How to use the MeSH.db Package

Koki Tsuyuzaki[1], Itoshi Nikaido[2], and Gota Morota[3].

September 26, 2013

[1]Department of Medical and Life Science, Tokyo University of Science.
[2]Bioinformatics Research Unit, RIKEN Advanced Center for Computing and Communication.
[3]Department of Animal Sciences, University of Wisconsin-Madison.

`k.t.the-answer@hotmail.co.jp`

Contents

1 Introduction

This document provides the way to use *MeSH.db* package. MeSH (Medical Subject Headings) is the NLM (U. S. National Library of Medicine) controlled vocabulary used to manually index articles for MEDLINE/Pubmed [1].
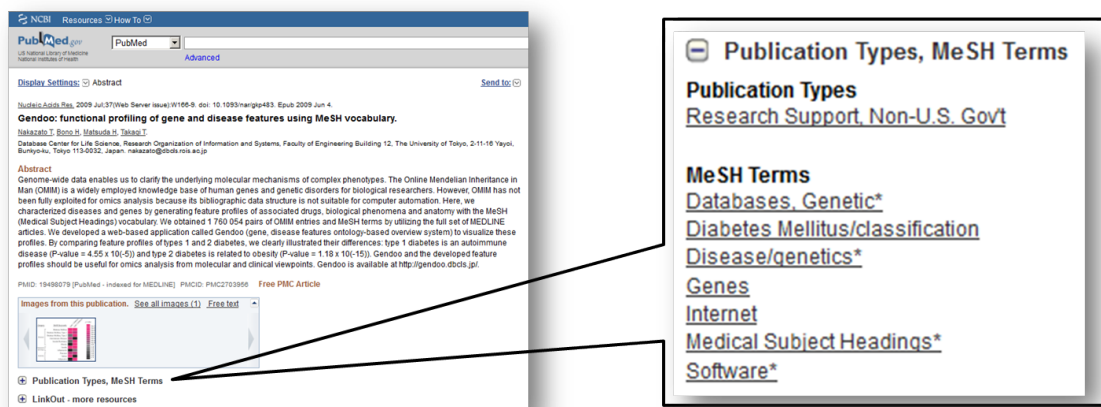

Figure 1: MeSH Term

The amount of vocabularies in MeSH is about twice as large as that of GO (Gene Ontology)[2] and its categories are also wider. Therefore MeSH is expected to be much detailed and exhaustive gene annotation tool. Some softwares or databases using MeSH are now proposed [3, 4, 5, 6].

MeSH.db is a free R package for handling MeSH in R. Original MeSH data is accessible by NLM FTP site (`http://www.nlm.nih.gov/mesh/filelist.html`). The data are downloadable as plain-text format (ASCII MeSH; d2013.bin and q2013.bin). These files were preprocessed by our programs (figure 2) and corresponding information was summalized as a table in sqlite file and packed into *MeSH.db* (see also section 4, explaining the meaning of all tables).

