Saaz: A Computational Database
The Query Framework
Alden King; Scott Baden, Advisor

Motivation
- Numerical simulations produce a lot of data.
- Methods to analyze this data are ad-hoc and not very sophisticated.
- “Developing new implementations [of data analysis codes] is often perceived as a poor use of human resources.” [1]
- Saaz provides a customizable framework for scientists to develop their own tools.
- Saves valuable time by reducing maintenance effort.

Framework Primitives
- Designed for Cartesian meshes.
  - Points
  - Domains which may be dense or sparse
  - Arrays are defined on domains
- Saaz is designed for computationally intensive query operations.
- We preserve data locality through loop nests.
- We use an imperative query model (as opposed to declarative, like SQL)

```plaintext
\forall Point1 \ p \in \text{domain}(w, j)
\quad \text{Domain2 plane} := \text{Slice(domain}(w), p, j)
\quad \text{sum} := 0
\quad \forall \text{Point} \ p2 \in \text{plane}
\quad \text{sum} += w[p2]
\quad \text{plane_avg}[p] := \text{sum}/\text{size(plane)}
```

Performance
- Queries can be composed, making it easy to build domain-specific abstractions.
- A simple mechanical process removes the performance overhead of the abstractions used by the Saaz framework.
- While this is done by hand at the moment, we are working on automating the process.
- The buoyancy flux (Left, Top) and \( \lambda_2 \) queries (Left, Bottom) speedup without the abstractions.

Case Study
Turbulent Flow in Computational Fluid Dynamics
With Eric Arobone and Sutanu Sarkar
A shear layer causes turbulent flow
- Saaz supports both online and offline analysis.
  - Online allows improved temporal resolution.
  - Offline allows more interactivity.

References

Support: NSF (OCE-0835839)