SOFTWARE TOOL ARTICLE

BED: a Biological Entity Dictionary based on a graph data model [version 1; referees: 1 approved with reservations]

Patrice Godard 1,2, Jonathan van Eyl 2

1Clarivate Analytics, Carlsbad, CA, 92008, USA
2UCB, Braine-l’Alleud, 1420, Belgium

Abstract
The understanding of molecular processes involved in a specific biological system can be significantly improved by combining and comparing different data set and knowledge resources. However these information sources often use different identification systems and an identifier conversion step is required before any integration effort. Mapping between identifiers is often provided by the reference information resources and several tools have been implemented to simplify their use. However these tools cannot be easily customized and optimized for any specific use. Also the information provided by different resources is not combined to increase the efficiency of the mapping process and deprecated identifiers from former version of databases are not taken into account. Finally finding automatically the most relevant path to map identifiers from one scope to the other is often not trivial. The Biological Entity Dictionary (BED) addresses these challenges by relying on a graph data model describing possible relationships between entities and their identifiers. This model has been implemented using Neo4j and an R package provides functions to query the graph but also to create and feed a custom instance of the database.

Corresponding author: Patrice Godard (patrice.godard@ucb.com)

Author roles: Godard P: Conceptualization, Methodology, Software, Validation, Writing – Original Draft Preparation, Writing – Review & Editing; van Eyl J: Supervision, Validation, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

How to cite this article: Godard P and van Eyl J. BED: a Biological Entity Dictionary based on a graph data model [version 1; referees: 1 approved with reservations] F1000Research 2018, 7:195 (doi: 10.12688/f1000research.13925.1)

Copyright: © 2018 Godard P and van Eyl J. This is an open access article distributed under the terms of the Creative Commons Attribution Licence, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Grant Information: This work was entirely supported by UCB Pharma. The authors declared that no grants were involved in supporting this work.

Introduction
Since the advent of genome sequencing projects, many technologies have been developed to get access to different molecular information on a large scale and with high throughput. DNA micro-arrays are probably the archetype of such technology because of their historical impact on gathering data related to nucleic acids: genomic DNA and RNA. They triggered the emergence of “omics” fields of research such as genomics, epigenomics or transcriptomics. Lately massive parallel sequencing further increased the throughput of data generation related to nucleic acids by several orders of magnitude. In a different way, mass spectrometry-related technologies allow the identification and the quantification of many kinds of molecular entities such as metabolites and proteins. Many information systems have been developed to manage the exploding amount of data and knowledge related to biological molecular entities. These resources manage different aspects of the data. For example some are genome or proteome centered, whereas others are focused on molecular interactions and pathways. Thus all these resources rely on different identifier systems to organize the concepts of interest. The value of all the experimental data and all the knowledge collected in public or private resources is very high as such but is also often synergistically leveraged by their cross comparison in a dedicated manner. Indeed many datasets can be relevant when addressing the understanding of a specific biological system, a phenotypic trait or a disease for example. These datasets can focus on different biological entities such as transcripts or proteins in different tissues, conditions or organisms. Comparing all these data and integrating them with available knowledge requires the ability to map the identifiers on which each resource relies.

To achieve this task public and proprietary information systems provide mapping tables between their own identifiers and those from other resources. Furthermore many tools have been developed to facilitate the access to this information. Ensembl BioMarts (Kinsella et al., 2011), mygene (Wu et al., 2015), and g:Profiler (Reimand et al., 2016a) are popular examples among many others. However, as pointed out by van Iersel et al. (2010), these tools are generally dedicated to a particular domain and not necessarily relevant or complete for all research projects, and keeping them up-to-date can also be an issue. Recognizing these challenges van Iersel et al. (2010) proposed the BridgeDb framework providing to bioinformatics developers a standard interface between tools and mapping services and also allowing the easy integration of custom data by a transitivity mechanism.

