

Package: remart (via r-universe)

October 12, 2026

Title Reimplementation of the 'biomaRt' Package Using Modern Ensembl APIs

Version 0.99.6

Description This package aims to offer a drop-in replacement to biomaRt using the same interface but the more modern and better maintained APIs provided by Ensembl. It provides an easy solution for compatibility with minimal changes for existing projects and packages. New projects may prefer to use another approach which is not constrained by the need for backward compatibility with biomaRt.

License MIT + file LICENSE

URL <https://huber-group-embl.github.io/remart/>,
<https://github.com/Huber-group-EMBL/remart>

BugReports <https://github.com/Huber-group-EMBL/remart/issues>

Depends R (>= 4.2.0)

Imports httr2, stats

Suggests BiocStyle, curl, httptest2, knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/roxygen2/version 8.1.0

Config/testthat/edition 3

Encoding UTF-8

Roxygen list(markdown = TRUE)

biocViews Annotation, DataImport

Config/pak/sysreqs libssl-dev

Repository <https://biocstaging.r-universe.dev>

Date/Publication 2026-10-12 15:04:39 UTC

RemoteUrl <https://github.com/BiocStaging/remart>

RemoteRef HEAD

RemoteSha 54993a4995baa8a11d48252376b2ce1a7adbba8

Contents

getBM	2
getGene	4
getLDS	5
getSequence	7
listAttributes	8

Index	10
--------------	-----------

getBM	<i>Retrieve gene/transcript annotations from Ensembl</i>
-------	--

Description

Retrieve gene/transcript annotations from Ensembl

Usage

```
getBM(  
  attributes,  
  filters = "",  
  values = "",  
  ...,  
  version = NULL,  
  species = NULL  
)
```

Arguments

attributes	Attributes you want to retrieve. A possible list of attributes can be retrieved using the function listAttributes() .
filters	Filters (one or more) that should be used in the query. A possible list of filters can be retrieved using the function listFilters() .
values	Values of the filter, e.g. vector of affy IDs. If multiple filters are specified then the argument should be a list of vectors of which the position of each vector corresponds to the position of the filters in the filters argument.
...	Ignored. Used to catch no longer necessary parameters such as mart, checkFilters, verbose, uniqueRows, bmHeader, quote and useCache from biomaRt functions.
version	Ensembl version to use for the query. If NULL, the latest version will be used.
species	Ensembl name (e.g. "mouse" or "mus_musculus") of the species to look Ensembl IDs for if, e.g., external_gene_name is provided in filters. In biomaRt , this was inferred from the mart argument, but since this argument is no longer used, the species must be provided explicitly.

Details

Only a subset of the attributes and filters supported by the biomaRt package are currently implemented, as this data has to be retrieved through the Ensembl REST lookup/id endpoint rather than a generic BioMart query engine.

Supported filters (only one can be used at a time): `ensembl_gene_id`, `ensembl_transcript_id`, `external_gene_name`.

Supported attributes: `ensembl_gene_id`, `ensembl_transcript_id`, `ensembl_peptide_id`, `external_gene_name`, `description`, `chromosome_name`, `start_position`, `end_position`, `strand`, `gene_biotype`, `transcript_biotype`, `version`.

Value

A data frame containing the requested gene or transcript annotations, with one column for each requested attribute. The number of rows does not necessarily match the number of requested identifiers, as some attributes can have multiple values for a single identifier (e.g. a gene with multiple transcripts).

Examples

```
attribs <- c("ensembl_gene_id", "external_gene_name", "chromosome_name")
getBM(
  attributes = attribs,
  filters = "ensembl_gene_id",
  values = c("ENSG00000157764", "ENSG0000004939")
)

# It is also possible to a gene symbol, but a species must be specified, as
# gene symbols are not unique across species.
getBM(
  attributes = attribs,
  filters = "external_gene_name",
  values = c("APOE", "MAPT"),
  species = "human"
)

# Query a specific Ensembl release (e.g. 114) instead of the latest one.
getBM(
  attributes = attribs,
  filters = "ensembl_gene_id",
  values = "ENSG00000157764",
  version = 114
)

# Archived releases can also be specified by their release name.
getBM(
  attributes = attribs,
  filters = "external_gene_name",
  values = "APOE",
  species = "human",
  version = "may2025"
```

)

getGene

*Retries gene annotation information from Ensembl.***Description**

Retries gene annotation information from Ensembl.

Usage

```
getGene(id, type = "ensembl_gene_id", ..., version = NULL, species = NULL)
```

Arguments

id	vector of gene identifiers one wants to annotate
type	type of identifier, possible values can be obtained by the listFilters function. Examples are entrezgene_id, hgnc_symbol (for hugo gene symbol), ensembl_gene_id, unigene, agilentprobe, affy_hg_u133_plus_2, refseq_dna, etc.
...	Ignored. Used to catch no longer necessary parameters such as mart from biomaRt functions.
version	Ensembl version to use for the query. If NULL, the latest version will be used.
species	Ensembl name (e.g. "mouse" or "mus_musculus") of the species to look Ensembl IDs for if, e.g., external_gene_name is provided in filters. In biomaRt , this was inferred from the mart argument, but since this argument is no longer used, the species must be provided explicitly.

