Check for updates

SOFTWARE TOOL ARTICLE

Dictionary of disease ontologies (DODO): a graph database to facilitate access and interaction with disease and phenotype ontologies [version 1; peer review: 1 approved, 1 approved with reservations]

Liesbeth François , Jonathan van Eyll, Patrice Godard

UCB Pharma, Braine l'Alleud, 1420, Belgium

V1 First published: 07 Aug 2020, 9:942
<https://doi.org/10.12688/f1000research.25144.1>Latest published: 07 Aug 2020, 9:942
<https://doi.org/10.12688/f1000research.25144.1>

Abstract

The formal, hierarchical classification of diseases and phenotypes in ontologies facilitates the connection to various biomedical databases (drugs, drug targets, genetic variant, literature information...). Connecting these resources is complicated by the use of heterogeneous disease definitions, and differences in granularity and structure. Despite ongoing efforts on integration, two challenges remain: (1) no resource provides a complete mapping across the multitude of disease ontologies and (2) there is no software available to comprehensively explore and interact with disease ontologies. In this paper, the DODO (Dictionary of Disease Ontology) database and R package are presented. DODO aims to deal with these two challenges by constructing a meta-database incorporating information of different publicly available disease ontologies. Thanks to the graph implementation, DODO allows the identification of indirect cross-references by allowing some relationships to be transitive. The R package provides several functions to build and interact with disease networks or convert identifiers between ontologies. They specifically aim to facilitate the integration of information from life science databases without the need to harmonize these upfront. The workflow for local adaptation and extension of the DODO database and a docker image with a DODO database instance are available.

Keywords

disease ontologies, diseases, phenotypes, database, identifiers

Open Peer Review

Approval Status ? ✓

	1	2
version 1	?	✓
07 Aug 2020	view	view

1. **Nicole Vasilevsky** , Oregon Health & Science University, Portland, USA
2. **Jeremy G. Frey** , University of Southampton, Southampton, UK
Samantha Kanza, University of Southampton, Southampton, UK

Any reports and responses or comments on the article can be found at the end of the article.

This article is included in the **Research on Research, Policy & Culture** gateway.

This article is included in the **RPackage** gateway.

Corresponding author: Liesbeth François (francois.liesbeth@gmail.com)

Author roles: **François L:** Conceptualization, Data Curation, Methodology, Software, Validation, Writing – Original Draft Preparation; **Eyll Jv:** Supervision, Visualization, Writing – Review & Editing; **Godard P:** Conceptualization, Methodology, Software, Supervision, Validation, Writing – Review & Editing

Competing interests: L.F., J.v.E., and P.G. are employees of UCB Pharma. J.v.E. and P.G. own stocks and/or shares from UCB Pharma. The authors declare no other competing interests.

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

Copyright: © 2020 François L *et al.* This is an open access article distributed under the terms of the **Creative Commons Attribution License**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

How to cite this article: François L, Eyll Jv and Godard P. **Dictionary of disease ontologies (DODO): a graph database to facilitate access and interaction with disease and phenotype ontologies [version 1; peer review: 1 approved, 1 approved with reservations]** F1000Research 2020, 9:942 <https://doi.org/10.12688/f1000research.25144.1>

First published: 07 Aug 2020, 9:942 <https://doi.org/10.12688/f1000research.25144.1>

Introduction

Ontologies have been developed to structure, classify, and describe diseases^{1–3}. As ontologies were independently created to support different biomedical databases dealing with genetics, treatments, and demographics, they differ in granularity and organization and define diseases in different ways^{2–8}. While this stimulated the construction of integrated biological knowledgebases, the use of independent, ontology-specific identifiers, heterogeneous decisions on disease definitions, and the inherent presence of errors complicates integrating disease ontologies^{6,8}. In addition, the navigation of these large integrated knowledgebases, with often an inherently complicated data model, is difficult for most, non-expert users^{4,6,9}.

Efforts have been made to establish new integrative disease ontologies^{8,10,11}. Using semantic similarity, the Monarch Disease Ontology (MonDO) aggregates different sources including OMIM, Orphanet, NCiT, GARD, DO, and MF^{10,11}. The Disease Ontology (DO) resource aims to standardize disease descriptions and classifications from a clinical perspective using equivalence mappings^{12–14}. Finally, the **Experimental Factor Ontology (EFO)** also establishes a unified ontology (not limited to diseases) by re-using several reference ontologies that lie within its scope. It subsequently enriches these classes with additional axioms when needed (Malone *et al.* 2010). Currently, it combines information from OMIM, Orphanet, ICD9/10 and SNOMEDCT, HPO, UBERON, and MonDO.

Despite these ongoing efforts, two bottlenecks hamper the connection of information around disease ontologies efficiently: (1) no resource provides a complete mapping across the multitude of ontologies and (2) there is no software available to comprehensively explore and interact with disease ontologies. The Dictionary of Disease Ontologies (DODO) was developed to address these two challenges. (1) While efforts such as MonDO and EFO try to integrate different disease ontologies through semantic learning and manual curation, these resources, like the different disease ontologies themselves, are currently not providing a complete mapping across all diseases ontologies^{8,9}. In addition, the existing efforts for integration are not flexible to extend easily to proprietary disease ontologies. DODO combines the information provided by the different ontologies and allows connecting ontologies to one another, even if they don't have direct cross-references. This is achieved by transitive mapping and carefully considering some indirect cross-reference relationships. (2) The second challenge addressed by DODO is the lack of an efficient and straightforward manner to access disease information through well-established bioinformatics platforms (such as R or python)^{8,15}. Such access would facilitate a more flexible connection to the different life science resources and create a more complete biomedical knowledge landscape. Currently, the programmatic access provided by many ontologies often requires expertise in creating SPARQL queries and a high level of understanding of the underlying databases or data model to be able to generate more complex queries^{4,8,9}. The DODO R package allows easier access, exploration, and definition of disease concepts of interest. It can work as an intermediate layer to facilitate access and ensure exhaustive extraction of information from life science databases without the need to harmonize upfront. Here, the DODO graph database and R package are introduced, and we exemplify their added value by going through some use cases.