Here we present BED: a biological entity dictionary. BED has been developed to address three main challenges. The first one is related to the completeness of identifier mappings. Indeed direct mapping information provided by the different systems are not always complete and can be enriched by mappings provided by other resources. More interestingly direct mappings not identified by any of these resources can be indirectly inferred by using mappings to a third reference. For example, many human Ensembl gene identifiers are not directly mapped to any Entrez gene identifiers but such mapping can be inferred using respective mappings to HGNC identifiers. The second challenge is related to the mapping of deprecated identifiers. Indeed entity identifiers can change from one resource release to another. The identifier history is provided by some resources, such as Ensembl or the NCBI, but it is generally not used by mapping tools. The third challenge is related to the automation of the mapping process according to the relationships between the biological entities of interest. Indeed mapping between gene and protein identifier scopes should not be done the same way than two scopes of gene identifiers. Also converting identifiers from different organism should be possible using gene ortholog information.

To meet these challenges we designed a graph data model describing possible relationships between different biological entities and their identifiers. This data model has been implemented with the Neo4j® graph database (Neo4j inc, 2017) and conversion rules have been defined and coded in an R (R Core Team, 2017) package. We provide an instance of the BED database focused on human, mouse and rat organism but many functions are available to construct other instances tailored to other needs.

Methods
Data model
The BED (Biological Entity Dictionary) system relies on a data model inspired by the central dogma of molecular biology (Crick, 1970) and describing relationships between molecular concepts usually manipulated in the frame of genomics studies (Figure 1). A biological entity identifier (BEID) can identify either a Gene (GeneID), a Transcript (TranscriptID), a Peptide (PeptideID) or an Object (ObjectID). Object entities can correspond to complex concepts coded by any number of genes (i.e. a protein complex or a molecular function). BEID are extracted from public or private databases (BEDB). BEID can provide an Attribute related to each BEID. For example it can be the sequencing region provided by the Ensembl database (Zerbino et al., 2018) or the identifier status provided by Uniprot (The UniProt Consortium, 2017). BEID can have one or several associated names (BENames) and symbols (BESymbol). GeneID can have one or several homologs in other organisms belonging to the same GeneIDFamily. Many genomics platforms, such as micro-array, allow the identification of biological entity by using probes identified
BEID identifying the same biological entity are related through three different kinds of relationship according to the information available in the source databases, and to the decision made by the database administrator about how to use them. Two BEID which corresponds_to each other both identify the same biological entity. A BEID which is_associated_to or which is_replaced_by another BEID does not directly identify any biological entity: the link is always indirect through one or several other BEID. Therefore, by design a BEID which is_associated_to or which is_replaced_by another BEID can be related to several different biological entities. It is not the case for other BEID which identify one and only one biological entity. This set of possible relationship allows the indirect mapping of different identifiers not necessarily provided by any integrated resource.

In order to efficiently leverage an indirect path through these different relationships the data model has been implemented in a Neo4j® graph database (Neo4j inc, 2017).
Feeding the database

Two R (R Core Team, 2017) packages have been developed to feed and query the database. The first one, neo2R, provides low level functions to interact with Neo4j®. The second R package, BED, provides functions to feed and query the BED Neo4j® graph database according to the data model described above.

Many functions are provided within the package to build a tailored BED database instance. These functions are not exported in order not to mislead the user when querying the database (which is the expected most frequent usage of the system). An R markdown document showing how to build a BED database instance for human, mouse and rat organisms is provided within the package. It can be adapted to other organisms or needs.

Briefly these functions can be divided according to three main levels:

- The lowest level function is the `bedImport` function which loads a table in the Neo4j® database according to a Cypher® query.
- Functions of the second level allow loading identifiers and relationships tables ensuring the integrity of the data model.
- Highest level functions are helpers for loading information provided by some public resources in different specific formats.

Querying the database

The `BED` R package provides several functions to retrieve identifiers from different resources, and also to convert identifiers from one reference to another. These functions generate and call Cypher® queries on the Neo4j® database. Converting thousands of identifiers can take some time (generally a few seconds). Also such conversions are often recurrent and redundant. In order to improve the performance for such recurrent and redundant queries, a cache system has been implemented. The first time, the query is run on Neo4j® for all the relevant ID related to user input and the result is saved in a local file. Next time similar queries are requested, the system does not call Neo4j® but loads the cached results and filters it according to user input. By default the cache is flushed when the system detects inconsistencies with the BED database. It can also be manually flushed if needed.

