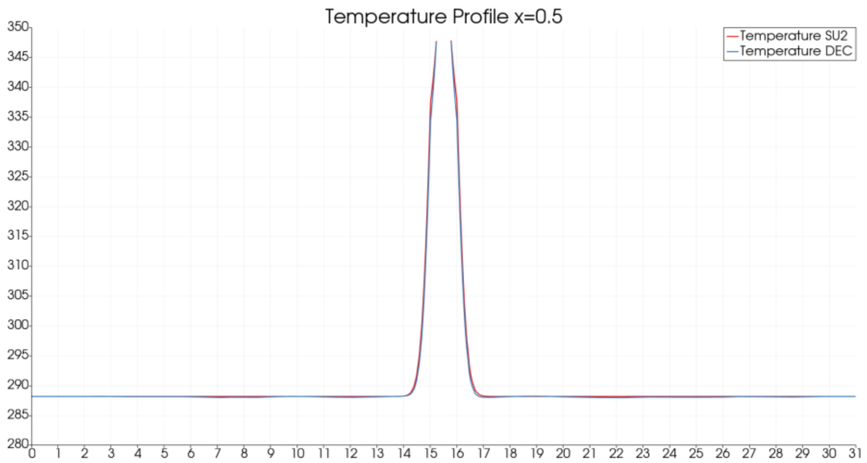
Temperature Field

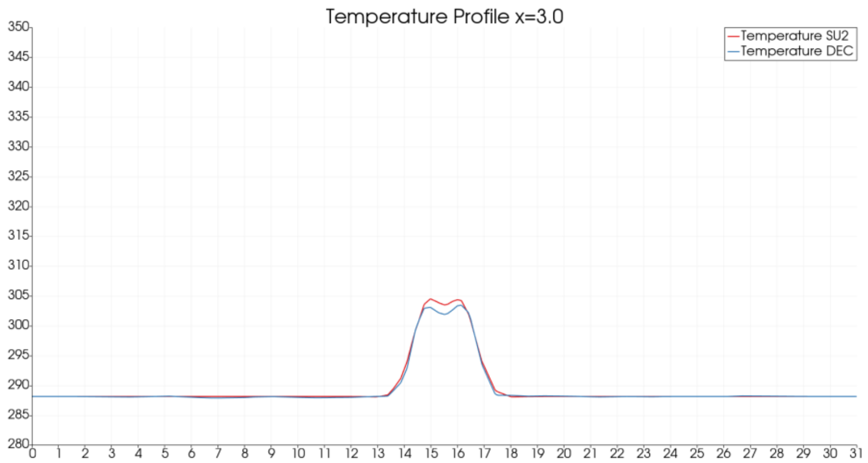




Temperature Profiles



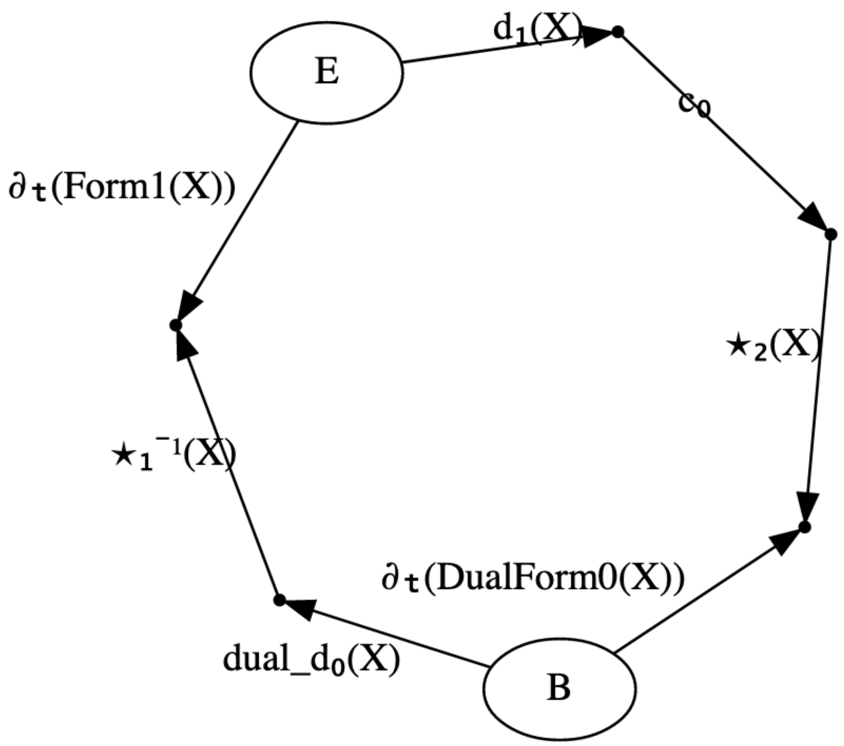


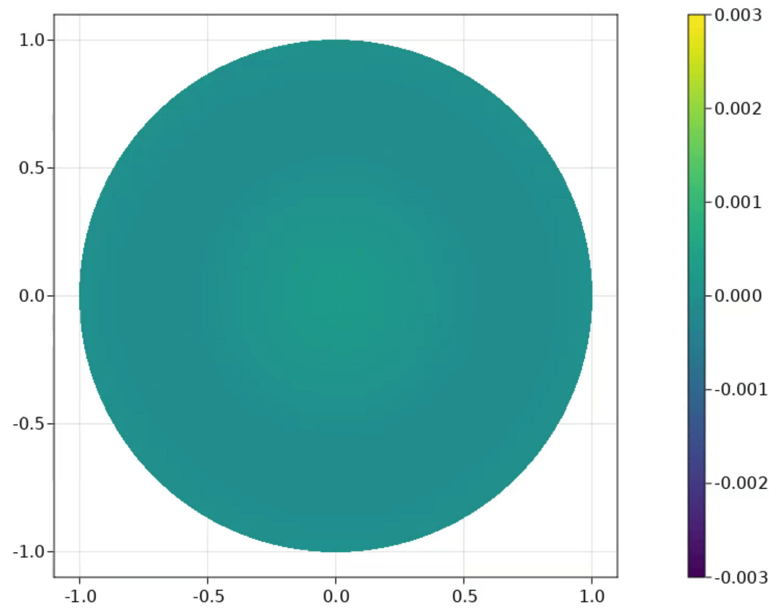


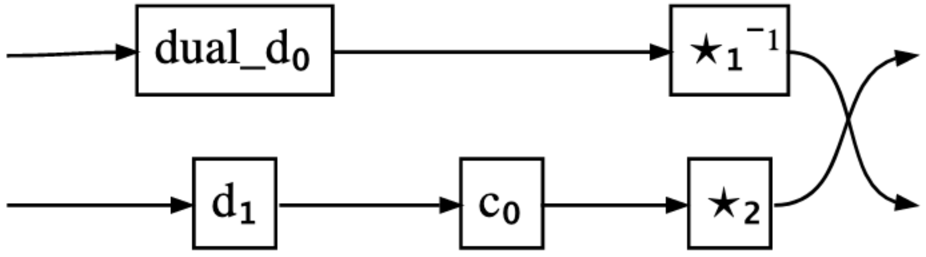
Maxwell's

```
# Define used quantities
@present EM2DQuantities(FreeExtCalc2D) begin
    X::Space
    E::Hom(munit(), Form1(X))       # electric field
    B::Hom(munit(), DualForm0(X))      # magnetic field
    c₀::Hom(Form2(X), Form2(X))    # 1/(μ₀ ϵ₀) (scalar)
end

# Define Electromagnetic physics
@present EM2D <: EM2DQuantities begin
    B ⋅ ∂ₜ(DualForm0(X)) == E ⋅ d₁(X) ⋅ c₀ ⋅ ⋆₂(X)
    E ⋅ ∂ₜ(Form1(X)) == B ⋅ dual_d₀(X) ⋅ ⋆₁⁻¹(X)
end
```