Methods

R version: R version 3.6.0 (2019-04-26)

Bioconductor version: 3.10

Package: 1.0.0

Implementation

In this section, an overview of the DODO database and the R package is presented.

Data model

The data model underlying DODO aims to capture the relationship between disease and phenotypes as described across different databases (Figure 1). *Disease* and *Phenotype* are two different kinds of *Concepts* sharing the same properties and they are related through a *has_pheno* relationship. *Concepts* are identified by name, the concatenation of the short identifier (*shortID*) and *database*. Additional properties of a *Concept* are (when available) a unique canonical *label*, disease *definition*, several *Synonyms*, and node *type*. Each *Concept* is *in* one or multiple *Databases* with their URL encoded in the *idURL* property. Therefore, the *Database name* can differ from the *Concept* database as some ontologies re-use Concept names like EFO⁷. If the concepts are hierarchically organized, the *level* captures the highest level a disease has in the ontological tree (the most general term being level 0). This information is encoded as a property of a *Concept* where it captures the *level* of the term in the original ontology. In addition, as *Concepts* can be re-used, the ontology-specific *level* is also encoded as a property of the *is_in* relationship. Hierarchical information is also encoded by identifying a parent *Concept* through an *is_a*

Figure 1. The DODO data model. It is shown as an Entity/Relationship (ER) diagram: entities (concept, disease, phenotype, database) correspond to graph nodes and relationships to graph edges. *ID* and *idx* refer to a unique or indexed entity respectively. Some properties are duplicated in upper case (*_up*) to improve the performance of case-insensitive searches. The system node is a technical node used to document the DODO instance."

relationship. The property *origin* is assigned to this edge and corresponds to the ontology from which the relationship is derived (this can be useful when *Concept* names are re-used). Former and alternative identifiers of a *Concept* are documented using *is_alt* relationship.

Cross-references between *Concepts* are managed through two types of relationships depending of the confidence put on them (more detailed explanation below). The *is_xref* relationship is considered for more trusted relationships compared to *is_related* relationship. Each cross-reference (*is_xref* and *is_related*) between two *Concept* nodes is recorded as two edges, each in one direction. A *Concept* can have multiple cross-reference relationships to nodes of the same database. This ambiguity (one-to-many) is captured by the property *FA* (forward ambiguity) on a cross-reference edge and captures the number of cross-references to the same *Concept* database. The *BA* (backward ambiguity) property of the edge is defined symmetrically as the *FA* property of the edge going in the opposite direction. Both types of cross-reference edges and forward and backward ambiguities are used to define the relations used for transitivity mapping as explained below.

Feeding the database

A DODO instance is built on data from external resources that should be pre-processed to organize the information. This work was done for several public ontologies and the scripts can be found in the corresponding GitHub repositories (Table 1).

A R markdown document showing how to construct a new instance is provided alongside with a set of scripts to load and feed a Neo4j instance. These are not exported to avoid confusing the user when querying the database. The different steps to construct a new DODO Neo4j instance are briefly described below:

1. Structuring and harmonize the information derived from each ontology
2. Combine the information obtained from the different ontologies and identify any duplicate or missing data
3. Start a new Neo4j instance and load data model

Table 1. Different disease ontologies included into DODO database, and link to GitHub repository and archived source code as at time of publication. All software under GPL-3.

Disease ontology	GitHub (DOI for archived source code)
Monarch Disease Ontology (MonDO)	https://github.com/Elysheba/Monarch (10.5281/zenodo.3932755)
Experimental Factor Ontology (EFO)	https://github.com/Elysheba/EFO (10.5281/zenodo.3932759)
Orphanet	https://github.com/Elysheba/Orphanet (10.5281/zenodo.3932778)
MedGen	https://github.com/Elysheba/MedGen (10.5281/zenodo.3932770)
Medical Subject Headings (MeSH)	https://github.com/Elysheba/MeSH (http://doi.org/10.5281/zenodo.3932763)
Human Phenotype Ontology (HPO)	https://github.com/patzaw/HPO (10.5281/zenodo.3935840)
ClinVar	https://github.com/patzaw/ClinVar (10.5281/zenodo.3935838)
Disease Ontology (DO)	https://github.com/Elysheba/DO (10.5281/zenodo.3932766)
International Classification of Diseases (ICD11)	https://github.com/Elysheba/ICD11 (10.5281/zenodo.3932772)

- Information is imported into Neo4j by type. First, information around database nodes and concept nodes is imported. Next, information on the different relationships are loaded into the instance (cross-references, parent/child, alternative identifiers, phenotype mappings). For cross-reference identifiers, the type of edge is defined based on (see Extended Data¹⁶) and subsequently imported into Neo4j. After import, the forward and backward ambiguity is calculated on each edge and assigned as property values to that edge.

Database instance availability

The DODO instance build using the workflow described above is provided as a Docker image¹⁷: <https://hub.docker.com/repository/docker/elysheba/public-dodo> (tag: 02.04.2020). This instance is built on information from the following disease ontologies listed in (Table 1).

DODO contains information on 54 different disease ontologies (Figure 2). There are 418,881 disease nodes and 18,354 phenotype nodes present in the database. 92,300 disease nodes have no recorded `is_xref` or `is_related` relationships across ontologies. The number of edges per relationship type is listed in Table 2.

Operation

The database is implemented in Neo4j which uses the Cypher query language¹⁸. A DODO R package was developed to interact with and provide higher level functions to query the Neo4j graph database based on the data model described above¹⁹.

The minimal system requirements are:

- R ≥ 3.6
- Operating system: Linux, macOS, Windows
- Memory $\geq 4\text{GB}$ RAM

The graph database has been implemented with Neo4j 3.5.14¹⁷ with the `apoc.path.expand` procedure 3.5.0.11. The DODO R package uses the following packages:

- *dplyr*²⁰
- *tibble*²¹
- *neo2R*²²
- *rlist*²³
- *stringr*²⁴
- *readr*²⁵

Figure 2. Overview of the number of nodes present for each disease ontology in DODO. 30 ontologies have less than 100 entries in DODO are summarized as 'other'.

