Julia: The Future of Numerical Computing and Data Science

December 2015

6th Emerging Markets Finance Conference
Why Julia for me?

Wall Street CTO
- Stuck with legacy systems, languages, databases
- Next generation platform for trading, portfolio management, risk

Engineering CTO
- Stuck with legacy languages that are cumbersome, and expensive
- Don’t want to re-develop for deployment in C++

Retail, Telco CIO
- Lots of data generated, with big data stack in place
- Want real-time analytics, inline with the transaction flow

IoT CTO
- Language barriers: sensors, gateway, and cloud. Skill shortage.
- Easy to use programming tools to build solutions for billions of devices
Julia in finance

- Customers of Julia Computing include 3 of the top 10 banks
- Use cases include:
  1. Asset allocation
  2. Asset pricing
  3. Portfolio optimization
  4. Risk computations
  5. Trading systems
- Unlike many other high productivity platforms, solutions developed in Julia are deployed in Julia
This website presents a series of lectures on quantitative economic modelling, designed and written by Thomas J. Sargent and John Stachurski. The primary programming languages are Python and Julia. You can send feedback to the authors via our web forum quantecon or webmaster@quant-econ.net.
We tested our code and found that the model estimation is about ten times faster with Julia than before, a very large improvement. Our ports of certain algorithms, such as Chris Sims’s gensys (which computes the model solution), also ran about six times faster in Julia.

Source: http://libertystreeteconomics.newyorkfed.org/2015/12/the-frbny-dsge-model-meets-julia.html
Technology Challenges in Finance

- Algorithm development for trading
- High speed trading
- Pricing complex instruments and VaR computations
- Regulatory compliance

Innovation Cycle can take Months

Develop Algorithms → Re-Program in low level language → Deploy on Public or Private Cloud
Julia Compresses Innovation Cycles

Today – Innovation Cycle in Months

1. Develop Algorithms
2. Re-Program in low level language
3. Deploy on Public or Private Cloud

Julia – Innovation Cycle in Days or Weeks

1. Develop Algorithms
2. Deploy on Public or Private Cloud
Open Source Julia Community across the World
100,000 users

Nerve Centres in **Bangalore, Boston, and New York**
Research anchored at **MIT**
Julia Performs as well as C and Fortran

Performance benchmark relative to C. A value of 1 means as fast as C. Lower values are better.
Julia Package Ecosystem Pulse

Last updated 2015-12-11

Total number of packages by Julia version

http://pkg.julialang.org/pulse.html
How can we help you?

info@juliacomputing.com