Renjin: A new R interpreter built on the JVM

Alexander Bertram

Keywords: JVM, Optimization, Big Data

Renjin is a new interpreter for the R language built on the Java Virtual Machine. It is intended to be 100% compatible with GNU R, to run on Cloud-based PaaS like Google AppEngine, and generally to open new opportunities for embedding libraries and programs written in R within larger systems. Though still under development, the interpreter is already used in production by several companies to embed existing R packages within web applications.

In addition to facilitating integration, Renjin's principal advantage over GNU R is an additional layer of abstraction between R-language code and R data structures. Renjin allows developers to provide their own implementations of R vectors, so that the same R-language code might compute on an in-memory array, a memory-mapped file, or a rolling buffer over a database cursor.

This layer of abstraction allows other optimizations within the interpreter itself, such as “views” on large vectors. In Renjin, most primitive operations will not allocate new memory for a computation on a large vector, but simply return a new view of the underlying data. This makes it possible to defer computation until late in a program at which point it is compiled to optimized JVM bytecode in such a way that takes the entire computation, and memory limits, into account.

There are naturally many other initiatives to improve R's handling of large data, but Renjin has the distinct advantage of requiring no change to existing R-language code: the underlying storage of the data is invisible to the statistician.

The presentation will walk though a few concrete examples of how Renjin has helped solve problems in production, and lay out future paths for development.

References

http://code.google.com/p/renjin