Table 2. The number of edges for each relationship type present in DODO graph database.

Relation	Number of edges
is_xref	280,691 (bidirectionally implemented)
is_related	225,021 (bidirectionally implemented)
has_pheno	363,589
is_a	221,360
is_alt	5,057

- *visNetwork*²⁶
- *shinythemes*²⁷
- *DT*²⁸
- *igraph*²⁹
- *shiny*³⁰

Querying the database

The DODO R package combines several functions to construct, interact, and explore the relationships between disease and phenotype identifiers. These can be divided in five different scopes (see Extended Data¹⁶):

- **Disease network functions:** these functions allow the user to build a disease network based on their relationships in the graph database: cross-reference, hierarchical information, and phenotypes. These functions include ways to extend, filter, split or combine disease networks. The information on diseases and phenotypes and their relationships is structured as a S3 disease network (disNet) object. It captures all information (disease node information, hierarchical information, phenotype information, alternative identifiers, and cross-reference information) around a disease.

- **Visualization and exploration:** these functions allow the exploration and visualization of relationships between disease and phenotype identifiers by querying them or providing a disease network.
- **Conversion:** this category of functions converts a list of disease and phenotype identifiers across ontologies or concepts based on the data model and provided parameters. Conversion can also include indirect relationships across ontologies using transitive mappings.
- **Connection and low level interactions:** these functions are technical helpers for establishing and managing connection with a DODO graph database. These functions also return information on the content of the current database and allow to directly send cypher queries to the database.
- **Data information:** these functions give access to content information such as the list of original databases, concept description and reference URL and allow ontology dump.

Transitivity mapping

Ambiguity of cross-reference relationships. Disease identifiers within the different ontologies are annotated with cross-reference relationships to connect to independent biomedical databases. However, no ontology provides a complete mapping across all existing ontologies^{8,9}. Efforts to create an integrated ontology such as **MonDO** and **EFO** are continuously expanding to enrich their mappings but are currently not exhaustive and will lack mappings to proprietary ontologies. Here we propose the use of transitive mappings to extend the cross-reference information and connect ontologies (and biomedical resources) that lack direct relationship(s). This is exemplified in **Figure 3**, where transitivity mappings is needed to connect the initial MONDO identifier to a MeSH identifier using the indirect cross-reference relationship to the EFO and ORPHA node.

Most mappings are unambiguous (one concept in an ontology is related to only one concept in another ontology); however, some concepts map to many similar concepts within the same ontology. This is conceptually visualized in **Figure 3** where the MONDO identifier maps to 2 ORPHA identifiers.

A real example is provided in **Figure 4** with focus on the “Coffin-Lowry syndrome” (ORPHA: 192), a rare syndrome affecting brain and skeleton development. It relates to many disease identifiers, that often share similar disease definitions in an unambiguous or one-on-one manner. These relationships are valuable for transitive mapping as they extend the initial node to very similar nodes in other ontologies. However, the cross-reference relationship to ICD10 deals with a very broad term of “Congenital malformation syndromes predominantly affecting facial appearance” (ICD:Q87.0). This identifier is highly ambiguous as it has 285 additional direct cross-reference relationships to other disease identifiers. The use of this type of indirect relationships needs to be carefully considered. This ambiguity is a consequence of the inter-ontology differences in concept definitions and precision where some cross-reference edges connect identifiers that are not equivalent. Ontologies such as MonDO or EFO use a greater granularity in disease definitions than others like ICD10 or ICD9. If cross-reference edges are all considered equal without taking this distinction into account, it would be detrimental to the relevance of the conversion as it will return numerous more distantly related concepts when traversing these edges. Therefore, such relationships need to be avoided for transitive mappings.

To prevent these inaccurate conversions, we propose the use of backward ambiguity filtering. First, the forward ambiguity of a cross-reference edge needs to be quantified and it is the number of relationships where a node maps to

Figure 3. Example of transitivity mapping to infer an indirect relation with information on the ambiguity on the cross-reference edges.

Figure 4. Subset of a disease network around Coffin-Lowry syndrome (ORPHA:192). The network shows equivalent cross-reference relations and ambiguous cross-reference relation to ICD10:Q87.0. This ICD10 disease identifier connects to 283 additional disease identifiers. The colors refer to different databases. *Is_xref* and *is_related* edges are indicated in blue and orange respectively. The arrows on the edge refer to the backward ambiguity: the identifier at the arrow destination has only one cross-reference in the database of the identifier at the arrow start. A double arrow indicates an unambiguous mapping between the 2 databases for the 2 identifiers in both directions. An edge should be considered transitive only in the direction of the arrow otherwise it can only be the final edge of a conversion path.

several similar concepts within the same ontology. The value of the forward ambiguity is shown for the example in Figure 4 on each cross-reference edge between the nodes. A forward ambiguity greater than one indicates that the original concept is likely more general than the concepts it maps to. By following the relationship in the other direction, this ambiguity value is considered as the backward ambiguity. Similarly, a backward ambiguity greater than one, indicates that the original concept is probably more precise than the concepts it is mapped to (Figure 4). Applied to our example, the filtering on ambiguity will prevent traversing through the ambiguous ICD10 when using the transitivity mechanisms and return only cross-reference identifiers close to the original node.

Defining subtypes of cross-reference edges. The filter on backward ambiguity improves greatly the accuracy of conversions. However, some inaccurate mappings may still be present due to higher general ambiguity between specific ontologies. The maximum total ambiguity of all relationships between two ontologies (the maximum value of the sum of forward and backward ambiguities of all cross-reference edges) quantifies the symmetry of their cross-reference relationships. The heatmap in Figure 5 shows this value in a $\log_{10}$ scale. While many ontologies are using concepts of similar level as identified by low maximum total ambiguity, a few can be identified that are more ambiguous in their mappings. Therefore, the general trust assigned to cross-reference relationships between ontologies is captured by defining two types of cross-reference edges: (1) the *is_xref* edge is used for equal cross-reference relationships where the concepts relate more directly to each other (similar concept definitions); (2) the *is_related* edge is used for all other cross-reference edges.

