

# Package ‘Rcolombos’

January 27, 2015

**Type** Package

**Title** Interface to Colombos Compendia using the exposed REST API

**Version** 1.5.0

**Date** 2013-06-26

**Author** Paolo Sonogo <paolo.sonogo@fmach.it>

**Maintainer** Paolo Sonogo <paolo.sonogo@fmach.it>

**Description** This package provides programmatic access to Colombos, a web based interface for exploring and analyzing comprehensive organism-specific cross-platform expression compendia of bacterial organisms.

**License** GPL-3

**Depends** httr

**Repository** CRAN

**URL** <http://colombos.net/>

**Collate** 'Rcolombos.R' 'utilities.R' 'zzz.R'

**NeedsCompilation** no

**Date/Publication** 2014-03-26 10:01:29

## R topics documented:

advanced_search . . . . .	2
advanced_search_by_both . . . . .	3
advanced_search_by_contrasts . . . . .	5
advanced_search_by_genes . . . . .	6
getCompendium . . . . .	6
listAnnotationTypes . . . . .	7
listContrasts . . . . .	8
listEntities . . . . .	9
listGenes . . . . .	10
listOrganisms . . . . .	10
parseCompendium . . . . .	11
quick_search . . . . .	12

<b>Index</b>	<b>13</b>
--------------	-----------

---

advanced_search	<i>This method mimics the advanced_search functionality of Colombos. It takes a series of parameters, representing the different settings available on Colombos advanced search and returns a list containing the locustags (gene_names), contrasts and M-values for the current selection.</i>
-----------------	---

---

### Description

This method mimics the advanced\_search functionality of Colombos. It takes a series of parameters, representing the different settings available on Colombos advanced search and returns a list containing the locustags (gene\_names), contrasts and M-values for the current selection.

### Usage

```
advanced_search(organism = NULL, g_ids = NULL,
  geneNames = FALSE, c_ids, by = "genes", g_search_type,
  ann_type, c_search_type)
```

### Arguments

organism	A character containing the organism id: use <a href="#">listOrganisms</a> to display the available organisms.
g_ids	A vector of strings representing contrast_id, go terms, experiment id or condition id according the search type.
geneNames	boolean if == FALSE (default) return the locustag otherwise the gene_name for the selected genes.
c_ids	A vector of strings representing contrast_id, go terms, experiment id or condition id according the search type.
by	A string either genes, contrasts, both allowing the selection by genes entities, contrast entities or both.
g_search_type	A string either genes, go or annotation.
ann_type	A string containing the selected gene_annotation_type: use <a href="#">listEntities</a> to display the available entities.
c_search_type	A string either contrast_names. experiment, go, condition use <a href="#">listOrganisms</a> to display the available organisms.

### Value

A data.frame containing locustag (gene\_names), contrasts and M-values for the current organism and genes.