Operation

Minimal system requirements for running BED and neo2R R packages:

- $R \geq 3.4$
- Operating system: Linux, macOS, Windows
- Memory $\geq 4$GB RAM

The graph database has been implemented with Neo4j® version 3 (Neo4j inc, 2017). The BED R package depends on the following packages available in the Comprehensive R Archive Network (CRAN):

- `visNetwork` (Almende et al., 2017)
- `dplyr` (Wickham et al., 2017)
- `htmltools` (RStudio inc, 2017)
- `DT` (Xie, 2016)
- `shiny` (Chang et al., 2017)
- `miniUI` (Cheng, 2016)
- `rstudioapi` (Allaire et al., 2017)

Use cases

Available database instance

An instance of the BED database (UCB-Human) has been built using the script provided in the BED R package and made available in a Docker® image (Docker inc, 2017) available here: https://hub.docker.com/r/patzaw/bed-ucb-human/
This instance used to exemplify the following use cases is focused on *Homo sapiens, Mus musculus* and *Rattus norvegicus* organisms and it has been built from the following resources:

- Ensembl (Zerbino et al., 2018)
- NCBI (NCBI Resource Coordinators, 2017)
- Uniprot (The UniProt Consortium, 2017)
- biomaRt (Durinck et al., 2009)
- GEOquery (Davis & Meltzer, 2007)
- Clarivate Analytics MetaBase® (Clarivate Analytics, 2017)

The numbers of biological entity (BE) identifiers (BEID) available in this BED database instance and which can be mapped to each other are shown in Table 1. In total, 3,519,181 BEID are available in this BED instance. This number includes deprecated identifiers without successor and which therefore cannot be mapped to any other identifier. All the genomics platforms included in this BED database instance are shown in Table 2. They provide mapping to BEID from 354,205 ProbeID in total.

### Table 1. Numbers of BEID available in the BED UCB-Human database instance.

Numbers have been split according to the BE type and the organism. Only BEID which can be mapped to each other are taken into account (e.g. excluding deprecated identifiers without successor).

