Package ‘fastdigest’

April 6, 2024

Version 0.6-4

Title Fast, Low Memory Footprint Digests of R Objects

Description Provides an R interface to Bob Jenkin's streaming, non-cryptographic 'SpookyHash' hash algorithm for use in digest-based comparisons of R objects. 'fastdigest' plugs directly into R's internal serialization machinery, allowing digests of all R objects the serialize() function supports, including reference-style objects via custom hooks. Speed is high and scales linearly by object size; memory usage is constant and negligible.

Author Gabriel Becker, based on SpookyHash algorithm and C++ code by Bob Jenkins

Suggests RUnit

Maintainer Gabriel Becker <`gabembecker@gmail.com`>

License Artistic-2.0

RoxygenNote 7.3.1

NeedsCompilation yes

Repository CRAN

Date/Publication 2024-04-05 23:42:59 UTC

R topics documented:

`fastdigest` *Fast, memory constant hashing of R objects*

Description

Fast, memory constant hashing of R objects

Usage

```
fastdigest(obj, ref_serializer = NULL)
```

Arguments

`obj`	The object to generate a hash digest for
`ref_serializer`	(optional) A serializer for reference-style objects, see `serialize`

Details

`obj` will be hashed using R's internal serialization logic with a custom target which applies applying Jenkins' SpookyHash (v2) in a streaming fashion. This avoids (ever) copying the data out of the R object itself, providing both speed and memory constancy.

It also guarantees that the "representation" of the R object being hashed is the same as the serialized version would be, if created.

Author(s)

Gabriel Becker

References

Jenkins, B. (2012). SpookyHash: a 128-bit noncryptographic hash. http://burtleburtle.net/bob/hash/spooky.html.

See Also

`serialize`

Examples

```
fastdigest(1:5)
fastdigest(list("what", 1:2))
```

Index