To assign relations to either type, a threshold is applied on the maximum total ambiguity between two ontologies. Different thresholds between 2 and 20 have been explored taking 14 different ontologies into account. For each step the cross-reference relationships between ontologies with the maximum total ambiguity falling below the threshold are defined as *is_xref* and otherwise as *is_related*. Next, all the identifiers from an ontology are converted to identifiers from any other ontology by applying transitive mappings only on *is_xref* relationships with a backward ambiguity of 1 (step = NULL, intransitive_ambiguity = 1). Since the number of *is_xref* relationships increases with the threshold applied, the number of conversions achieved from each identifier increases as well (see Figure 6 for an example focused on the MONDO ontology).

When choosing a conservative and low threshold, the conversions may be more precise but will hamper the ability of the transitive mappings to identify the most relevant mappings. It may therefore not return all

Figure 5. The heatmap shows the maximum value of total ambiguity between ontologies using a $\log_{10}$ transformation.

Figure 6. Number of conversions from each MONDO identifiers. This number depends on the cut-offs in the (total) ambiguity to define *is_xref* and *is_related* cross-reference edges on an ontology scale shown in x axis.

cross-reference identifiers whereas a too high threshold impacts strongly the number of converted nodes at the cost of some inaccurate conversions. Especially the effect of on a limited set of identifiers is strong with the return of many, more distantly related and less precise cross-reference identifiers as can be seen by looking at the tail of the boxplot shown on [Figure 6](#).

After having compared the results of the different ontologies, the threshold on the maximum total ambiguity value has been arbitrarily set to 4: cross-references between ontology with a maximum total ambiguity below 4 were considered as *is_xref* and others as *is_related* edges (see Extended Data¹⁶). However, two exceptions are ICD9/ICD10 and OMIM/Orphanet. Both ICD9 and ICD10 use higher level disease definitions, and therefore, these will never be connected through an *is_xref* edge except between themselves. OMIM and Orphanet on the

other hand, use very narrow subtypes of diseases or mention specific variants. This results in a higher ambiguity as a general disease may have many relations to subtypes but as these ontologies use a higher granularity their relationships are still encoded as an *is_xref* edge.

Use cases

Conversion of concept identifiers

One of the basic functionalities of DODO is the ability to convert disease and phenotype identifiers between ontologies. The conversion of identifiers is generally performed using a two-step process based on the type of cross-reference edge to traverse and ambiguity values to filter on. The first step uses transitivity on *is_xref* edges only to identify the high confidence identifiers mapped to the initial identifier of interest. This makes the core of cross-referenced identifiers. It is strongly recommended to limit the backward ambiguity in this transitive mapping to one (*transitive_ambiguity* = 1 (default)) to avoid unwilling conversion to less accurate concepts. Once the core of identifiers is established, the second step expands it by a single step using both types of cross-reference edges (*is_xref* and *is_related*) with specification on the backward ambiguity filtering (“*intransitive_ambiguity*” parameter).

Six separate use cases can be identified for converting disease or phenotype identifiers to other ontologies or concepts:

- *Direct conversion*: only return the direct cross-references without any transitive mapping
- *Strict indirect conversion*: uses transitive mapping and applies a filter on the backward ambiguity of the last step to only return equivalent concepts. Can be used to convert between ontologies that use similar granularity to define concepts.
- *Extended indirect conversion (default)*: uses transitive mapping without any filter on the backward ambiguity of the last step to return both equivalent and more broader terms. This way of conversion returns ambiguous relations, but these relations are not used for transitive mapping steps. In general, when the aim is to reach a broader concept related to the original identifiers but not move through it, it is recommended to put no filter on the backward ambiguity filtering of the final step.
- *Loosened indirect conversion*: a specific conversion procedure is created for ontologies that are less connected through *is_xref* edges such as ontologies like WHO’s ICD10 or ICD9. These ontologies have highly ambiguous mappings and use very broad disease definitions. The absence of *is_xref* cross-reference relationships restrict the utility of the transitivity mechanism when applying the standard conversion. Therefore, we recommend an additional step to the standard conversion implemented in the *get_related* function.
- *Conversion between concepts*: convert between concepts types, i.e. from disease identifier to phenotype identifiers or vice versa.
- *Return deprecated identifiers*: return previous version (deprecated) identifiers

The use cases will be illustrated using the Mondo identifier for epilepsy (MONDO:0005027). The conversion between concept types (phenotype to disease or vice versa) is exemplified here for one identifier; however, the conversion procedure can take multiple identifiers at once as input. Finally, a comparison of the different conversion possibilities is performed using the entire MonDO ontology.

Use case 1: Direct conversion

A first basic use case is conversion of the identifier of interest to direct cross-references (without any transitive mapping - parameter “step = 1”). As an example, MonDO identifier for epilepsy (MONDO:0005027) is mapped to its corresponding EFO identifier using *convert_concept* function.

In this first use case, only the direct relations are used to return the corresponding EFO identifiers (highlighted in red on [Figure 7](#)). Using transitive mappings is also not required for this example, as there would be no additional EFO identifiers returned by moving through indirect relationships.

