Package 'sparkwarc'

January 13, 2017

Type Package

Title Load WARC Files into Apache Spark

Version 0.1.1

Maintainer Javier Luraschi <javier@rstudio.com>

Description Load WARC (Web ARChive) files into Apache Spark using 'sparklyr'. This allows to read files from the Common Crawl project <http://commoncrawl.org/>.

License Apache License 2.0

BugReports https://github.com/javierluraschi/sparkwarc

Encoding UTF-8

LazyData true

Imports sparklyr, DBI

RoxygenNote 5.0.1

NeedsCompilation no

Author Javier Luraschi [aut, cre]

Repository CRAN

Date/Publication 2017-01-13 06:42:24

R topics documented:

cc_warc *Provides WARC paths for commoncrawl.org*

Description

Provides WARC paths for commoncrawl.org. To be used with `spark_read_warc`.

Usage

```
cc_warc(start, end = start)
```

Arguments

`start`	The first path to retrieve.
`end`	The last path to retrieve.

Examples

```
cc_warc(1)
cc_warc(2, 3)
```

spark_read_warc *Reads a WARC File into Apache Spark*

Description

Reads a WARC (Web ARChive) file into Apache Spark using sparklyr.

Usage

```
spark_read_warc(sc, name, path, repartition = 0L, memory = TRUE,
  overwrite = TRUE, group = FALSE, parse = FALSE, ...)
```

Arguments

`sc`	An active `spark_connection`.
`name`	The name to assign to the newly generated table.
`path`	The path to the file. Needs to be accessible from the cluster. Supports the '`"hdfs://"`', '`"s3n://"`' and '`"file://"`' protocols.
`repartition`	The number of partitions used to distribute the generated table. Use 0 (the default) to avoid partitioning.
`memory`	Boolean; should the data be loaded eagerly into memory? (That is, should the table be cached?)

overwrite	Boolean; overwrite the table with the given name if it already exists?
group	TRUE to group by warc segment. Currently supported only in HDFS and uncompressed files.
parse	TRUE to parse warc into tags, attribute, value, etc.
...	Additional arguments reserved for future use.

Examples

```
library(sparklyr)
sc <- spark_connect(master = "spark://HOST:PORT")
df <- spark_read_warc(
  sc,
  system.file("samples/sample.warc", package = "sparkwarc"),
  repartition = FALSE,
  memory = FALSE,
  overwrite = FALSE
)

spark_disconnect(sc)
```

Index