MeSH has hierarchical structure like GO. *MeSH.db* provides its Ancestor-Offspring Relationships (AOR) and Parent-Child Relationships (PCR) as corresponding table.
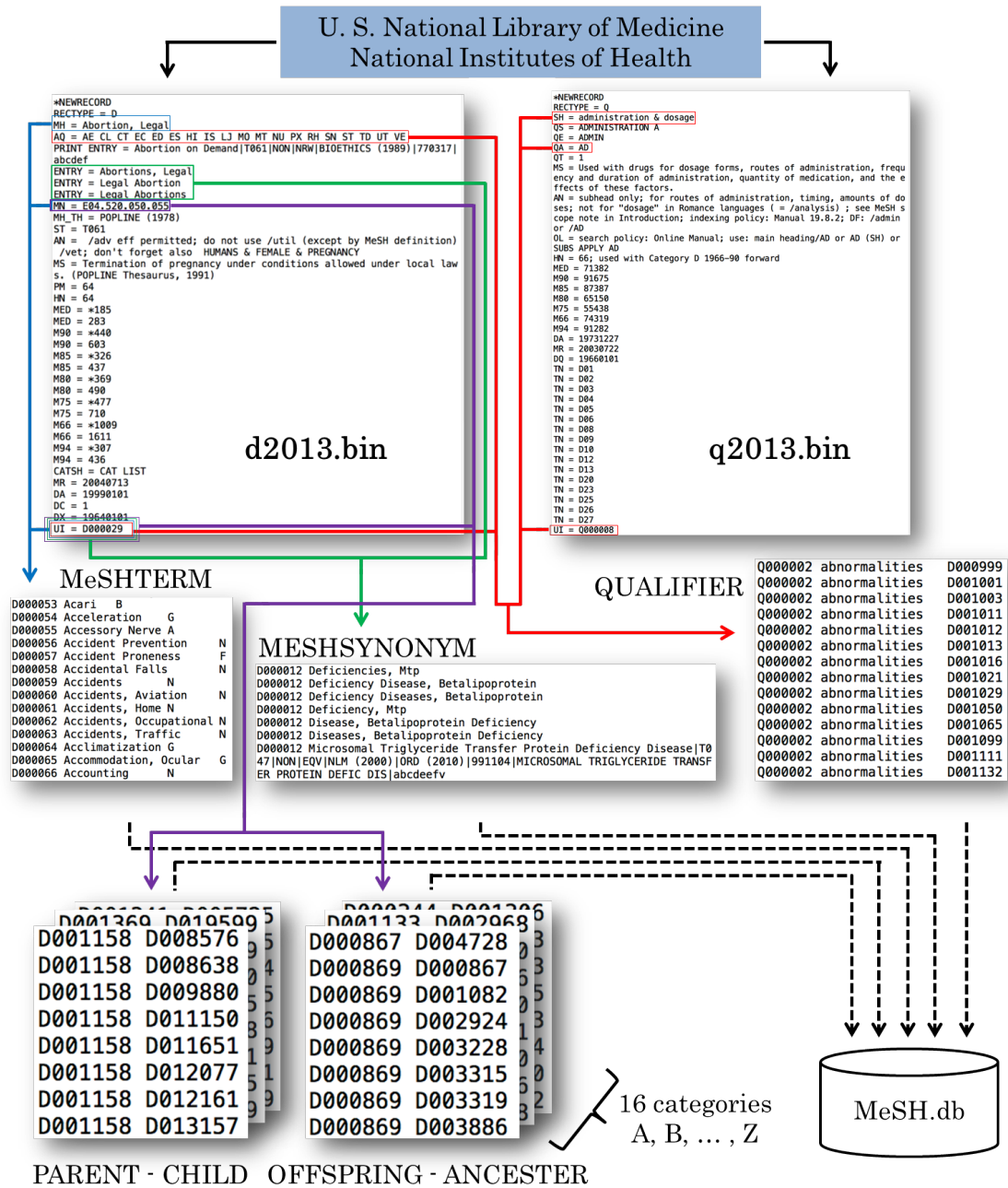
Figure 2: Data preprocess for MeSH.db

MeSH in 2013 proposed its 19 categories and MeSH.db provides 16 of them, which are actually assigned to some MeSH terms. Each category is expressed as single capital alphabet defined by NLM as abbreviation.

Abbreviation	Category
A	Anatomy
B	Organisms
C	Diseases
D	Chemicals and Drugs
E	Analytical, Diagnostic and Therapeutic Techniques and Equipment
F	Psychiatry and Psychology
G	Phenomena and Processes
H	Disciplines and Occupations
I	Anthropology, Education, Sociology and Social Phenomena
J	Technology and Food and Beverages
K	Humanities
L	Information Science
M	Persons
N	Health Care
V	Publication Type
Z	Geographical Locations

2 Getting started

To load the *MeSH.db* package, just type library(MeSH.db). 5 methods and 36 data are provided by *MeSH.db*.

3 Methods

Following 5 methods are provided by *MeSH.db*.

MeSH	Function for retrieval of the summary of all object in MeSH.db
MeSH_dbconn	Function for retrieval of the connection of .sqlite database
MeSH_dbfile	Function for retrieval of the directory of .sqlite file
MeSH_dbschema	Function for retrieval of the schema of .sqlite database
MeSH_dbInfo	Function for retrieval of the information of .sqlite database

4 Data

Following 36 data are provided by *MeSH.db*.