Figure 7. The network of identifiers around the Mondo identifier for 'Epilepsy' ('MONDO:0005027'). The seed node, MONDO:0005027, is depicted as a yellow triangle. Subsequently, the results of the different use cases are highlighted. The results of use case 1 (direct conversion) is indicated in red. The results of use case 2 (strict indirect conversion) and 3 (extended indirect conversion) to obtain corresponding Orphanet identifiers using transitivity are indicated in blue and green, respectively. *Is_xref* and *is_related* edges are indicated in blue and orange respectively. The arrows on the edge indicate the direction mapping can occur taking into account a backward ambiguity of one. Absence of arrow on an edge indicate these relations can only be exploited when no filtering is applied (generally final step of conversion).

```
## Use case 1
conv <- convert_concept(from = "MONDO:0005027",
                        to = "EFO",
                        from.concept = "Disease",
                        to.concept = "Disease",
                        step = 1,
                        intransitive_ambiguity = 1)

conv
```

```
## # A tibble: 1 × 3
##   from      to      deprecated
##   <chr>     <chr>    <lgl>
## 1 MONDO:0005027 EFO:0000474 NA
```

Use case 2: Strict indirect conversion

A second use case deals with the conversion using equivalent indirect relations. The same seed MonDO identifier for epilepsy will be mapped to return the corresponding Orphanet ontology identifier. By specifying “step = NULL”, transitive mappings will be used to convert the identifier(s). To return only equivalent concepts, thereby avoiding the mapping to less precise disease concepts, backward ambiguity filtering is applied on the final step of conversion (“intransitive_ambiguity = 1”). This can be a good practice when converting between ontologies that define concept with similar granularity. As there is no direct mapping provided by the resources between MonDO and Orphanet, transitive mapping provided by DODO needs to be applied. This conversion of the seed identifier MONDO:0005027 returns one corresponding Orphanet identifier (ORPHA:166463) as shown in [Figure 7](#) (blue node).

```
## Use case 2
conv <- convert_concept(from = "MONDO:0005027",
  to = "ORPHA",
  from.concept = "Disease",
  to.concept = "Disease",
  step = NULL,
  intransitive_ambiguity = 1)

conv
```

```
## # A tibble: 1 × 3
##   from      to      deprecated
##   <chr>     <chr>    <lgl>
## 1 MONDO:0005027 ORPHA:166463 NA
```

Use case 3: Extended indirect conversion (default)

In some instances, you may want to define a list of disease identifiers more largely related to a disease of interest. The third use case addresses this by extending the transitivity mapping with no filtering on the final step of conversion (“intransitive_ambiguity = NULL” and “step = NULL”). The conversion of the seed identifier MONDO:0005027 to Orphanet with this setup, returns an additional Orphanet identifier (ORPHA:101993) indicated in green on [Figure 7](#). The initial transitive mapping steps apply filtering with ambiguity equal to one via the *is_xref* edges (blue). The final mapping step is intransitive and will therefore return all nodes related through either an *is_related* or *is_xref* edge with no filtering on backward ambiguity. It does return ambiguous relations at the last step, but these relations are not used for transitive mapping steps. This conversion can be used to get all identifiers around a disease concept both equivalent and more broader terms or when converting from a narrower concept to a broader concept. In general, when the aim is to reach a broader concept related to the original identifiers but not move through it, it is recommended to put no filter on the “intransitive_ambiguity”.

```
## Use case 3
conv <- convert_concept(from = "MONDO:0005027",
  to = "ORPHA",
  from.concept = "Disease",
  to.concept = "Disease",
  step = NULL,
  intransitive_ambiguity = NULL)

conv
```

```
## # A tibble: 2 × 3
##   from      to      deprecated
##   <chr>     <chr>    <lgl>
## 1 MONDO:0005027 ORPHA:101998 NA
## 2 MONDO:0005027 ORPHA:166463 NA
```

Use case 4: Loosened indirect conversion

The fourth use case deals with the specific conversion procedure that is recommended for the ontologies which are less connected through *is_xref* edges to the “core” ontologies in DODO (e.g. MonDO, EFO, MedGen, etc.) such as ontologies like WHO’s ICD10 or ICD9. These ontologies have highly ambiguous mappings and use very broad disease definitions. When starting the mapping from such ambiguous identifiers, the *convert_concept* function will not return cross-reference relationships to most other databases. It is restricted by the rules defined by transitivity mechanism. Therefore, we recommend an additional step to the standard conversion implemented in the *get_related* function. It performs an additional expansion step through *is_related* and *is_xref* edges before the standard conversion procedure. The ambiguity on this additional step is the same for the final step in the standard conversion procedure (use case 2 – strict indirect conversion - modified by the *intransitive_ambiguity* parameter equal to NULL). To illustrate the difference, we will show the conversion of the general ICD10 identifier for Epilepsy (ICD10:G40.9) to corresponding identifiers in DO. When using the standard *convert_concept* function, it is not possible to convert the ICD10 identifier to any DO identifier. As ICD10 is only related to other nodes via *is_related* edges, transitive relationships cannot be used to identify cross-references using the standard conversion. Using the *get_related* function, recommended for ontologies such as ICD10, does convert ICD10:G40.9 to corresponding DO identifiers through transitive relations and returns DOID:1826 (Epilepsy).

```
## Use case 4
## convert_concept()
conversion <- convert_concept(from = "ICD10:G40.9",
                             to = "DOID",
                             from.concept = "Disease",
                             to.concept = "Disease")

conversion

## # A tibble: 0 × 3
## # ... with 3 variables: from <chr>, to <chr>, deprecated <lgl>
```

```
## get_related()
related <- get_related(from = "ICD10:G40.9",
                      to = "DOID",
                      from.concept = "Disease",
                      to.concept = "Disease")

related

## # A tibble: 1 × 3
##   from      to      deprecated
##   <chr>    <chr>    <lgl>
## 1 ICD10:G40.9 DOID:1826 FALSE
```

The underlying reason for this behavior and difference between the two functions is depicted in [Figure 8](#). As there are no relationships from the ICD10 seed node that meet the criteria defined for transitive mapping (*intransitivity_ambiguity* = 1 and cross-reference relationships of *is_xref* type), the seed node cannot be mapped to the DO node of interest using the *convert_concept* function. The *get_related* function relaxes these transitive mapping criteria in the first step and allow to reach the directly related nodes connected by either type of relationships and with no constraint on backward ambiguity (green round circles in [Figure 8](#)). After this initial step, transitivity rules are applied to map the direct cross-reference identifiers to the identifier in DO (orange node in [Figure 8](#)).

