Scalable Embedded Scientific Computing with OpenCPU

Jeroen Ooms$^{1,2,3,⋆}$

1. UCLA Dept. of Statistics.
2. UCLA Center for Embedded Networked Sensing.
⋆ Contact author: jeroen.ooms@stat.ucla.edu

Keywords: Systems, Cloud Computing, Participatory Sensing, Web Applications, Reproducible Research

The UCLA Center for Embedded Networked Sensing (CENS) develops scalable open source systems for data collection and analysis [3]. In a current (2010-2015) project called Mobilize, thousands of students from high schools in Los Angeles will be armed with smartphones to collect data about their life and environment. Schools and students launch their own campaigns, and mobilize their peers to fill out surveys, make pictures and collect GPS-tagged data using an Android ‘app’ that communicates with a central server system. The students can share, explore and analyze data that they collectively gathered through web applications, or directly in $R$. The goal of the Mobilize project is to use participatory sensing to introduce the concept of learning from data to pre-college students in an intuitive and engaging way [1]. However, the same software is also used in other domains, like Mobile Health [2].

The Mobilize project illustrates how the OpenCPU framework [5] is used to embed $R$ into scalable software systems. OpenCPU is a web framework on top of rApache [4] which defines a protocol for interacting with $R$ over HTTP. It provides a layer of domain-specific functionality like object serialization, security, load balancing, resource control, reproducibility etc, while abstracting away technicalities. The OpenCPU protocol defines a direct mapping between R-functions and the RPC/REST interface. Hence, being able to write an R function is sufficient to publish R web services. This makes integration of $R$ in applications and systems much easier, because the statistician only has to supply R code, and the web/system developer only has to call the API. OpenCPU has proven to enable teams of statisticians and web developers to build R-powered web applications, without requiring them to learn each others language.

References