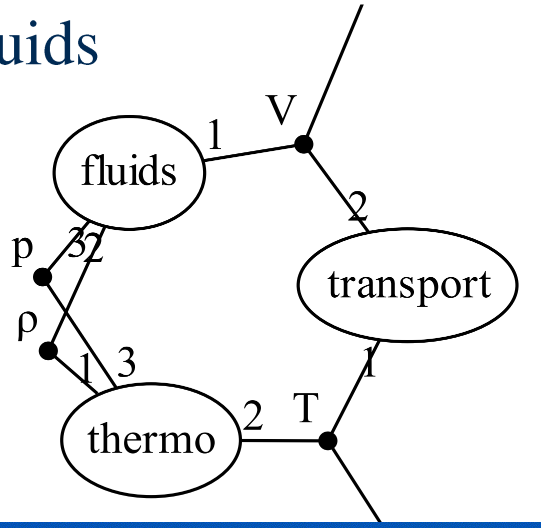
Conclusions

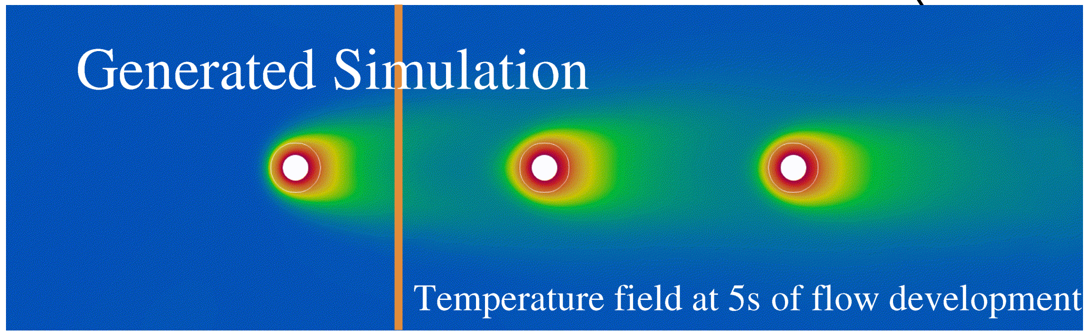
Diagrammatic Equations and DECAPODES let's us:

- Specify physical laws with a new language of hierarchical diagrams
- Providing rigorous definition of "multi-physics" and Tonti Diagrams
- Reduces development level of effort + Increases flexibility of software
- Simulation is automatically generated from the Diagram + Initial/Boundary conditions
- Simulation quality is comparable to SU2 on benchmark problems.

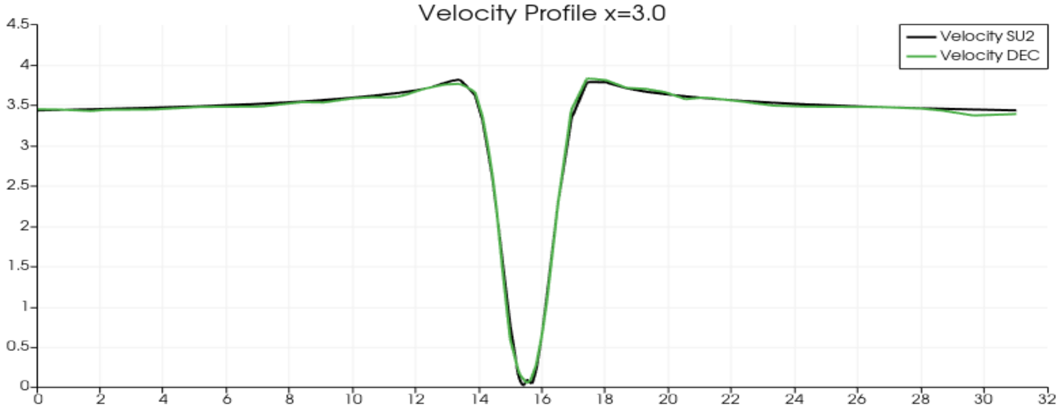
Conjugate Heat Transfer with Navier Stokes of Viscous Fluids