Currently, the default conversion procedure refers to the third use case (extended indirect conversion) when transitivity mappings are used to extend through cross-reference relationships and no filtering is applied to the final step of extension through the network (default parameters: *step* = NULL, *intransitive_ambiguity* = 1). This conversion can be used to get all identifiers around a disease concept and it is the approach to use when converting from a narrower concept to a broader concept. Additional filtering can then subsequently be put in place to allow adaptation for specific use cases. However, for the first step using transitivity mapping on *is_xref* edges it is strongly recommended to use the default filtering on ambiguity by limiting (backward) ambiguity to one.

Figure 8. The entire network of disease identifiers around the ICD10 identifier for 'Epilepsy' (ICD10:G40.9 – green triangle). *Is_xref* and *is_related* edges are indicated in blue and orange respectively. The arrows on the edge indicate the direction mapping can occur taking into account the BA = 1 rule. Absence of arrow on an edge indicate these relations can only be exploited when no filtering is applied. Using the *get_related* functionality adds an additional step of conversion before applying the standard conversion procedure. The nodes indicated in green are those that would be returned using the standard *convert_concept* function. The grey nodes are those reachable by *get_related* function that relaxes the initial step of transitivity mappings through *is_related* edges. The retrieval of corresponding DO identifiers for ICD10:G40.9 returns DOID:1826 (indicated in orange).

Use case 5: Conversion between concepts

Conversion can also be used to convert between concepts types, i.e. from disease identifier to phenotype identifiers or vice versa. This conversion is achieved by the same *convert_concept* function that will leverage *has_pheno* relationships. Practically, it is handled in two phases. The first phase uses the transitivity mechanism as detailed above to connect disease identifiers to phenotype identifiers even when no direct *has_pheno* relation

```
## From disease to phenotype
toPhenotype <- convert_concept(from = "MONDO:0012391",
                               to = "HP",
                               from.concept = "Disease",
                               to.concept = "Phenotype")

toPhenotype <- toPhenotype %>%
  mutate(diseaseLabel = describe_concept(from)$label,
         phenotypeLabel = describe_concept(to)$label)
toPhenotype
```

is available. This initial step takes the same options as listed above to convert identifiers within the same concept (this step can be avoided by using parameter “step = NA”). The second phase converts identifiers between concepts by returning phenotype or disease nodes related to the original identifiers (including the converted identifiers obtained in the first phase) with the possibility to return direct and/or indirect relations with the parameter “step”.

```
## # A tibble: 18 x 5
##   from to deprecated diseaseLabel phenotypeLabel
##   <chr> <chr> <lgl> <chr> <chr>
## 1 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Cerebellar atrophy
## 2 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Cerebral atrophy
## 3 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Increased neuronal autoflu~
## 4 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Clumsiness
## 5 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Progressive visual loss
## 6 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Mental deterioration
## 7 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Irritability
## 8 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Generalized tonic-clonic s~
## 9 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ EEG abnormality
## 10 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Focal impaired awareness s~
## 11 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Intellectual disability
## 12 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Developmental regression
## 13 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Curvilinear intracellular~
## 14 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Restlessness
## 15 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Psychosis
## 16 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Behavioral abnormality
## 17 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Slow progression
## 18 MONDO:~ HP:000~ FALSE neuronal ceroid lipof~ Autosomal recessive inheri~
```

```
## From phenotype to disease
toDisease <- convert_concept(from = "HP:0002384",
                             from.concept = "Phenotype",
                             to.concept = "Disease")

toDisease <- toDisease %>%
  mutate(phenotypeLabel = describe_concept(from)$label,
         diseaseLabel = describe_concept(to)$label)
toDisease
```

```
## # A tibble: 72 x 5
##   from to deprecated phenotypeLabel diseaseLabel
##   <chr> <chr> <lgl> <chr> <chr>
## 1 HP:000~ HP:0002~ FALSE Focal impaired awar~ neuronal ceroid lipofuscino~
## 2 HP:000~ ClinVar~ FALSE Focal impaired awar~ spinocerebellar ataxia type~
## 3 HP:000~ UMLS:C0~ FALSE Focal impaired awar~ hyperekplexia-epilepsy synd~
## 4 HP:000~ MONDO:0~ FALSE Focal impaired awar~ bilateral parasagittal pari~
## 5 HP:000~ MONDO:0~ FALSE Focal impaired awar~ benign familial neonatal-in~
## 6 HP:000~ MONDO:0~ FALSE Focal impaired awar~ unilateral polymicrogyria
## 7 HP:000~ MONDO:0~ FALSE Focal impaired awar~ Spinocerebellar ataxia type~
## 8 HP:000~ MONDO:0~ FALSE Focal impaired awar~ Progressive epilepsy - inte~
## 9 HP:000~ MONDO:0~ FALSE Focal impaired awar~ Lafora disease
## 10 HP:000~ MONDO:0~ FALSE Focal impaired awar~ isolated focal cortical dys~
## # ... with 62 more rows
```

Use case 6: return deprecated identifiers

Finally, conversion can also be used to return previous version (deprecated) identifiers when these are available.


```

deprecated <- convert_concept(from = "HP:0009638",
                             deprecated = TRUE,
                             from.concept = "Phenotype",
                             to.concept = "Phenotype")

deprecated

```

```

## # A tibble: 2 x 3
##   from      to      deprecated
##   <chr>    <chr>    <lgl>
## 1 HP:0009638 HP:0004079 TRUE
## 2 HP:0009638 HP:0006073 TRUE

```

Efficiency of conversion strategies

The different use cases were applied to only one identifier to show in more detail the conversion procedure. However, the conversion procedure is designed to take multiple identifiers as input. Here, the first three different approaches outlined in the previous section are applied to convert all 21,653 identifiers in the MonDO ontology to their corresponding identifiers in EFO, MeSH, and DO. The mapping outcomes are summarized in [Table 3](#).

```

mondo <- get_ontology("MONDO")

## option 1

```

Table 3. Comparison of different conversion strategies using the MonDO ontology.