<table>
<thead>
<tr>
<th>BE</th>
<th>Organism</th>
<th>Database</th>
<th>BEID</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gene</td>
<td>Homo sapiens</td>
<td>MIM_GENE</td>
<td>17,146</td>
<td><a href="http://www.omim.org">http://www.omim.org</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Homo sapiens</td>
<td>miRBase</td>
<td>1,881</td>
<td><a href="http://www.mirbase.org">http://www.mirbase.org</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Homo sapiens</td>
<td>Ens_gene</td>
<td>68,460</td>
<td><a href="http://www.ensembl.org">http://www.ensembl.org</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Homo sapiens</td>
<td>HGNC</td>
<td>41,195</td>
<td><a href="http://www.genenames.org">http://www.genenames.org</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Homo sapiens</td>
<td>Vega_gene</td>
<td>19,141</td>
<td><a href="http://vega.sanger.ac.uk">http://vega.sanger.ac.uk</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Mus musculus</td>
<td>miRBase</td>
<td>1,193</td>
<td><a href="http://www.mirbase.org">http://www.mirbase.org</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Mus musculus</td>
<td>Ens_gene</td>
<td>56,954</td>
<td><a href="http://www.ensembl.org">http://www.ensembl.org</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Mus musculus</td>
<td>MGI</td>
<td>78,547</td>
<td><a href="http://www.informatics.jax.org">http://www.informatics.jax.org</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Mus musculus</td>
<td>Vega_gene</td>
<td>45,237</td>
<td><a href="http://vega.sanger.ac.uk">http://vega.sanger.ac.uk</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Mus musculus</td>
<td>MetaBase_gene</td>
<td>20,628</td>
<td><a href="https://portal.genego.com">https://portal.genego.com</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Rattus norvegicus</td>
<td>miRBase</td>
<td>495</td>
<td><a href="http://www.mirbase.org">http://www.mirbase.org</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Rattus norvegicus</td>
<td>Ens_gene</td>
<td>34,963</td>
<td><a href="http://www.ensembl.org">http://www.ensembl.org</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Rattus norvegicus</td>
<td>RGD</td>
<td>46,976</td>
<td><a href="https://rgd.mcw.edu">https://rgd.mcw.edu</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Rattus norvegicus</td>
<td>Vega_gene</td>
<td>1,146</td>
<td><a href="http://vega.sanger.ac.uk">http://vega.sanger.ac.uk</a></td>
</tr>
<tr>
<td>Gene</td>
<td>Rattus norvegicus</td>
<td>MetaBase_gene</td>
<td>17,505</td>
<td><a href="https://portal.genego.com">https://portal.genego.com</a></td>
</tr>
<tr>
<td>Transcript</td>
<td>Homo sapiens</td>
<td>Ens_transcript</td>
<td>228,389</td>
<td><a href="http://www.ensembl.org">http://www.ensembl.org</a></td>
</tr>
<tr>
<td>Transcript</td>
<td>Homo sapiens</td>
<td>Vega_transcript</td>
<td>37,017</td>
<td><a href="http://vega.sanger.ac.uk">http://vega.sanger.ac.uk</a></td>
</tr>
<tr>
<td>Name</td>
<td>Description</td>
<td>BE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------</td>
<td>------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL6101</td>
<td>Illumina ratRef-12 v1.0 expression beadchip</td>
<td>Gene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL6947</td>
<td>Illumina HumanHT-12 V3.0 expression beadchip</td>
<td>Gene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL10558</td>
<td>Illumina HumanHT-12 V4.0 expression beadchip</td>
<td>Gene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL1355</td>
<td>[Rat230_2] Affymetrix Rat Genome 230 2.0 Array</td>
<td>Gene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL1261</td>
<td>[Mouse430_2] Affymetrix Mouse Genome 430 2.0 Array</td>
<td>Gene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL96</td>
<td>[HG-U133A] Affymetrix Human Genome U133A Array</td>
<td>Gene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL13158</td>
<td>[HT_HG-U133_Plus_PM] Affymetrix HT HG-U133+ PM Array Plate</td>
<td>Gene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL571</td>
<td>[HG-U133A_2] Affymetrix Human Genome U133A 2.0 Array</td>
<td>Gene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL570</td>
<td>[HG-U133_Plus_2] Affymetrix Human Genome U133 Plus 2.0 Array</td>
<td>Gene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL6480</td>
<td>Agilent-014850 Whole Human Genome Microarray 4x44K G4112F</td>
<td>Gene</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GPL6885</td>
<td>Illumina MouseRef-8 v2.0 expression beadchip</td>
<td>Transcript</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Exploring identifiers of biological entities

The `getBeIds` function returns all BE identifiers from a specific scope. A scope is defined by the type of BE or probe, the source of the identifiers (database or platform) and the organism. For example, the following code returns all the Ensembl identifiers of human genes.

```r
beids <- getBeIds(
  be="Gene", source="Ens_gene", organism="human",
  restricted=FALSE
)
head(beids)
```

```
##                    id preferred  Gene db.version db.deprecated
## 82643 ENSG00000283891      TRUE 64781         91         FALSE
## 82642 ENSG00000207766      TRUE 64783         91         FALSE
## 82645 ENSG00000276678      TRUE 64785         91         FALSE
## 82644 ENSG00000265993      TRUE 64787         91         FALSE
## 82647 ENSG00000283793      TRUE 64789         91         FALSE
## 82646 ENSG00000283621      TRUE 64791         91         FALSE
```

The `id` column corresponds to the BEID from the source of interest. The column named according to the BE type (in this case `Gene`) corresponds to the internal identifiers of the related BE. This internal identifier is not a stable reference that can be used as such. Nevertheless, it is useful to identify BEID identifying the same BE. In the example above even if most of `Gene` BE are identified by only one Ensembl gene BEID, many of them are identified by two or more (5,809 / 59,515 = 10%); 277 BE are even identified by more than 10 Ensembl BEID (Figure 2.a). In this case, most of these redundancies come from deprecated ID from former versions of the Ensembl database (version in use here: 91) and can be excluded by setting the `restricted` parameter to `TRUE` when calling the `getBeIds` function (Figure 2.b). However many BE are still identified by two or more current Ensembl BEID (2,715 / 59,515 = 5%). This result comes from the way the BED database is constructed: When two identifiers from the same resource correspond to the same identifier in another resource (correspond_to relationship in the data model), all these BEID are considered to identify the same BE.

