

# Package ‘RESS’

February 26, 2015

**Type** Package

**Title** Integrates R and Essentia

**Version** 1.0

**Date** 2015-02-12

**Author** Ben Waxer

**Maintainer** Who to complain to <bwaxer@auriq.com>

**Description** Query the Essentia Database and access the results in R

**License** Same as Essentia. Free to use but do not make or distribute any copies.

## R topics documented:

RESS-package	1
essQuery	2
read.udb	3

<b>Index</b>	<b>5</b>
--------------	----------

---

RESS-package	<i>Integrates R and Essentia.</i>
--------------	-----------------------------------

---

## Description

essQuery takes a single Essentia command and captures the output in R, where you can save the output to a dataframe or stream it directly into additional analysis.

read.udb takes a file containing any number of Essentia commands and captures the output of the specified statements into R dataframes.

read.udb can be used to stream multiple files directly into R with a single command but essQuery is limited to one file per command.

**Details**

Package: RESS  
 Type: Package  
 Version: 1.0  
 Date: 2015-02-24  
 License: Same as Essentia. Free to use but do not make or distribute any copies.

**Author(s)**

Ben Waxer, Data Scientist with Auriq Systems.  
 Maintainer: Who to complain to <bwaxer@auriq.com>

---

 essQuery

*essQuery*


---

**Description**

Query the UDB database and return the results to R.

**Usage**

```
essQuery(essentia, aq="", flags="")
```

**Arguments**

essentia	<p>The <code>essentia</code> command to run. The options are "ess task stream category start-date enddate" and "ess task exec".</p> <p>Each stream command can only be used to stream one file directly into your R analysis. However, each stream command can save multiple files into a single R dataframe.</p> <p>The default value for the <code>essentia</code> argument is "ess task exec".</p>
aq	<p>This can be any combination of the <code>aq_tools</code> and standard UNIX commands. However, the output <b>MUST</b> be in a csv format if you want R to capture the output. If you only want to run the command without R capturing the output, add "#Rignore" to the flags argument.</p>
flags	<p>Any of the <code>essentia</code> flags can be used here in addition to any of these RESS-specific flags:</p> <p>#Rignore : Ignore an 'ess task exec' statement. Do not capture the output of the statement into R.</p> <p>#Rinclude : Include an 'ess task stream' statement. Capture the output of the statement into R.</p> <p>#-notitle : Tell R not to use the first line of the output as the header.</p> <p>#Rseparate : Can be used when saving multiple files into an R dataframe using an 'ess task stream' command. Saves each file into a different R dataframe, entitled <code>command1</code> to <code>commandN</code>, where N is the number of files.</p>

`#R#name#R#` : Allows any automatically saved dataframe to be renamed to whatever is entered in place of 'name'. This only applies in `essQuery` when streaming multiple files.

## Details

`essQuery` is used to directly query the database using a single statement. You can call `essQuery` multiple times to run different statements.

However, you can also use `read.udb` to read all of the statements in a file instead. Thus if you plan to run multiple statements that may be somewhat related to each other, it is recommended that you use `read.udb`.

## Value

The value returned is the output from querying the database. This can be saved into an R dataframe or directly analyzed in R.

If you use `essQuery` to save multiple files into an R dataframe using a single stream command, the files are stored automatically in an R dataframe called `command1` and no value is returned. To change the name of the stored dataframe, use the `#R#NAME#R#` flag.

## Author(s)

Ben Waxer, Data Scientist with Auriq Systems.

## References

See our website at [www.auriq.net](http://www.auriq.net) or our documentation at [www.auriq.net/documentation](http://www.auriq.net/documentation)

---

`read.udb`

*read.udb*

---

## Description

Read the `essentia` commands in the stated file, query the UDB database, and save the results into R as dataframes.

## Usage

```
read.udb(file, linenumber = "all")
```

## Arguments

<code>file</code>	<p>The file containing the <code>essentia</code> commands you want to run. 'ess task stream' statements will have their output ignored by R unless there is a '<code>#Rinclude</code>' flag somewhere in the statement line. 'ess task exec' statements will have their output included unless there is a '<code>#Rignore</code>' flag somewhere in the statement line.</p> <p>The file should contain primarily the query commands but can contain the entire <code>essentia</code> script including loading the UDB database if desired. Any command that you want to capture the output from must have its output in csv format.</p>
<code>linenumber</code>	<p>This is set to default to all line numbers so that every command in the file is executed. You can specify the line number of the command you wish to run if you dont want to run the entire set of commands in the file.</p>

## Details

read.udb reads all of the statements in a file (unless linenumber is specified, see above) and captures the output of the specified commands into R dataframes. By default only the output of 'ess task exec' statements is captured and it's stored in R dataframes command1 to commandN, where N is the number of captured statements.

You can include 'ess task stream' statements by adding a '#Rinclude' flag. This method can be used to stream multiple files into R for data exploration or analysis. Thus if you plan to run multiple statements that may be somewhat related to each other, or you want to stream multiple files into R, it is recommended that you use read.udb.

## Value

There is no value returned. This command creates a set of R dataframes containing the output from the specified essentia commands in file.

## Note

The flags added to the essentia commands in file can include:

#Rignore : Ignore an 'ess task exec' statement. Do not capture the output of the statement into R.

#Rinclude : Include an 'ess task stream' statement. Capture the output of the statement into R.

#-notitle : Tell R not to use the first line of the output as the header.

#Rseparate : Can be used when saving multiple files into an R dataframe using an 'ess task stream' command. Saves each file into a different R dataframe, entitled command1 to commandN, where N is the number of files.

#R#name#R# : Allows any automatically saved dataframe to be renamed to whatever is entered in place of 'name'.

## Author(s)

Ben Waxer, Data Scientist with Auriq Systems.

## References

See our website at [www.auriq.net](http://www.auriq.net) or our documentation at [www.auriq.net/documentation](http://www.auriq.net/documentation).

# Index

\*Topic **package**

RESS-package, [1](#)

essQuery, [2](#)

read.udb, [3](#)

RESS (*RESS-package*), [1](#)

RESS-package, [1](#)