MeSHMAPCOUNTS	Summary of all table
MeSHTERM	MeSH ID, MeSH Term, and MeSH Category
MeSHSYNONYM	The synonym of MeSH Term
MeSHQUALIFIER	Optional Information of MeSH Term (Subheadings)
MeSHAAOR	Ancestor-Offspring Relationships in A category
MeSHBAOR	Ancestor-Offspring Relationships in B category
MeSHCAOR	Ancestor-Offspring Relationships in C category
MeSHDAOR	Ancestor-Offspring Relationships in D category
MeSHEAOR	Ancestor-Offspring Relationships in E category
MeSHFAOR	Ancestor-Offspring Relationships in F category
MeSHGAOR	Ancestor-Offspring Relationships in G category
MeSHHAOR	Ancestor-Offspring Relationships in H category
MeSHIAOR	Ancestor-Offspring Relationships in I category
MeSHJAOR	Ancestor-Offspring Relationships in J category
MeSHKAOR	Ancestor-Offspring Relationships in K category
MeSHLAOR	Ancestor-Offspring Relationships in L category
MeSHMAOR	Ancestor-Offspring Relationships in M category
MeSHNAOR	Ancestor-Offspring Relationships in N category
MeSHVAOR	Ancestor-Offspring Relationships in V category
MeSHZAOR	Ancestor-Offspring Relationships in Z category
MeSHAPCR	Parent-Child Relationships in A category
MeSHBPCR	Parent-Child Relationships in B category
MeSHCPCR	Parent-Child Relationships in C category
MeSHDPCR	Parent-Child Relationships in D category
MeSHEPCR	Parent-Child Relationships in E category
MeSHFPCR	Parent-Child Relationships in F category
MeSHGPCR	Parent-Child Relationships in G category
MeSHHPCR	Parent-Child Relationships in H category
MeSHIPCR	Parent-Child Relationships in I category
MeSHJPCR	Parent-Child Relationships in J category
MeSHKPCR	Parent-Child Relationships in K category
MeSHLPCR	Parent-Child Relationships in L category
MeSHMPCR	Parent-Child Relationships in M category
MeSHNPCR	Parent-Child Relationships in N category
MeSHVPCR	Parent-Child Relationships in V category
MeSHZPCR	Parent-Child Relationships in Z category

In *MeSH.db*, all data are extracted by 4 functions defined by *AnnotationForge*; **keytypes**, **columns**, **keys** and **select**. **keys** function has 1 parameter (keytype) and **select** function also has 3 parameters (keys, columns and keytype). **columns** is the columns which you can retrieved by **select** and **keytypes** is the columns which you can specify as the parameter in **keys** and **select** functions.

Object Name	columns	keytype
MeSHMAPCOUNTS	MAPNAME, COUNT	MAPNAME
MeSHTERM	MESHID, MESHTERM, CATEGORY	MESHID, MESHTERM, CATEGORY
MeSHSYNONYM	MESHID, MESHSYNONYM	MESHID
MeSHQUALIFIER	QUALIFIERID, SUBHEADING, MESHID	QUALIFIERID, MESHID
MeSHAAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHBAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHCAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHDAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHEAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHFAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHGAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHHAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHIAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHJAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHKAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHLAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID

MeSHMAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHNAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHVAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHZAOR	ANCESTERMESHID, OFFSPRINGMESHID	ANCESTERMESHID, OFFSPRINGMESHID
MeSHAPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHBPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHCPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHDPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHEPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHFPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHGPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHHPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHIPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHJPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHKPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHLPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHMPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHNPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHVPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID
MeSHZPCR	PARENTMESHID, CHILDMESHID	PARENTMESHID, CHILDMESHID

5 Examples

5.1 Exercises in columns, keytypes, keys and select

MeSH.db uses **columns**, **keytypes**, **keys** and **select** functions defined by *AnnotationForge*. In this section we will show you how to use these functions in *MeSH.db*.