A complex example of such mapping is shown in Figure 3 mapping all the BEID of the human TAS2R8 gene which codes for a protein of the family of candidate taste receptors. There are three identifiers corresponding to this gene.

![Figure 2](image.png)

**Figure 2.** Barplots showing the number of gene BE (log scale) identified by one or more Ensembl gene BEID. 
(a) All Ensembl gene ID. 
(b) Current Ensembl gene ID (version 91).
All these BEID are thus considered to identify the same gene. It turns out that the three Ensembl BEID correspond to the same gene mapped on different sequence version of the chromosome 12: the canonical (ENSG00000121314), CHR_HSCHR12_2_CTG2 (ENSG00000272712) and CHR_HSCHR12_3_CTG2 (ENSG00000277316). This information provided by Ensembl is encoded in the seq_region attribute for each Ensembl BEID (see data model) and is used to define preferred BEID which are mapped on canonical version of chromosome sequences. The ENSG00000272712 identifier shows also a complex history in former Ensembl versions.

Converting identifiers

The main goal of BED is to convert identifiers from one scope to another easily, rapidly and with high completeness. It has been thought in order to allow recurring comparisons to each other of many lists of biological entities from various origins.

The function guessIdOrigin can be used to guess the scope of any list of identifiers. A simple example regarding the conversion of human Ensembl gene to human Entrez gene identifiers is shown below and discussed hereafter.

```r
bedConv <- convBeIds(
  ids=beids$id, from="Gene", from.source="Ens_gene", from.org="human",
  to.source="EntrezGene", restricted=TRUE
)
```

Among all the 68,460 human Ensembl gene identifiers available in the database, 21,718 (32%) were not converted to any human Entrez gene identifier: 21,073 (33%) of the 64,661 non-deprecated and 645 (17%) of the 3,799 deprecated identifiers.

Three other tools were used on January 04, 2018 to perform the same conversion task: biomaRt (Durinck et al., 2009; Kinsella et al., 2011), mygene (Mark et al., 2014; Wu et al., 2013), and gProfileR (Reimand et al., 2016a; Reimand, 2016b). At that time, biomaRt and mygene were based on the Ensembl 91 release whereas gProfileR was based on release 90.

The numbers of human Ensembl gene identifiers successfully converted by each method are compared in Figure 4. Five identifiers were only converted by gProfileR. They were provided by former versions of Ensembl or NCBI but are now deprecated in the current releases of these two resources. All the other gene identifiers converted by the different methods were also converted by BED. However, BED was able to map at least 17,912 more identifiers than all the other tools (Figure 4.a). A few of these mappings (3,154) are explained by the fact that BED is the only tool mapping deprecated identifiers to current versions. Nevertheless, even when focusing on the mapping of current versions of
Ensembl identifiers BED was able to map 14,758 more identifiers than all the other tools (Figure 4.b). A few of these mappings (627) are directly provided by the NCBI. But most of them (14,131) are inferred from a mapping of the Ensembl and Entrez gene identifiers to the same HGNC (Gray et al., 2015) identifier.

A rough approximation of running times of the different methods is provided in Table 3. The aim of this table is to show that BED, as a dedicated and locally available tool, is a very efficient option to convert large lists of identifiers on the fly and recurrently. The aim of BED is to improve the efficiency of identifier conversion in a well defined context (organism, information resources of interest...) and not to replace biomaRt, mygene, gProfileR or other tools which provide many more features for many organisms and which should not be narrowed to this task for a complete comparison.

The BED convBeIds function can be used to convert identifiers from any available scope to any other one. It automatically find the most relevant path according to the considered biological entities. It allows elaborate mapping such as the conversion between probe identifiers from a platform focused on mouse transcripts into human protein identifiers. Because such mappings can be intricate, BED also provides a function to show the shortest relevant path between two different identifiers (Figure 5).