Conversion options	#MonDO identifiers ¹	#Target ontology identifiers ²	Distribution of equivalent mapping ³			#MonDO with ambiguous conversion ⁴
			Median	Mean	Max	
Database: EFO						
Direct conversion ⁵	2989	2999	1	1.006	2	19
Strict indirect conversion ⁶	3621	3150	1	1.145	9	378
Extended indirect conversion ⁷	3944	3152	1	1.158	9	444
Database: MeSH						
Direct conversion ⁵	7798	7835	1	1.005	2	38
Strict indirect conversion ⁶	9574	9207	1	1.361	90	1652
Extended indirect conversion ⁷	10318	9315	1	1.377	91	1895
Database: DO						
Direct conversion ⁵	21653	30644	1	1.416	4	8933
Strict indirect conversion ⁶	21653	31735	2	2.070	294	11314
Extended indirect conversion ⁷	21653	31736	2	2.133	294	12269

¹Number of unique MonDO identifiers with a conversion

²Number of unique converted identifiers in the targeted ontology

³Distribution of the number of conversions returned per MonDO identifier.

⁴Number of MonDO identifiers with ambiguous conversions

⁵Conversion as performed by use case 1: direct conversion (see above), the parameter is step = 1

⁶Conversion as performed by use case 2: strict indirect conversion (see above), the parameters are step = NULL and intransitive_ambiguity = 1

⁷Conversion as performed by use case 3: extended indirect conversion (see above), the parameters are step = NULL and intransitive_ambiguity = NULL)

```

conv1 <- convert_concept(from = mondo$nodes$id,
                        to = "EFO",
                        from.concept = "Disease",
                        to.concept = "Disease",
                        step = 1)
summary(conv1 %>% count(from) %>% pull(n))

conv2 <- convert_concept(from = mondo$nodes$id,
                        to = "EFO",
                        from.concept = "Disease",
                        to.concept = "Disease",
                        step = NULL,
                        intransitive_ambiguity = 1)
summary(conv2 %>% count(from) %>% pull(n))

conv3 <- convert_concept(from = mondo$nodes$id,
                        to = "EFO",
                        from.concept = "Disease",
                        to.concept = "Disease",
                        step = NULL,
                        intransitive_ambiguity = NULL)
summary(conv3 %>% count(from) %>% pull(n))

```

Except for Disease Ontology (DO) (which is included by MonDO while constructing their ontology based on semantic similarity), the majority of MonDO identifiers cannot be converted to EFO or MeSH identifiers. However, compared to direct mapping of encoded relationships (use case 1 – direct conversion), the use of transitive mapping allows the conversion of 20% additional MonDO identifiers (use case 2 – strict indirect conversion). Enabling the extension to broader disease concepts (use case 3 – extended indirect conversion), still increases, by design and as expected, the number of mappings between two ontologies.

In parallel, the average ambiguity in mappings also increases with the different use cases (Table 3 - column 3). While it varies strongly across the different ontologies, the ambiguity is mostly minor for most identifiers (based on median and mean). Still, a limited set of mappings is strongly affected by ambiguity with as many as 294 corresponding DO identifiers and 91 corresponding MeSH identifiers for “autosomal dominant non-syndromic deafness” (MONDO:0019587). This identifier could not be mapped to EFO. Looking into this identifier in more detail shows that it only has six direct cross-references among which one DO identifier (DOID:0050564) (Figure 9). However, these direct cross-reference identifiers themselves report a multitude of cross-reference identifiers encoding various subtypes of the condition. As mentioned before, the mappings are derived directly from the original resources. The observed ambiguity after transitive mapping highlights disease areas that are heterogeneously defined across ontologies. This ambiguity is naturally more prevalent for those indications with a lot of reported subtypes reported in different ontology and/or the ontology using a higher level of granularity used to define diseases.

Build and explore a network of diseases

Building a disease network. While conversion facilitates connecting biomedical resources directly, another possibility provided by DODO is the exploration of diseases and their relationships as a disease network. Contrary to conversion, a network retains all relationships as they are encoded in the DODO graph database. In this use case, we will show how to construct such a disease network and which functionalities are available to interact with a network.

```

disNet <- build_disNet(term = "amyotrophic lateral sclerosis",
                      fields = c("label", "synonym"))
disNet

```

Figure 9. Direct cross-reference of MonDO identifier for 'autosomal dominant non-syndromic deafness' (MONDO:0019587).

```
## # A tibble: 251 x 7
##   id      label      definition      shortID  level type  database
##   <chr>   <chr>      <chr>      <chr>    <int> <chr>  <chr>
## 1 ORPHA:~ Juvenile amyotro~ Juvenile amyotrophi~ 300605      6 Concep~ ORPHA
## 2 UMLS:C~ Amyotrophic late~ <NA>      C18629~    NA Concep~ UMLS
## 3 ClinVa~ Amyotrophic late~ Amyotrophic lateral~ 16012      NA Concep~ ClinVar
## 4 OMIM:6~ AMYOTROPHIC LATE~ AMYOTROPHIC LATERAL~ 606640      NA Concep~ OMIM
## 5 MedGen~ Amyotrophic late~ Amyotrophic lateral~ C26754~    NA Concep~ MedGen
## 6 UMLS:C~ <NA>      <NA>      C18629~    NA Concep~ UMLS
## 7 UMLS:C~ Amyotrophic late~ <NA>      C18654~    NA Concep~ UMLS
## 8 MedGen~ Amyotrophic late~ Amyotrophic lateral~ C32805~    NA Concep~ MedGen
## 9 OMIM:6~ AMYOTROPHIC LATE~ AMYOTROPHIC LATERAL~ 613954      NA Concep~ OMIM
## 10 MedGen~ Amyotrophic late~ Amyotrophic lateral~ C31514~    NA Concep~ MedGen
## # ... with 241 more rows
##
## The disNet contains:
## - 250 disease nodes from 11 ontologies and 1 phenotype nodes from 1 ontologies
## - 1126 synonyms of the disease nodes
## - 62 parent/child edges
## - 470 crossreference edges
## - 0 alternative edges
## - 66 phenotype edges
## - The disNet was build based on 251 seeds
```

