webQueries: Retrieve genes or protein annotations by querying NCBI.

Frederic Commo[1]

[1]*INSERM U981, Institut Gustave Roussy, 114 rue Edouard Vaillant, 94805 Villejuif, France*

March 18, 2015

1 Introduction

`webQueries` provides R functions for retrieving up-to-date gene and protein annotations from NCBI databases, given HUGO symbols or ids, e.g. entrezgene or Uniprot ids.
The main function, `runQuery()`, is a wrapper for the NCBI E-utilities functions [1], and is designed to query *Gene*, *Protein*, and *SNP* databases. `runQueries` requires a functional web connexion.
In case a HUGO symbol is called, `runQuery` first interogate NCBI in order to get identifiers, using E-utilities `Esearch`, then use the returned values to get the corresponding annotations, using E-utilities `Esummary`.

2 Examples

Here are some examples showing how a single, or multiple, annotation(s) can be retrieved using either official symbols or identifiers.

Querying the NCBI *Gene* database, using HUGO symbols:

```
> require(webQueries)

Loading required package:  webQueries

> gquery <- runQuery("erbb2", "gene")
>
> # The first 5 items
> as.list(gquery)[1:5]
$query
[1] erbb2
Levels: erbb2

$Name
[1] ERBB2
Levels: ERBB2

$Description
[1] erb-b2 receptor tyrosine kinase 2
Levels: erb-b2 receptor tyrosine kinase 2
```

```
$Orgname
[1]
Levels:

$Status
[1] 0
Levels: 0
```

Querying the NCBI *Protein* database, using Uniprot ids:

```
> pquery <- runQuery("P04626", "protein", bySymbol = FALSE)
>
> # The first 5 items
> as.list(pquery)[1:5]
$query
[1] P04626
Levels: P04626

$Caption
[1] P04626
Levels: CAA27060 NP_001276866 NP_004439 P04626

$Title
[1] RecName: Full=Receptor tyrosine-protein kinase erbB-2; AltName: Full=Metastatic lymph node ge
4 Levels: RecName: Full=Receptor tyrosine-protein kinase erbB-2; AltName: Full=Metastatic lymph n

$Extra
[1] gi|119533|sp|P04626.1|ERBB2_HUMAN[119533]
4 Levels: gi|119533|sp|P04626.1|ERBB2_HUMAN[119533] ...

$Gi
[1] 119533
Levels: 119533 31198 54792096 584277106
```

When querying the NCBI `SNP` database, one may not be interested only in the last updates, but in all the outputs. To do so, the `updateOnly` argument must be set to `FALSE`:

```
> query <- runQuery("erbb2", "snp", updateOnly = FALSE)
> query[,1:5]
   query    SNP_ID Organism ALLELE_ORIGIN  GLOBAL_MAF
1  erbb2 587776805
2  erbb2 578192771                          T=0.0002/1
3  erbb2 578155528                          G=0.0002/1
4  erbb2 578138670
5  erbb2 578108865                          T=0.0002/1
6  erbb2 578046124                          T=0.0002/1
7  erbb2 577933020                         C=0.0068/34
8  erbb2 577787590                          G=0.0004/2
9  erbb2 577767674                         C=0.0166/83
10 erbb2 577608686                          T=0.0002/1
11 erbb2 577560557
12 erbb2 577523466                          A=0.0002/1
13 erbb2 577467099                          G=0.0002/1
```

```
14 erbb2 577449121                          T=0.0004/2
15 erbb2 577329788                          G=0.0002/1
16 erbb2 577272317                          T=0.0008/4
17 erbb2 577260523                          T=0.0002/1
18 erbb2 577200967                          T=0.0002/1
19 erbb2 577121666
20 erbb2 577083829                          G=0.0002/1
```

A simple way to run multiple queries would be to call `runQuery` within a loop, e.g. `lapply`. However, each returned xml file may not contain exactly the same items - some may not be available. `multiQueries` takes care of this, and returns the common items over all the queries.

```
> # Multiple queries on the Gene database, using HUGO symbols
> ids <- c("egfr", "erbb2", "fgfr1")
> annots <- multiQueries(ids, "gene")
Searching egfr
Searching erbb2
Searching fgfr1
> annots[,1:8]
  query  Name                          Description Orgname Status CurrentID
1  egfr  EGFR    epidermal growth factor receptor              0         0
2 erbb2 ERBB2   erb-b2 receptor tyrosine kinase 2              0         0
3 fgfr1 FGFR1 fibroblast growth factor receptor 1              0         0
  Chromosome GeneticSource
1          7       genomic
2         17       genomic
3          8       genomic
> # List of returned items
> names(annots)
 [1] "query"              "Name"               "Description"
 [4] "Orgname"            "Status"             "CurrentID"
 [7] "Chromosome"         "GeneticSource"      "MapLocation"
[10] "OtherAliases"       "OtherDesignations"  "NomenclatureSymbol"
[13] "NomenclatureName"   "NomenclatureStatus" "TaxID"
[16] "Mim"                "int"                "GenomicInfo"
[19] "GenomicInfoType"    "ChrLoc"             "ChrAccVer"
[22] "ChrStart"           "ChrStop"            "ExonCount"
[25] "GeneWeight"         "Summary"            "ChrSort"
[28] "Organism"           "ScientificName"     "CommonName"
[31] "GI"
```

3 Accessing R code

R source code for `webQueries` is available on github: `https://github.com/fredcommo/webQueries`

References

[1] URL: `http://www.ncbi.nlm.nih.gov/books/NBK25500/`.