Additional features
Some additional use cases and examples are provided in the BED R package vignette. Several functions are available for annotating BEID with symbols and names, again taking advantage of information related to connected identifiers. Other functions are also provided to seek relevant identifiers of a specific biological entity. These functions are used by a shiny (Chang et al., 2017) gadget (Figure 6) providing an interactive dictionary of BEID which is also made available as an Rstudio add-in (Allaire et al., 2017; Cheng, 2016).

![Figure 4](image.png)

**Figure 4.** Venn diagrams showing the number of human Ensembl gene identifiers mapped to at least one human Entrez gene identifier by the different tested tools when focusing (a) on all 68,460 or (b) on current 64,661 BEID (Ensembl 91 release).

<table>
<thead>
<tr>
<th>Method</th>
<th>Running time</th>
</tr>
</thead>
<tbody>
<tr>
<td>BED (Not cached)</td>
<td>~9.9 secs</td>
</tr>
<tr>
<td>BED (Cached)</td>
<td>~2.5 secs</td>
</tr>
<tr>
<td>biomaRt</td>
<td>~40 secs</td>
</tr>
<tr>
<td>mygene</td>
<td>~3.9 mins</td>
</tr>
<tr>
<td>gProfileR</td>
<td>~1.2 mins</td>
</tr>
</tbody>
</table>

**Table 3.** Rough approximation of running time of different methods to convert human Ensembl gene identifiers in human Entrez gene identifiers.
Figure 5. BED conversion shortest path between the ILMN_1220595 probe identifier targeting a transcript of the mouse Il17a gene and the Uniprot Q16552 identifier of the human IL17 protein. The legend is shown to the left of the figure. The red arrow represents the is_homolog_of relationship. This graph has been drawn with the exploreConvPath function.

Figure 6. findBe Shiny gadget to seek relevant identifiers of a specific biological entity. In this example the user is looking after human Ensembl transcript identifiers corresponding to “il6”.

Conclusions
BED is a system dedicated to the mapping between identifiers of molecular biological entities. It relies on a graph data model implemented with Neo4j® and on rules coded in an R package. BED leverages mapping information provided by different resources in order to increase the mapping efficiency between each of them. It also allows the mapping of deprecated identifiers. Rules are used to automatically convert identifiers from one scope to another using the most appropriate path.
The intent of BED is to be tailored to specific needs, and beside functions for querying the system, the BED R package provides functions to build custom instances of the database. Database instances can be locally installed or shared across a community. This design combined with a cache system makes BED efficient for converting large lists of identifiers from and to a large variety of scopes.

Because of our research field we provide an instance focused on human, mouse and rat organisms. This database instance can be directly used in relevant projects but it can also be enriched depending on user or community needs.

Software availability
Latest source code is available at:
https://github.com/patzaw/BED
https://github.com/patzaw/neo2R

Archived source code as at time of publication:
https://zenodo.org/badge/latestdoi/119707445 (Godard, 2018a)
https://zenodo.org/badge/latestdoi/119698430 (Godard, 2018b)

Software is available to use under a GPL-3 license

Competing interests
No competing interests were disclosed.

Grant information
This work was entirely supported by UCB Pharma. The authors declared that no grants were involved in supporting this work.

Acknowledgments
We are grateful to Frédéric Vanclef, Malte Lucken, Liesbeth François, Matthew Page, Massimo de Francesco, and Marina Bessarabova for fruitful discussions and constructive criticisms.

References

Reference Source
Reference Source
Reference Source
Reference Source
Clarivate Analytics: MetaCore delivers high-quality biological systems content in context. 2017.
Reference Source
CRAN: The Comprehensive R Archive Network.
Reference Source
PubMed Abstract | Publisher Full Text
PubMed Abstract | Publisher Full Text
Reference Source
PubMed Abstract | Publisher Full Text
Data Source
Data Source
PubMed Abstract | Publisher Full Text | Free Full Text
PubMed Abstract | Publisher Full Text | Free Full Text
Publisher Full Text
PubMed Abstract | Publisher Full Text | Free Full Text
Reference Source

Reference Source

Reference Source

Reference Source

PubMed Abstract | Publisher Full Text | Free Full Text

Reference Source

PubMed Abstract | Publisher Full Text | Free Full Text

Reference Source

PubMed Abstract | Publisher Full Text | Free Full Text
Open Peer Review

Current Referee Status: ??