### References

<http://colombos.net>

**Examples**

```

## Not run:
library("Rcolombos")

# modules by gene entities
g.gn <- advanced_search(organism="bsubt",
  g_ids=c("cgeB", "yfnG"),
  by="genes", g_search_type="genes")
g.go <- advanced_search(organism="bsubt",
  g_ids="response to antibiotic, transcription",
  by="genes", g_search_type="go")
g.anno <- advanced_search(organism="bsubt",
  g_ids="biotin-carboxyl carrier protein assembly",
  by="genes", g_search_type="annotation", ann_type="Pathway")

# modules by contrast entities
c.cn <- advanced_search(organism="bsubt",
  c_ids=c("GSM27217.ch2-vs-GSM27217.ch1", "GSM27218.ch1-vs-GSM27218.ch2"),
  by="contrasts", c_search_type="contrast_names")
c.go <- advanced_search(organism="bsubt",
  c_ids="response to antibiotic, transcription",
  by="contrasts", c_search_type="go")
c.exp <- advanced_search(organism="bsubt",
  c_ids="GSE22296", by="contrasts", c_search_type="experiment")
c.cond <- advanced_search(organism="bsubt",
  c_ids=c("DAPTOMYCIN", "H202", "HPUra", "IPTG", "MMC", "MNCL2", "MOENOMYCIN", "RAMOPLANIN"),
  by="contrasts", c_search_type="condition")

# modules by both gene and contrast entities
b.go.cn <- advanced_search(organism="bsubt",
  g_ids="response to antibiotic, transcription", geneNames=F,
  c_ids=c("GSM27217.ch2-vs-GSM27217.ch1", "GSM27218.ch1-vs-GSM27218.ch2"),
  g_search_type="go", c_search_type="contrast_names", by="both")
b.gn.ge <- advanced_search(organism="bsubt", g_ids=c("BSU00020", "BSU00100"),
  geneNames=F, c_ids="GSE22296", g_search_type="genes",
  c_search_type="experiment", by="both")
b.go.ge <- advanced_search(organism="bsubt", g_ids="response to antibiotic, transcription",
  geneNames=F, c_ids="GSE22296", g_search_type="go",
  c_search_type="experiment", by="both")
b.gn.cn <- advanced_search(organism="bsubt",
  g_ids=c("dnaA", "dnaN", "yaaA", "recF", "yaaB", "gyrB"), geneNames=FALSE,
  c_ids=c("GSM27217.ch2-vs-GSM27217.ch1", "GSM27218.ch1-vs-GSM27218.ch2",
  "GSM27219.ch2-vs-GSM27219.ch1", "GSM27278.ch2-vs-GSM27278.ch1",
  "GSM27279.ch1-vs-GSM27279.ch2"),
  g_search_type="genes", c_search_type="contrast_names", by="both")
heatmap(as.matrix(b.gn.cn), col=terrain.colors(15))

## End(Not run)

```

---

advanced\_search\_by\_both

*Accessory function allowing the advanced\_search by both g\_ids and c\_ids*

---

## Description

Accessory function allowing the advanced\_search by both g\_ids and c\_ids

## Usage

```
advanced_search_by_both(organism, g_ids, geneNames,
  c_ids, g_search_type, ann_type, c_search_type)
```

## Arguments

organism	A character containing the organism id: use <a href="#">listOrganisms</a> to display the available organisms.
g_ids	A vector of strings representing contrast_id, go terms, experiment id or condition id according the search type.
geneNames	boolean if == FALSE (default) return the locustag otherwise the gene_name for the selected genes.
c_ids	A vector of strings representing contrast_id, go terms, experiment id or condition id according the search type.
g_search_type	A string either genes, go or annotation.
ann_type	A string containing the selected gene_annotation_type: use <a href="#">listEntities</a> to display the available entities.
c_search_type	A string either contrast_names. experiment, go, condition.

## Value

A data.frame containing locustag (gene\_names), contrasts and M-values for the current organism and genes.

## References

<http://colombos.net>

## Examples

```
## Not run:
c.cn <- advanced_search_by_contrasts(organism="bsubt",
  ids=c("GSM27217.ch2-vs-GSM27217.ch1", "GSM27218.ch1-vs-GSM27218.ch2"), search_type="contrast_names")

## End(Not run)
```

---

advanced\_search\_by\_contrasts

*Accessory function allowing the advanced\_search by contrast\_ids, go, experiment, condition*

---

## Description

Accessory function allowing the advanced\_search by contrast\_ids, go, experiment, condition

## Usage

```
advanced_search_by_contrasts(organism = NULL, ids = NULL,  
geneNames = FALSE, c_search_type = NULL)
```

## Arguments

organism	A character containing the organism id: use <a href="#">listOrganisms</a> to display the available organisms.
ids	A vector of strings representing contrast_id, go terms, experiment id or condition id according the search type.
geneNames	boolean if == FALSE (default) return the locustag otherwise the gene_name for the selected genes.
c_search_type	A string either contrast_names. experiment, go, condition.

## Value

A data.frame containing locustag (gene\_names), contrasts and M-values for the current organism and genes.

