Julia on trial

fitting in our tech radar.

Martijn Visser

2026-01-06

Introduction
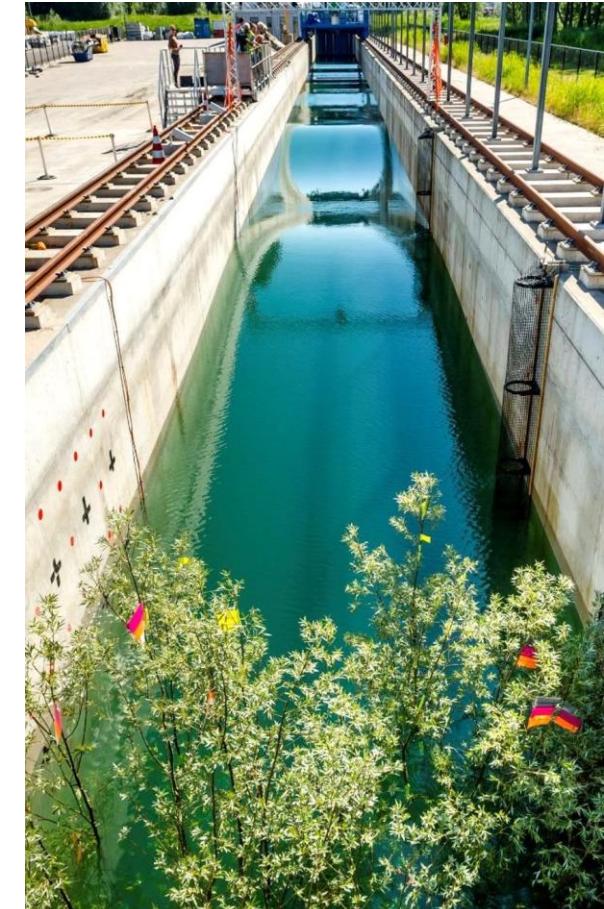
Martijn Visser
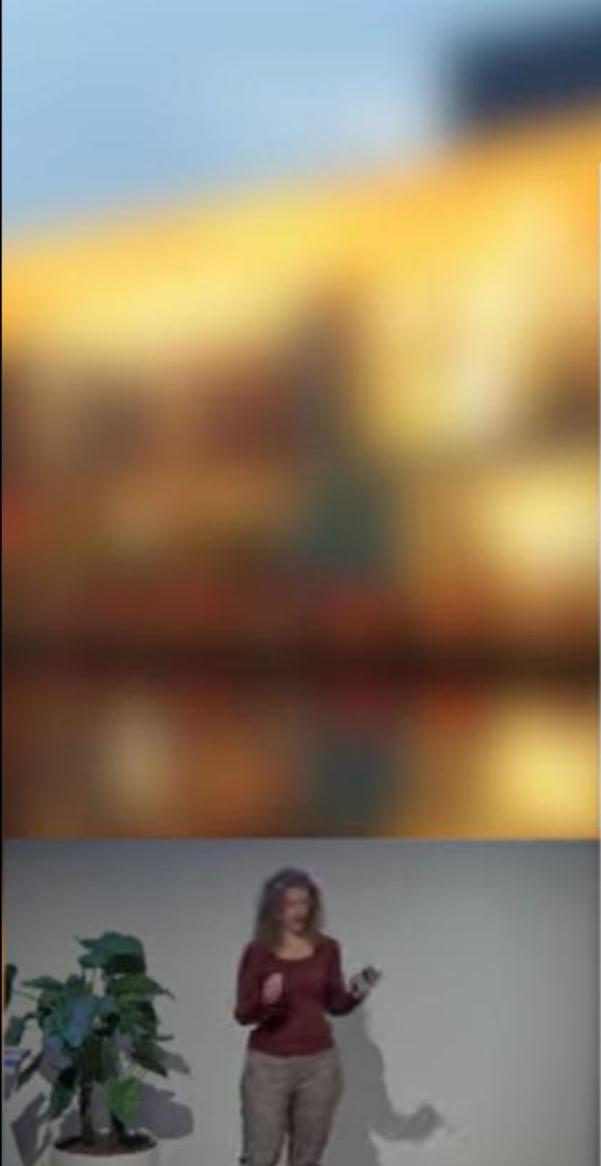
- Hydrologist at Deltares, the Netherlands
- JuliaGeo maintainer
- GitHub: visr
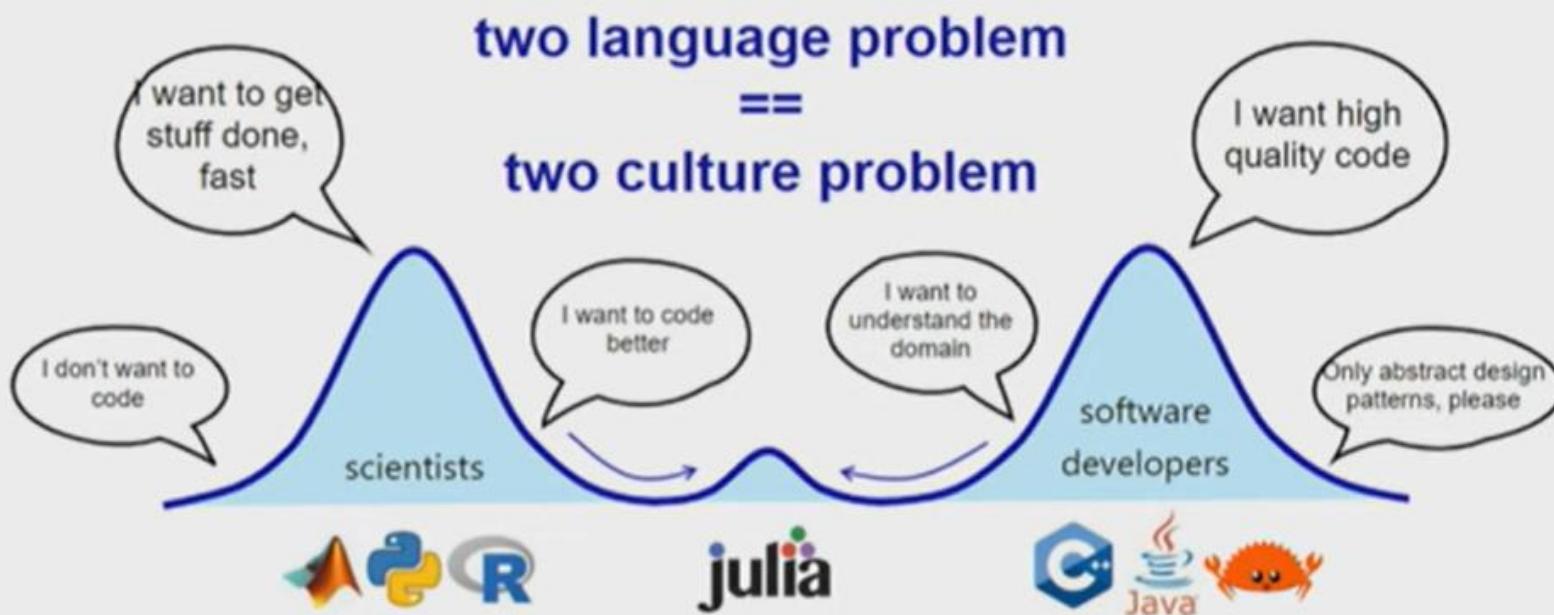
Deltares

