SciFinance: Commercial product since 1998
- Transforms high-level declarative language into efficient, stand-alone code
- Uses PDE and Monte Carlo simulations
- Sold to major banks to price derivatives

Case studies show reuse of math (PDE) knowledge
- Explore seismic wave propagation between nearby oil wells
- Measure sonic wave transit time in a moving fluid
- Model fracture propagation under hydraulic pressure
- Simulate chemical reaction kinetics
- Model simple black hole behavior

Features of layered synthesis
- Intermediate pseudo-code makes it easy to add code generators (Fortran90, C, C++)
- Facilitates parallelism
  - Specifications declarative, equation and constraint based
  - Imperative/sequential constructs introduced only when required
- Enables math methods independent of domain specifics and target language, (architecture-specific methods OK)

Math capabilities
- PDEs and PIDEs
  - any linear, many non-linear
  - any number of equations
  - any dimensionality
  - fast ADI and other implicit solvers
- SDEs (Stochastic Differential Equation)
  - jumps (arbitrary distributions)
  - correlations
- Discrete events (off regular time steps)
- Sensitivity measures on any parameter
- Complete (but optional) control over discretizations, grids, boundary conditions

Advantages of symbolic algebra
- Algorithm template language (independent from dimensions, equations, data structures/stencils)
- Generic specification of scientific computations
  - Coordinate transforms, discretizations
  - Sensitivity and error calculations
- Algorithm elaboration and optimization
- Program generation and optimization
  - Comments generated to aid human understanding
  - Symbolic simplification of general data representations and control constructs