Model





Comparison to SU2 (established literature)
Velocity Profile

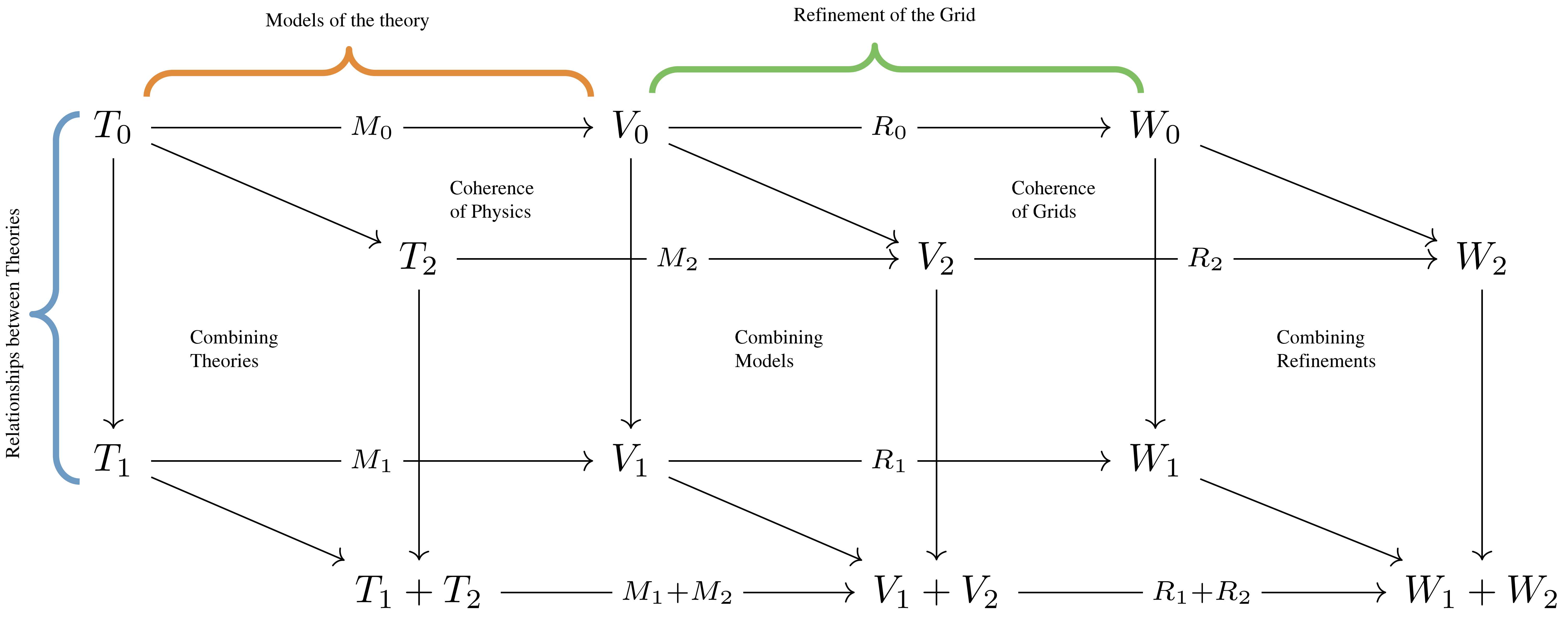


Future Work

- Numerics
  - New operators to solve more equations (1yr)
  - Interpolation/Discretization for converting to/from legacy specifications (3-5yrs)
  - FEEC, Higher Order Stencils, Spline/Polynomial, Spectral Methods (5-10 yrs)
  - Automatically choosing numerics based on problem structure (5-10yrs)
- Meshing and Geometry
  - Convergence Analysis as you change the mesh
  - Multigrid and Adaptive Mesh Refinement
  - Cubical Complexes
  - Arbitrary Mesh Shapes
  - Mesh-free Solvers
- Performance:
  - Parallel DEC operators / Task parallelism
  - HPC implementations of advanced numerics
  - Novel Architectures (non-VN)
- Optimization
  - Model Calibration / Inverse Problems
  - Structure / Mesh optimization
  - Data Assimilation / Digital Twins
  - Optimizing over equations with expert guidance
  - Learning Physics from Data

10 Year Goal: Fundamentally reshape how scientists and engineers develop simulations

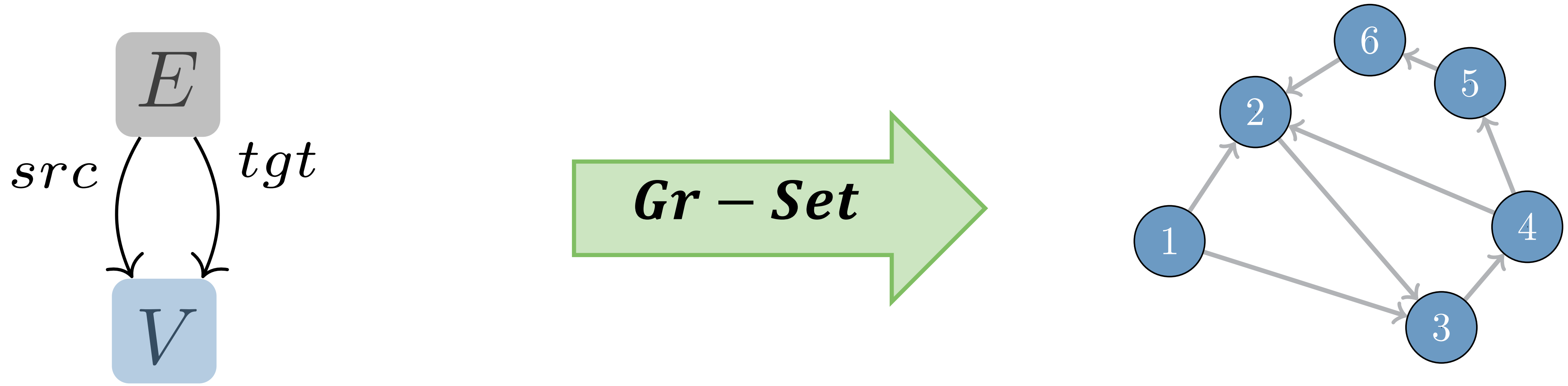
Multigrid for Multiphysics



$\mathcal{C}$-Sets: Categorical Data Structures

Graphs are ubiquitous because they a simple & useful structure

Gr



$\mathcal{C}$-Sets generalize algebraic graph theory

Rxn