At first, install and load the *MeSH.db*.

```
> library(MeSH.db)
```

ls shows all object in this package.

```
> ls("package:MeSH.db")
```

```
 [1] "MeSH"          "MeSH_dbconn"  "MeSH_dbfile"   "MeSH_dbInfo"
 [5] "MeSH_dbschema" "MeSHAAOR"     "MeSHAPCR"      "MeSHBAOR"
 [9] "MeSHBPCR"      "MeSHCAOR"     "MeSHCPCR"      "MeSHDAOR"
[13] "MeSHDPCR"      "MeSHEAOR"     "MeSHEPCR"      "MeSHFAOR"
[17] "MeSHFPCR"      "MeSHGAOR"     "MeSHGPCR"      "MeSHHAOR"
[21] "MeSHHPCR"      "MeSHIAOR"     "MeSHIPCR"      "MeSHJAOR"
[25] "MeSHJPCR"      "MeSHKAOR"     "MeSHKPCR"      "MeSHLAOR"
[29] "MeSHLPCR"      "MeSHMAOR"     "MeSHMAPCOUNTS" "MeSHMPCR"
[33] "MeSHNAOR"      "MeSHNPCR"     "MeSHQUALIFIER" "MeSHSYNONYM"
[37] "MeSHTERM"      "MeSHVAOR"     "MeSHVPCR"      "MeSHZAOR"
[41] "MeSHZPCR"
```

Here we will use **columns**, **keytypes**, **keys** and **select** against MeSHMAPCOUNTS.

columns returns the rows which we can retrieve in MeSHMAPCOUNTS.

```
> columns(MeSHMAPCOUNTS)
```

```
[1] "MAPNAME" "COUNT"
```

keytypes returns the rows which we can use as the optional parameter in **keys** and **select** functions against MeSHMAPCOUNTS.

```
> keytypes(MeSHMAPCOUNTS)
```

```
[1] "MAPNAME"
```

MAPNAME is accessible.

keys function specifies the value of keytype.

```
> k <- keys(MeSHMAPCOUNTS, keytype = "MAPNAME")
> head(k)

         MAPNAME
1        MeSHTERM
2     MeSHSYNONYM
3 MeSHQUALIFIER
4        MeSHAAOR
5        MeSHBAOR
6        MeSHCAOR
```

select function specifies the rows in particular columns, which are having user-defined keys. This function retrieves data as single dataframe like SQL's SELECT statement. Now we will retrieve the rows in which MAPNAME is equivalent to "MeSHTERM".

```
> select(MeSHMAPCOUNTS, keys = k[1, ], columns = c("MAPNAME", "COUNT"),
+     keytype = "MAPNAME")

   MAPNAME COUNT
1 MeSHTERM 28921
```

By the way, in this situation, keytype-parameter does not have to be specified against MeSHMAPCOUNTS. Because MeSHMAPCOUNTS has only single column which is possible to be keytype and keytype is consequently specified.

```
> select(MeSHMAPCOUNTS, keys = k[1, ], columns = c("MAPNAME", "COUNT"))

   MAPNAME COUNT
1 MeSHTERM 28921
```

The same can be said of MeSHSYNONYM.

5.2 Annotation of *Leukemia*

Next we will annotate *Leukemia* by MeSH.

```
> columns(MeSHTERM)
```

```
[1] "MESHID"       "MESHTERM"     "MESHCATEGORY"
```

MESHID, MESHTERM and MESHCATEGORY can be retrieved from MeSHTERM.

```
> keytypes(MeSHTERM)
```

```
[1] "MESHID"       "MESHTERM"     "MESHCATEGORY"
```

All of them are available as keytype-parameters in **select** function.

select function retrieves the rows in which MESHTERM is ”*Leukemia*” in MeSHTERM table.

```
> LEU <- select(MeSHTERM, keys = "Leukemia", columns = c("MESHID",
+     "MESHTERM", "MESHCATEGORY"), keytype = "MESHTERM")
> LEU
```

```
   MESHID MESHTERM MESHCATEGORY
1 D007938 Leukemia            C
```

select function shows that MESHID of *Leukemia* is D007938 and *Leukemia* is in C (Diseases) category.