## References

<http://colombos.net>

## Examples

```
## Not run:  
c.cn <- advanced_search_by_contrasts(organism="bsubt",  
ids=c("GSM27217.ch2-vs-GSM27217.ch1", "GSM27218.ch1-vs-GSM27218.ch2"), search_type="contrast_names")  
  
## End(Not run)
```

---

advanced\_search\_by\_genes

*Accessory function allowing the advanced\_search by gene\_ids, go, annotation*

---

### Description

Accessory function allowing the advanced\_search by gene\_ids, go, annotation

### Usage

```
advanced_search_by_genes(organism = "bsubt", ids = NULL,
  geneNames = FALSE, g_search_type = "genes", ann_type)
```

### Arguments

organism	A character containing the organism id: use <a href="#">listOrganisms</a> to display the available organisms.
ids	A vector of strings representing gene_id, go terms or annotation entities according the search type.
geneNames	boolean if == FALSE (default) return the locustag otherwise the gene_name for the selected genes.
g_search_type	A string either genes, go or annotation.
ann_type	A string containing the selected gene_annotation_type: use <a href="#">listEntities</a> to display the available entities.

### Value

A data.frame containing locustag (gene\_names), contrasts and M-values for the current organism and genes.

### References

<http://colombos.net>

---

getCompendium

*This method allows to download/import the full compendium for the selected organism*

---

### Description

This method allows to download/import the full compendium for the selected organism

**Usage**

```
getCompendium(organism = "hpylo", path = NULL)
```

**Arguments**

organism	A character containing the organism id: use <a href="#">listOrganisms</a> to display the available organisms.
path	A string indicating the path where the file will be either downloaded or read, if already retrieved

**Value**

A list containing three data.frame:

exprdata	the full compendium for the selected organism
condannot	The condition annotation for the selected organism
condontol	the condition ontology for the selected organism

**References**

<http://colombos.net>

**Examples**

```
## Not run:  
library('Rcolombos')  
hpylo <- getCompendium("hpylo")  
  
## End(Not run)
```

---

`listAnnotationTypes` *This method takes as parameter a string (the nickname of an organism) and returns a character vector corresponding to the currently available annotation type for the selected organism.*

---

**Description**

This method takes as parameter a string (the nickname of an organism) and returns a character vector corresponding to the currently available annotation type for the selected organism.

**Usage**

```
listAnnotationTypes(organism = "ecoli")
```

**Arguments**

organism	A character containing the organism id: use <a href="#">listOrganisms</a> to display the available organisms.
----------	---

**Value**

A data.frame containing the name and description of the annotation for the selected organism.

**References**

<http://colombos.net>

**Examples**

```
## Not run:  
library('Rcolombos')  
listAnnotationTypes()  
  
## End(Not run)
```

---

listContrasts	<i>This method takes as parameter a single string, representing an organism, and returns a character vector corresponding to the currently available organisms.</i>
---------------	---

---

**Description**

This method takes as parameter a single string, representing an organism, and returns a character vector corresponding to the currently available organisms.

**Usage**

```
listContrasts(organism = "ecoli")
```

**Arguments**

organism      A character containing the organism id: use [listOrganisms](#) to display the available organisms.

**Value**

A data.frame containing the contrasts and GSM of all the contrasts for the selected organism.

**References**

<http://colombos.net>

**Examples**

```
## Not run:  
library('Rcolombos')  
listContrasts()  
  
## End(Not run)
```



---

listEntities	<i>This method takes a string containing the nickname for the selected organism and a string containing the annotation type and return the available entities</i>
--------------	---

---

### Description

This method takes a string containing the nickname for the selected organism and a string containing the annotation type and return the available entities

### Usage

```
listEntities(organism = "ecoli", annotation = "Pathway")
```

### Arguments

organism	A character containing the organism id: use <a href="#">listOrganisms</a> to display the available organisms.
annotation	A character containing the selected annotation type: use <a href="#">listAnnotationTypes</a> to display the available types.