- Independent research institute for applied research in the field of water and subsurface.

Myrthe Scheepers - ASMLs Julia Journey

Wflow

github.com/Deltares/Wflow.jl

Moselle (28,077 km^2)
Elevation [m]

- Spatially distributed hydrological model
- Originally written in Python
- Replacement in Julia
- Free and open-source software (MIT)
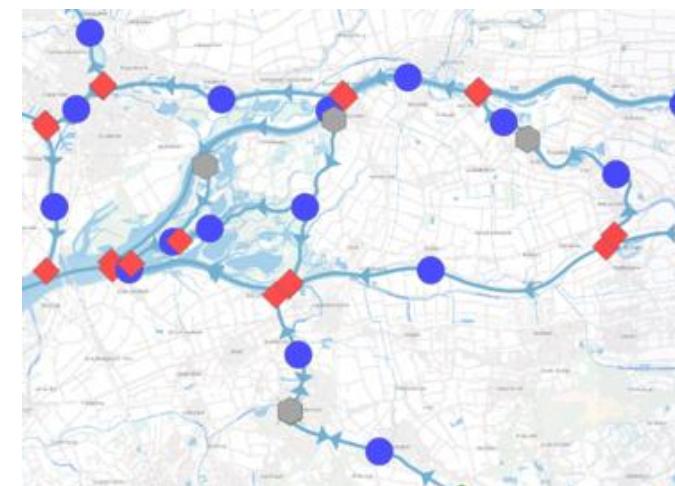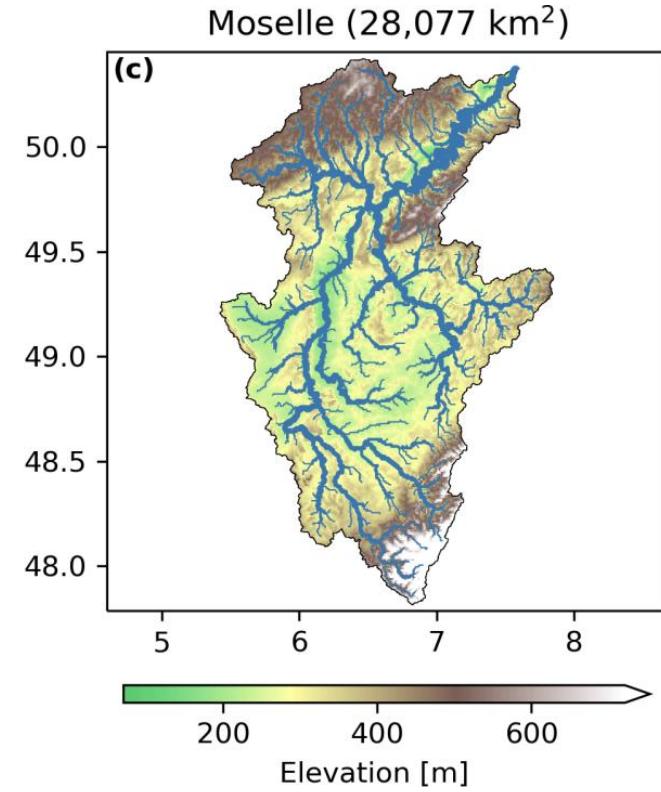
Ribasim

