

# int.mppi.db

February 3, 2010

---

`int.mppi_dbconn`     *Collect information about the package annotation DB*

---

## Description

Some convenience functions for getting a connection object to (or collecting information about) the package annotation DB.

## Usage

```
int.mppi_dbconn()
int.mppi_dbfile()
int.mppi_dbschema(file="", show.indices=FALSE)
int.mppi_dbInfo()
```

## Arguments

`file`            A connection, or a character string naming the file to print to (see the file argument of the `cat` function for the details).

`show.indices`   The CREATE INDEX statements are not shown by default. Use `show.indices=TRUE` to get them.

## Details

`int.mppi_dbconn` returns a connection object to the package annotation DB. **IMPORTANT:** Don't call `dbDisconnect` on the connection object returned by `int.mppi_dbconn` or you will break all the `AnnDbObj` objects defined in this package!

`int.mppi_dbfile` returns the path (character string) to the package annotation DB (this is an SQLite file).

`int.mppi_dbschema` prints the schema definition of the package annotation DB.

`int.mppi_dbInfo` prints other information about the package annotation DB.

**Examples**

```
## Show the first three rows.
dbGetQuery(int.mppi_dbconn(), "select * from mppi limit 3")

## The connection object returned by int.mppi_dbconn() was created with:
dbConnect(SQLite(), dbname=int.mppi_dbfile(), cache_size=64000, synchronous=0)

int.mppi_dbschema()

int.mppi_dbInfo()
```

---

```
int.mppi.db          annotation data package
```

---

**Description**

Welcome to the int.mppi.db annotation Package. The annotation package was built using a downloadable R package - PAnnBuilder (download and build your own). The purpose is to provide detailed information about the mammalian protein-protein interactions from MPPI database: <http://mips.gsf.de/proj/ppi/data/mppi.gz>

Each of these objects has their own manual page detailing where relevant data was obtained along with examples of how to use it. Many of these objects also have a reverse map available. When this is true, expect to usually find relevant information on the same manual page as the forward map.

**Examples**

```
# You can learn what objects this package supports with the following command:
ls("package:int.mppi.db")
```

---

```
int.mppiMAPCOUNTS  Number of mapped keys for the maps in package int.mppi.db
```

---

**Description**

int.mppiMAPCOUNTS provides the "map count" (i.e. the count of mapped keys) for each map in package int.mppi.db.

**Details**

This "map count" information is precalculated and stored in the package annotation DB. This allows some quality control and is used by the [checkMAPCOUNTS](#) function defined in AnnotationDbi to compare and validate different methods (like `count.mappedkeys(x)` or `sum(!is.na(as.list(x)))`) for getting the "map count" of a given map.

**See Also**

[mappedkeys](#), [count.mappedkeys](#), [checkMAPCOUNTS](#)

**Examples**

```
int.mppiMAPCOUNTS
mapnames <- names(int.mppiMAPCOUNTS)
int.mppiMAPCOUNTS[mapnames[1]]
x <- get(mapnames[1])
sum(!is.na(as.list(x)))
count.mappedkeys(x) # much faster!

## Check the "map count" of all the maps in package int.mppi.db
checkMAPCOUNTS("int.mppi.db")
```

---

int.mppiPPI	<i>Protein-Protein interaction</i>
-------------	------------------------------------

---

**Description**

int.mppiPPI gives protein-protein interactions from MPPI database.

**Details**

Protein is given as Uniprot protein accession number.

Mappings were based on data provided by: MPPI (<http://mips.gsf.de/proj/ppi/data/mppi.gz>) on

**Examples**

```
# Convert to a list
xx <- as.list(int.mppiPPI)
if(length(xx) > 0){
  # Gets the first five interaction list
  xx[1:5]
}
```

# Index

## \*Topic **datasets**

`int.mppiMAPCOUNTS`, 2

`AnnDbObj`, 1

`checkMAPCOUNTS`, 2

`count.mappedkeys`, 2

`dbDisconnect`, 1

`int.mppi.db`, 2

`int.mppi_dbconn`, 1

`int.mppi_dbfile`  
(`int.mppi_dbconn`), 1

`int.mppi_dbInfo`  
(`int.mppi_dbconn`), 1

`int.mppi_dbschema`  
(`int.mppi_dbconn`), 1

`int.mppiMAPCOUNTS`, 2

`int.mppiPPI`, 3

`mappedkeys`, 2