Version 1

Referee Report 05 March 2018
doi:10.5256/f1000research.15138.r31026

Denise Slenter, Martina M. Summer-Kutmon
Department of Bioinformatics - BiGCaT, Maastricht University, Maastricht, Netherlands

The article introduces BED a new identifier mapping tool. Using a graph database like Neo4j provides a fast way to query relationships between the biological entities and retrieve mappings of interest. The available source code is nicely documented and for bioinformaticians, setting up the database and running queries should be straight-forward.

Nevertheless, there are several major issues that we would like to comment on:

- Already in the abstract, it is indicated that current tools cannot be easily customized and optimized for any specific use. It is unclear what the authors actually mean with this statement and how this is solved through BED. Further on it is also stated that current tools are generally dedicated to a particular domain, which is also true for BED. BED only focuses on gene related identifies (genes, transcripts, proteins) similar to mygene, Ensembl BioMart and g:Profiler.

- In the introduction, three main challenges are mentioned which are addressed by BED.
  
  1. Integration of mappings from different resources - very relevant but the difficult question is if transitive mappings are always biological meaningful. They can also lead to conflicting statements when resources show inconsistent relationships (we have experienced this when comparing Ensembl → UniProt and UniProt → Ensembl mappings) - how are you dealing with that? We want to state that mygene is also integrating mappings from multiple resources.

  2. Mapping of deprecated identifiers - this is indeed an interesting problem when analysing older datasets and the visualization in Figure 3 can be very useful when running into such issues. While you mention that BED contains all deprecated identifiers, it is not discussed why g:Profiler has five deprecated identifiers that are not in BED (Figure 4).

  3. Mapping scope - It is not clear why the automation of mapping between different scopes needs to be done differently and how BED is solving this. Importantly, BioMarts and mygene also provide easy ways to map between the different scopes (gene - gene / gene - protein / gene - homolog).

- Figure 3 - we believe that it would make sense to use two different edge styles for is_replaced_by and is_associated_to since they have very different meaning. Also check the layout (in this example, it looks like the blue node is placed over the edge from the purple to the light-purple node).
Figure 5 - what do the bold borders of nodes mean in the network? Preferred identifiers? How are those selected? Additionally, when talking about the shortest relevant path, the arrows on the edges might be misleading and confusing (since there is no path from ILMN_1220595 to Q16552 taking the directionality into account).

The authors shortly mention the neo2R package to build the database. The functionality is not discussed in detail and it is unclear why the existing R package provided by Neo4j (https://neo4j.com/developer/r/) was not used. Neo4j can also be easily queried from other programming languages. Are you planning to provide APIs in other languages that would allow the integration in tools other than R?

While the conversion rate from Ensembl to Entrez Gene is very interesting, we are missing a comparison between the tools for real research examples, e.g. selection of several datasets and mapping from probe to Ensembl identifier / Entrez Gene identifier (one of the most common use cases in R workflows). This is also mentioned under the criteria for a software tool article in F1000: “The article should provide examples of suitable input data sets and include an example of the output that can be expected from the tool and how this output should be interpreted.”

Is it possible to only include edges from certain resources when performing the identifier conversion? Or do the users need to build their own database with only those selected resources?

As a final comment, we think that structure of the article is sometimes hard to follow and paragraphs are often not linked to each other. In the section “Converting identifiers” you state the following: “The aim of BED is to improve the efficiency of identifier conversion in a well defined context (organism, information resources of interest. . .) and not to replace biomaRt, mygene, gProfileR or other tools which provide many more features for many organisms and which should not be narrowed to this task for a complete comparison.” We believe that this efficiency, especially in the context of run time, is the key advantage of this tool and this should be made more clear in the article (abstract/intro/conclusion).

Is the rationale for developing the new software tool clearly explained?
Partly

Is the description of the software tool technically sound?
Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?
Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?
Partly

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?
Partly
**Competing Interests:** We would like to note that the BridgeDb framework is developed within our group.

We have read this submission. We believe that we have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however we have significant reservations, as outlined above.

---

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com