Using MeSHSYNONYM, we can also check whether *Leukemia* has some synonyms.

```
> select(MeSHSYNONYM, keys = LEU[1, 1], columns = c("MESHID", "MESHSYNONYM"),
+     keytype = "MESHTERM")
```

```
  MESHID MESHSYNONYM
1 D007938 Leucocythaemias
2 D007938 Leucocythaemia|T191|NON|EQV|NLM (2012)|110224|abcdef
3 D007938 Leucocythemias
4 D007938 Leucocythemia|T191|NON|EQV|NLM (2012)|110224|abcdef
5 D007938 Leukemias
```

Leukemia has some synonyms like *Leucocythaemia*, *Leucocythaemias*, *Leucocythemias* and *Leukemias*.

NLM also defines Qualifier terms, which means more rough annotation (subheadings) than MeSH. We can also use **select** function against MeSHQUALIFIER.

```
> select(MeSHQUALIFIER, keys = LEU[1, 1], columns = c("QUALIFIERID",
+      "SUBHEADING", "MESHID"), keytype = "MESHID")

   QUALIFIERID           SUBHEADING  MESHID
1      Q000097                blood D007938
2      Q000134  cerebrospinal fluid D007938
3      Q000139    chemically induced D007938
4      Q000145        classification D007938
5      Q000150         complications D007938
6      Q000151            congenital D007938
7      Q000175             diagnosis D007938
8      Q000178          diet therapy D007938
9      Q000188          drug therapy D007938
10     Q000191             economics D007938
11     Q000196            embryology D007938
12     Q000201            enzymology D007938
13     Q000208             ethnology D007938
14     Q000209              etiology D007938
15     Q000235              genetics D007938
16     Q000266               history D007938
17     Q000276            immunology D007938
18     Q000378            metabolism D007938
19     Q000382          microbiology D007938
20     Q000401             mortality D007938
21     Q000451               nursing D007938
22     Q000453          epidemiology D007938
23     Q000469           parasitology D007938
24     Q000473              pathology D007938
25     Q000503         physiopathology D007938
26     Q000517   prevention & control D007938
27     Q000523             psychology D007938
28     Q000530            radiography D007938
29     Q000531  radionuclide imaging D007938
30     Q000532           radiotherapy D007938
31     Q000534         rehabilitation D007938
32     Q000601                surgery D007938
33     Q000628                therapy D007938
34     Q000652                  urine D007938
35     Q000662             veterinary D007938
36     Q000736         ultrasonography D007938
37     Q000821               virology D007938
```

As mentioned in section 1, MeSH has hierarchical structures. AOR provides us upper (or lower) hierarchical MeSH Term. We already know *Leukemia* is categorized in C (Diseases), so MeSHCAOR is available.

```
> ANC <- select(MeSHCAOR, keys = LEU[1, 1], columns = c("ANCESTORMESHID",
+     "OFFSPRINGMESHID"), keytype = "OFFSPRINGMESHID")
> ANC

  ANCESTORMESHID OFFSPRINGMESHID
1        D009370         D007938
```

D009370 has found above *Leukemia*.

These MeSH ID can be translated to MeSH Term.

```
> select(MeSHTERM, keys = ANC[, 1], columns = c("MESHTERM"), keytype = "MESHID")

                    MESHTERM
1 Neoplasms by Histologic Type
```