### Value

A vector containing the available entities for the selected annotation type.

### References

<http://colombos.net>

### Examples

```
## Not run:  
library("Rcolombos")  
pathway_entities <- listEntities(organism="bsubt", annotation="Pathway")  
Tr_entities <- listEntities("bsubt","Transcriptional regulation")  
  
## End(Not run)
```

---

listGenes	<i>This method takes as parameter a single string, representing an organism, and returns a character vector corresponding to the currently available organisms.</i>
-----------	---

---

**Description**

This method takes as parameter a single string, representing an organism, and returns a character vector corresponding to the currently available organisms.

**Usage**

```
listGenes(organism = "ecoli")
```

**Arguments**

organism	A character containing the organism id: use <a href="#">listOrganisms</a> to display the available organisms.
----------	---

**Value**

A data.frame containing the locustag and description of all the genes for the selected organism.

**References**

<http://colombos.net>

**Examples**

```
## Not run:  
library('Rcolombos')  
listGenes()  
  
## End(Not run)
```

---

listOrganisms	<i>Returns a character vector corresponding to the currently available organisms.</i>
---------------	---

---

**Description**

Returns a character vector corresponding to the currently available organisms.

**Usage**

```
listOrganisms()
```

**Value**

A list containing the currently available organisms.

**References**

<http://colombos.net>

**Examples**

```
## Not run:
library('Rcolombos')
listOrganisms()

## End(Not run)
```

---

parseCompendium	<i>This method allows importing the full compendium for the selected organism from a local file</i>
-----------------	---

---

**Description**

This method allows importing the full compendium for the selected organism from a local file

**Usage**

```
parseCompendium(destfile)
```

**Arguments**

destfile      A character containing the full path of the downloaded file

**Value**

A list containing three data.frame:

exprdata	the full compendium for the selected organism
condannot	The condition annotation for the selected organism
condontol	the condition ontology for the selected organism

**References**

<http://colombos.net>

**Examples**

```
## Not run:
library('Rcolombos')
mtube <- parseCompendium("mtube_compendium_data.zip")

## End(Not run)
```

---

quick_search	<i>This method mimics the quick_search functionality of Colombos. It takes a string containing the nickname for the selected organism and a vector of string representing the genes of interest for the specified organism and returns a list containing the locustags (gene_names), contrasts and M-values for the current selection.</i>
--------------	--

---

### Description

This method mimics the quick\_search functionality of Colombos. It takes a string containing the nickname for the selected organism and a vector of string representing the genes of interest for the specified organism and returns a list containing the locustags (gene\_names), contrasts and M-values for the current selection.

### Usage

```
quick_search(organism = "ecoli", genes,  
             geneNames = FALSE)
```

### Arguments

organism	A character containing the organism id: use <a href="#">listOrganisms</a> to display the available organisms.
genes	A vector of strings representing the genes of interest.
geneNames	boolean if == FALSE (default) returns the locustag otherwise the gene_name for the selected genes.

### Value

A data.frame containing locustag (gene\_names), contrasts and M-values for the current organism and genes.

### References

<http://colombos.net>

### Examples

```
## Not run:  
library("Rcolombos")  
my_module <- quick_search(organism="ecoli",  
                           genes=c("b0400", "b2805", "b0567"),  
                           geneNames=FALSE)  
heatmap(as.matrix(my_module), col=terrain.colors(15))  
  
## End(Not run)
```

# Index

`advanced_search`, [2](#)  
`advanced_search_by_both`, [3](#)  
`advanced_search_by_contrasts`, [5](#)  
`advanced_search_by_genes`, [6](#)

`getCompendium`, [6](#)

`listAnnotationTypes`, [7](#), [9](#)  
`listContrasts`, [8](#)  
`listEntities`, [2](#), [4](#), [6](#), [9](#)  
`listGenes`, [10](#)  
`listOrganisms`, [2](#), [4–10](#), [10](#), [12](#)

`parseCompendium`, [11](#)

`quick_search`, [12](#)