github.com/Deltares/Ribasim

- River basin planning and management software
- Legacy Fortran code dating back to the 1980s
- Replacement in Julia based on SciML & JuMP
- Used for Dutch national hydrological model
- Free and open-source software (MIT)

Deltares
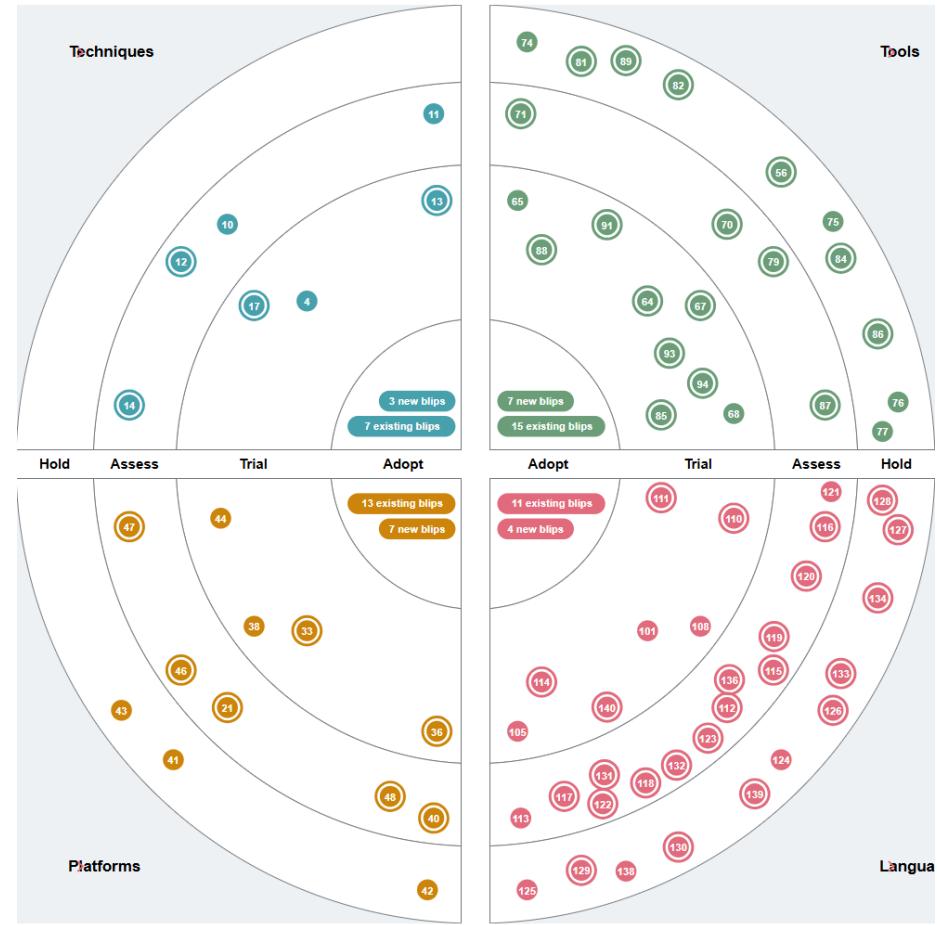
Power to the user, in Julia or not

- The user shouldn't need to know Julia to use Ribasim
- **Ribasim needs to integrate with existing workflows**
- We want to deliver compiled executables 📦
- 👉 PackageCompiler.jl / juliac

Deltares

techradar.deltares.nl

- **Adopt**: C++, Python

- **Trial**: Julia

- **Assess**: Rust, Fortran

- **Hold**: R, Matlab

- **Trial**. Blips we think are ready for use, and for which we are planning or performing trials. After trial, we will decide on their promotion to Adopt or demotion to Hold.

- **Motivation**: Julia seems to be a language with high potential, but there are some missing components involving ease of use for developers, and security scanning.

Things that can help

- [JuliaLang blog: Launching the Julia Security Working Group](#)

- [JETLS - new language server](#)

- [JuliaLang/JuliaC.jl](#)

- [github.com/visr/JI.jl](#) – rethinking the cli experience and defaults

- Chance to change defaults?
  - JULIA_NUM_THREADS=auto
  - JULIA_PROJECT=@script
  - juliaup config autoinstallchannels true
  - juliaup config manifestversiondetect true
  - instantiate
  - workspaces

Deltares

Contact

✉ martijn.visser@deltares.nl   ⊙ @visr   in linkedin.com/in/visr/