Once keytype-parameter set to opposite direction (OFFSPRINGMESHID to ANCESTORMESHID), other MeSH IDs in lower hierarchies also can be retrieved.

```
> OFF <- select(MeSHCAOR, keys = LEU[1, 1], columns = c("ANCESTORMESHID",
+     "OFFSPRINGMESHID"), keytype = "ANCESTORMESHID")
> OFF

   ANCESTORMESHID OFFSPRINGMESHID
1         D007938         D001353
2         D007938         D001752
3         D007938         D004915
4         D007938         D007939
5         D007938         D007940
6         D007938         D007941
7         D007938         D007942
8         D007938         D007943
9         D007938         D007945
10        D007938         D007946
11        D007938         D007947
12        D007938         D007948
13        D007938         D007951
14        D007938         D007952
15        D007938         D007953
16        D007938         D015448
17        D007938         D015451
18        D007938         D015452
19        D007938         D015456
20        D007938         D015458
21        D007938         D015459
22        D007938         D015461
23        D007938         D015463
```

```
24          D007938          D015464
25          D007938          D015465
26          D007938          D015466
27          D007938          D015470
28          D007938          D015471
29          D007938          D015472
30          D007938          D015473
31          D007938          D015477
32          D007938          D015479
33          D007938          D016582
34          D007938          D016583
35          D007938          D023981
36          D007938          D054066
37          D007938          D054198
38          D007938          D054218
39          D007938          D054403
40          D007938          D054429
41          D007938          D054438
```

There are a lot of MeSH ID, which means *Leukemia* has many lower hierarchies.

PCR provides directly lower (or upper) hierarchies.

```
> CHI <- select(MeSHCPCR, keys = LEU[1, 1], columns = c("PARENTMESHID",
+     "CHILDMESHID"), keytype = "PARENTMESHID")
> CHI

  PARENTMESHID CHILDMESHID
1      D007938     D007942
2      D007938     D007943
3      D007938     D007945
4      D007938     D007946
5      D007938     D007951
6      D007938     D007952
7      D007938     D007953
8      D007938     D016582
9      D007938     D016583
```

These MeSH IDs can be translated to MeSH Terms.

```
> select(MeSHTERM, keys = CHI[, 2], columns = c("MESHTERM"), keytype = "MESHID")

                    MESHTERM
1       Leukemia, Experimental
3         Leukemia, Hairy Cell
4           Leukemia, Lymphoid
5         Leukemia, Mast-Cell
```

14

```
6            Leukemia, Myeloid
7          Leukemia, Plasma Cell
8   Leukemia, Radiation-Induced
9             Leukemia, Feline
10      Enzootic Bovine Leukosis
```

Leukemia has a lot of subtypes like $Leukemia, Myeloid$, $Leukemia, Plasma Cell$ and so on.

6 Setup

This vignette was built on:

```
> sessionInfo()
```

```
R Under development (unstable) (2013-09-25 r63985)
Platform: x86_64-apple-darwin10.8.0 (64-bit)

locale:
[1] ja_JP.UTF-8/ja_JP.UTF-8/ja_JP.UTF-8/C/ja_JP.UTF-8/ja_JP.UTF-8

attached base packages:
[1] parallel  stats     graphics  grDevices utils     datasets  methods
[8] base

other attached packages:
[1] MeSH.db_0.99.0        AnnotationForge_1.3.26 org.Hs.eg.db_2.9.0
[4] RSQLite_0.11.4        DBI_0.2-7              AnnotationDbi_1.23.23
[7] Biobase_2.21.7        BiocGenerics_0.7.5

loaded via a namespace (and not attached):
[1] IRanges_1.19.38 stats4_3.1.0    tools_3.1.0
```

References

[1] S. J. Nelson and et al. The MeSH translation maintenance system: structure, interface design, and implementation. *Stud. Health Technol. Inform.*, 107: 67-69, 2004.

[2] M. Ashburner and et al. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat. Genet.*, 25(1): 25-29, 2000.

[3] T. Nakazato and et al. BioCompass: a novel functional inference tool that utilizes MeSH hierarchy to analyze groups of genes. *In Silico Biol.*, 8(1): 53-61, 2007.

[4] T. Nakazato and et al. Nucleic Acids Res. *Gendoo: functional profiling of gene and disease features using MeSH vocabulary.*, 37: W166-W169, 2009.

[5] D. J. Saurin and et al. GeneMeSH: a web-based microarray analysis tool for relating differentially expressed genes to MeSH terms. *BMC Bioinformatics*, 11: 166, 2010.

[6] M. A. Sartor and et al. Metab2MeSH: annotating compounds with medical subject headings. *Bioinformatics*, 28(10): 1408-1410, 2012.