Package: PhysioAnnotationHub (via r-universe)

May 16, 2026

Type Package

Title Anatomical and Clinical Knowledge Graph for Physiological Data

Version 0.1.0

Description Centralized annotation hub providing anatomical ontology, muscle/bone metadata, nerve innervation, clinical codes (ICD-10, ICF), and knowledge graph queries for the physio-ecosystem.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Depends R (>= 4.1.0)

Suggests testthat (>= 3.0.0), PhysioMSKNet

Config/testthat/edition 3

Repository https://x-biosignal.r-universe.dev

Date/Publication 2026-03-16 11:31:52 UTC

RemoteUrl https://github.com/x-biosignal/PhysioAnnotationHub

RemoteRef HEAD

RemoteSha 722b239b236e317cddc650a3f11619b2b7e31a84

Contents

`getBoneAnnotation` *Get bone annotations*

Description

Get bone annotations

Usage

```
getBoneAnnotation(bones = NULL, hub = NULL, fuzzy = TRUE)
```

Arguments

`bones`	Character vector of bone names (NULL for all)
`hub`	AnnotationHub object
`fuzzy`	Logical; use fuzzy matching (default TRUE)

Value

data.frame of bone annotations

`getClinicalCodes` *Get clinical codes for muscles*

Description

Get clinical codes for muscles

Usage

```
getClinicalCodes(muscles, system = c("icd10", "icf"), hub = NULL)
```

Arguments

`muscles`	Character vector of muscle names
`system`	Character; "icd10" or "icf" (default "icd10")
`hub`	AnnotationHub object

Value

data.frame of matching clinical codes

getMuscleAnnotation *Get muscle annotations*

Description

Get muscle annotations

Usage

```
getMuscleAnnotation(muscles = NULL, hub = NULL, fuzzy = TRUE)
```

Arguments

muscles	Character vector of muscle names (NULL for all)
hub	AnnotationHub object
fuzzy	Logical; use fuzzy matching (default TRUE)

Value

data.frame of muscle annotations

getNerveAnnotation *Get nerve annotations*

Description

Get nerve annotations

Usage

```
getNerveAnnotation(nerves = NULL, hub = NULL)
```

Arguments

nerves	Character vector of nerve names (NULL for all)
hub	AnnotationHub object

Value

data.frame of nerve annotations

kgEnrichment *Functional enrichment analysis for a set of muscles*

Description

Tests whether a set of muscles is enriched for specific annotations (actions, innervation, body regions) compared to the full set.

Usage

```
kgEnrichment(
  muscles,
  annotation_type = c("action", "nerve", "body_region", "spinal_level"),
  hub = NULL
)
```

Arguments

muscles	Character vector of muscle names (query set)
annotation_type	Character; "action", "nerve", "body_region", "spinal_level"
hub	AnnotationHub object

Value

data.frame with term, count, expected, fold_enrichment, p_value

kgNeighbors *Get KG neighbors of an entity*

Description

Get KG neighbors of an entity

Usage

```
kgNeighbors(entity, depth = 1, hub = NULL)
```

Arguments

entity	Character; entity name
depth	Integer; traversal depth (default 1)
hub	AnnotationHub object

Value

data.frame of all triples within depth hops

kgShortestPath *Find shortest path between two entities in the KG*

Description

Find shortest path between two entities in the KG

Usage

```
kgShortestPath(from, to, hub = NULL, max_depth = 5)
```

Arguments

from	Character; source entity name
to	Character; target entity name
hub	AnnotationHub object
max_depth	Integer; maximum search depth (default 5)

Value

List with path (entities), predicates, and depth

loadAnnotationHub *Load the PhysioAnnotationHub*

Description

Loads all annotation data into a cached environment for fast repeated queries.

Usage

```
loadAnnotationHub(reload = FALSE)
```

Arguments

reload	Logical; force reload even if cached (default FALSE)

Value

An AnnotationHub object (list) with muscle, bone, nerve, kg, clinical data

print.PhysioAnnotationHub *Print method for PhysioAnnotationHub*

Description

Print method for PhysioAnnotationHub

Usage

```
## S3 method for class 'PhysioAnnotationHub'
print(x, ...)
```

Arguments

x	A PhysioAnnotationHub object
...	Additional arguments (ignored)

Value

Invisibly returns x

queryKG *Query the Knowledge Graph by triple pattern*

Description

Query the Knowledge Graph by triple pattern

Usage

```
queryKG(subject = NULL, predicate = NULL, object = NULL, hub = NULL, exact = FALSE)
```

Arguments

subject	Character or NULL; entity name pattern (grep)
predicate	Character or NULL; relation type (exact or grep)
object	Character or NULL; target entity pattern (grep)
hub	AnnotationHub object (loaded if NULL)
exact	Logical; use exact matching instead of grep (default FALSE)

Value

data.frame of matching triples

Index