Value

A data frame containing the following gene annotations for the requested IDs:

- ensembl_gene_id
- hgnc_symbol
- description
- chromosome_name
- band (not available from Ensembl REST API, will be filled with NA)
- strand
- start_position
- end_position

Examples

```

getGene(
  "ENSG00000157764",
  type = "ensembl_gene_id"
)

ids <- c(
  "ENSG0000003987",
  "ENSG0000004939"
)
getGene(
  id = ids,
  type = "ensembl_gene_id",
  version = "may2025"
)

```

getLDS

*Retrieve orthologous gene annotations across two species***Description**

Retrieve orthologous gene annotations across two species

Usage

```

getLDS(
  attributes,
  filters = "",
  values = "",
  attributesL,
  filtersL = "",
  valuesL = "",
  ...,
  version = NULL,
  species = NULL,
  speciesL = NULL
)

```

Arguments

attributes	Attributes you want to retrieve of primary dataset. A possible list of attributes can be retrieved using the function listAttributes() .
filters	Filters that should be used in the query. These filters will be applied to primary dataset. A possible list of filters can be retrieved using the function listFilters() .
values	Values of the filter, e.g. list of affy IDs
attributesL	Attributes of linked dataset that needs to be retrieved

<code>filtersL</code>	Filters to be applied to the linked dataset
<code>valuesL</code>	Values for the linked dataset filters
<code>...</code>	Ignored. Used to catch no longer necessary parameters such as <code>mart</code> , <code>martL</code> , <code>verbose</code> , <code>uniqueRows</code> and <code>bmHeader</code> from biomaRt functions.
<code>version</code>	Ensembl version to use for the query. If <code>NULL</code> , the latest version will be used.
<code>species</code>	Ensembl name (e.g. "mouse" or "mus_musculus") of the species to look Ensembl IDs for if, e.g., <code>external_gene_name</code> is provided in <code>filters</code> . In biomaRt , this was inferred from the <code>mart</code> argument, but since this argument is no longer used, the species must be provided explicitly.
<code>speciesL</code>	Ensembl name (e.g. "mouse" or "mus_musculus") of the species to look up orthologues in. This replaces the <code>martL</code> argument used in biomaRt , as connections to Ensembl datasets are no longer needed, but the species of the linked dataset can not be inferred from <code>attributesL/filtersL/valuesL</code> alone.

Details

This function relies on the Ensembl REST `homology/id` endpoint, and therefore only supports retrieving orthologues, and not the more general cross-database attribute linking `biomaRt::getLDS()` could perform.

Supported filters/attributes (for both `filters/attributes` and `filtersL/attributesL`): `ensembl_gene_id`, `ensembl_peptide_id`, `external_gene_name`, `description`, `chromosome_name`, `start_position`, `end_position`, `strand`, `gene_biotype`.

Only "ensembl_gene_id" and "external_gene_name" are supported for filter.

Value

A data frame containing source-species and target-species ortholog annotations.

Examples

```
getLDS(
  attributes = c("ensembl_gene_id", "external_gene_name"),
  filters = "ensembl_gene_id",
  values = "ENSG00000157764",
  attributesL = c("ensembl_gene_id", "external_gene_name"),
  speciesL = "mouse"
)

# It is also possible to a gene symbol, but a species must be specified, as
# gene symbols are not unique across species.
getLDS(
  attributes = c("ensembl_gene_id", "external_gene_name"),
  filters = "external_gene_name",
  values = "APOE",
  species = "human",
  attributesL = c("ensembl_gene_id", "external_gene_name"),
  speciesL = "mouse"
)
```

```
# Query a specific Ensembl release for the source and target genes.
getLDS(
  attributes = c("ensembl_gene_id", "external_gene_name"),
  filters = "ensembl_gene_id",
  values = "ENSG00000157764",
  attributesL = c("ensembl_gene_id", "external_gene_name"),
  speciesL = "mouse",
  version = 114
)
```

getSequence

Get Sequences from Ensembl

Description

Get Sequences from Ensembl

Usage

```
getSequence(
  chromosome,
  start,
  end,
  id,
  type,
  seqType,
  upstream,
  downstream,
  ...,
  version = NULL
)
```

Arguments

chromosome	Chromosome name
start	start position of sequence on chromosome
end	end position of sequence on chromosome
id	An identifier or vector of identifiers.
type	The type of identifier used. Supported types are hugo, ensembl, embl, entrez-gene, refseq, ensemblTrans and unigene. Alternatively one can also use a filter to specify the type. Possible filters are given by the listFilters() function.
seqType	Type of sequence that you want to retrieve. Allowed seqTypes are given in the details section.
upstream	To add the upstream sequence of a specified number of basepairs to the output.

downstream	To add the downstream sequence of a specified number of basepairs to the output.
...	Ignored. Used to catch no longer necessary parameters such as mart from biomaRt functions.
version	Ensembl version to use for the query. If NULL, the latest version will be used.

Value

A data frame containing the requested sequences and their associated metadata.

Examples

```
getSequence(  
  seqType = "gene_exon_intron",  
  type = "ensembl_gene_id",  
  id = "ENSG00000001497"  
)  
  
ids <- c(  
  "ENSG00000003987",  
  "ENSG00000004939"  
)  
getSequence(  
  seqType = "gene_exon_intron",  
  type = "ensembl_gene_id",  
  id = ids  
)  
  
# Retrieve sequences from a specific Ensembl release.  
getSequence(  
  seqType = "cdna",  
  type = "ensembl_gene_id",  
  id = "ENSG00000001497",  
  version = 114  
)
```

listAttributes	<i>List supported attributes</i>
----------------	----------------------------------

Description

List supported attributes

Usage

```
listAttributes(...)
```

Arguments

... Ignored. Used to catch no longer necessary parameters from the **biomaRt** functions.

Value

A character vector of supported attributes to be used in [getBM\(\)](#) and [getLDS\(\)](#).

Examples

```
listAttributes()
```

Index

getBM, [2](#)
getBM(), [9](#)
getGene, [4](#)
getLDS, [5](#)
getLDS(), [9](#)
getSequence, [7](#)

listAttributes, [8](#)
listAttributes(), [2](#), [5](#)
listFilters(), [2](#), [5](#), [7](#)