- Julia Computing, Inc.
- Lead Investigator
  - Dr. Andrew Greenwell, PhD Computational Optics/Photonics
    - UCF/CREOL, Interactive Supercomputing, Microsoft HPC, WiTricity
- Company Founders and Key Employees:
  - Prof. Alan Edelman – MIT Mathematics Department, Julia co-creator
  - Dr. Jeff Bezanson – PhD CS MIT, BS CS Harvard, Julia co-creator
  - Dr. Viral Shah – PhD CS UCSB, BS CS Univ. of Mumbai, Julia co-creator
  - Stefan Karpinski – PhD (abd) CS UCSB, BS CS Harvard, Julia co-creator
  - Keno Fischer – MS/BS Physics Harvard, Julia core contributor
  - Deepak Vinchhi – MS MechE Univ. of Cincinnati, BS IIT Bombay, JC business lead
  - Jameson Nash – BS Aero/Astro MIT, Julia core contributor
  - Maren Cattonar – MS Business/Eng. UPENN, GradCert, JHU, BSs RPI- Eng/Econ.

Julia computing

- Julia is the future of technical computing
  - Open-source language and ecosystem
    - Julia combines a fast, sophisticated compiler, natural mathematical syntax, best-of-breed numerical libraries with an active world-wide community (> 100k users)
      - Base language and >800 packages available on GitHub
      - Existing JuliaOpt, JuliaStats, JuliaGeometry, JuliaDB organizations
  - Simple integration with other languages
    - Julia can be integrated easily with existing toolchains
    - Call C/Fortran libraries directly, no boilerplate wrappers
    - Embed Julia as a library in existing C applications
    - PyCall.jl, JavaCall.jl, RCall.jl, Matlab.jl, Cxx.jl, Clang.jl

Julia computing

- Julia Computing areas of research interest:
  - High performance, high productivity technical computing
  - Improve overall workflow for scientific computing
  - Modern programming language design
- Application of Julia to EDA/TCAD workflow
  - Rapid prototyping of numerically intensive algorithms
    - Physics simulation, optimize route/trace, device and circuit parameter extraction
  - Keep using existing software stacks where desired
    - No need to throw out what already works
    - Call Julia algorithms as libraries from existing C/C++ code

Julia computing

- Julia provides scientists and engineers with high performance programming tools that solve the "two-language problem"
  - Prototyping in one language (MATLAB®, Python, R), deploying in another language (C/C++, Fortran, Java)
- Julia Computing can assist in developing next-generation SCE EDA/TCAD tools
  - SuperTools seeks to introduce new features throughout the design stack in a market with decades of legacy
  - JC can partner on numerical and scientific algorithm development, integration, and deployment

Contact Information

- Dr. Andrew Greenwell
- Senior Application Engineer
- Julia Computing, Inc.
- andy@juliacomputing.com
- (617) 286-6399
- www.juliacomputing.com
- www.julialang.org