Extension through different relationships. After the construction of a disease network, it is likely that it doesn't contain the complete information on that particular disease of interest. The *extend_disNet* function enriches the *disNet* and extends it to cross-reference identifiers, child/parent terms, annotated phenotypes/disease, and/or alternative identifiers when available. In concordance with the conversion procedure, extension follows the same two-step approach using transitive mapping on *is_xref* edges followed by a final one-step extension on any cross-reference relationship taking filtering on backward ambiguity into account. To perform this extension the same parameters can be supplied with similar aims as for the conversion procedure (see above).

```
disNet <- build_disNet(term = "amyotrophic lateral sclerosis",
  fields = c("label", "synonym"))
extendedDisNet <- extend_disNet(disNet,
  relations = c("xref", "child"),
  intransitive.ambiguity = 1)
extendedDisNet
```

```
## # A tibble: 494 x 7
##   id      label      definition      shortID level type      database
##   <chr>   <chr>      <chr>          <chr>   <int> <chr>   <chr>
## 1 ORPHA:~ Juvenile amyotro~ Juvenile amyotrophic~ 300605      6 Conce~ ORPHA
## 2 UMLS:C~ Amyotrophic late~ <NA>      C18629~    NA Conce~ UMLS
## 3 GTR:GT~ <NA>         <NA>      GTRT00~    NA Conce~ GTR
## 4 MedGen~ Inclusion body m~ Inclusion body myopa~ C38094~    NA Conce~ MedGen
## 5 GARD:0~ <NA>         <NA>      0010499    NA Conce~ GARD
## 6 ClinVa~ Amyotrophic late~ Amyotrophic lateral ~ 16012      NA Conce~ ClinVar
## 7 MeSH:C~ <NA>         <NA>      C566550    NA Conce~ MeSH
## 8 OMIM:6~ AMYOTROPHIC LATE~ AMYOTROPHIC LATERAL ~ 606640    NA Conce~ OMIM
## 9 OMIM:1~ <NA>         <NA>      164015     NA Conce~ OMIM
## 10 OMIM:6~ <NA>         AMYOTROPHIC LATERAL ~ 602572~    NA Conce~ OMIM
## # ... with 484 more rows
##
## The disNet contains:
## - 493 disease nodes from 24 ontologies and 1 phenotype nodes from 1 ontologies
## - 1431 synonyms of the disease nodes
## - 81 parent/child edges
## - 888 crossreference edges
## - 0 alternative edges
## - 0 phenotype edges
## - The disNet was build based on 251 seeds
```

The disease network gathers 488 disease concepts across 25 ontologies; only 251 were identified directly matching the search term (Figure 10). The additional terms were obtained through extension of both cross-references and parent/child relationships. Of specific note is the extension to (or from) phenotype information.

Figure 10. The disNet build on the term 'amyotrophic lateral sclerosis' querying both labels and synonyms provided in DODO (green nodes). This disNet is subsequently extended to return all cross-reference identifiers and child terms using the *extend_disNet* function (orange nodes) (parameters *relations* = c('xref', 'child') and *intransitive_ambiguity* = 1 to return only equivalent identifiers)

Within one extension all different parameters (xref, child, parent, alt, and disease/phenotype) can be supplied with the exception that it is not possible to extend to both disease and phenotype simultaneously. In contrast with the conversion procedure, it does not use the transitivity mechanisms but rather takes all the diseases within the network and returns any associated phenotypes that can be obtained through the `has_pheno` relationship.

```
disNet <- build_disNet(id = c("HP:0003394", "HP:0002180", "HP:0002878"))
disNet <- extend_disNet(disNet = disNet, relations = "disease")
disNet
```

```
## # A tibble: 397 x 7
##   id      label      definition      shortID level type      database
##   <chr>   <chr>       <chr>         <chr>   <int> <chr>   <chr>
## 1 MONDO:~ glycogen storag~ "Phosphoglycerate ki~ 0010392    11 Concep~ MONDO
## 2 OMIM:6~ NEURODEGENERATI~ "NEURODEGENERATION W~ 610217    NA Concep~ OMIM
## 3 MONDO:~ adult-onset dis~ "Adult-onset distal ~ 0018006   10 Concep~ MONDO
## 4 MONDO:~ Charcot-Marie-T~ "Autosomal dominant ~ 0011675    9 Concep~ MONDO
## 5 OMIM:6~ MOTOR NEURON DI~ "MOTOR NEURON DISEAS~ 600333    NA Concep~ OMIM
## 6 MONDO:~ pure mitochondr~ "Pure mitochondrial ~ 0016807   11 Concep~ MONDO
## 7 ORPHA:~ Congenital musc~ "Congenital muscular~ 258        8 Concep~ ORPHA
## 8 OMIM:3~ MITOCHONDRIAL C~ "MITOCHONDRIAL COMPL~ 301021    NA Concep~ OMIM
## 9 ORPHA:~ Nocardiosis      "Nocardiosis is a lo~ 31204      4 Concep~ ORPHA
## 10 ORPHA:~ Mitochondrial e~ " mutation is charac~ 1194      9 Concep~ ORPHA
## # ... with 387 more rows
##
## The disNet contains:
## - 394 disease nodes from 6 ontologies and 3 phenotype nodes from 1 ontologies
## - 2206 synonyms of the disease nodes
## - 0 parent/child edges
## - 0 crossreference edges
## - 0 alternative edges
## - 398 phenotype edges
## - The disNet was build based on 3 seeds
```

Explore a network of diseases. DODO is built as a meta-database incorporating several disease ontologies and their listed relationships. As disease concepts and definitions are not a natural process but rather an artificial, human-biased effort, concepts might not always be clearly defined or related to each other in a straightforward manner. The different ontologies employ heterogeneous definitions, cross-reference axes are not always exact, and errors present in the original ontologies will impact DODO as well. The `explore_disNet` on a single `disNet` object returns a data table presenting information on the different identifiers present in the network. The `plot` function displays as a network how diseases are related to each other across the different ontologies (Figure 11).

It may be required to review the returned network of diseases after building and/or extending it to assess whether all nodes are relevant or of interest. This process can be simplified by considering clusters of cross-references (nodes dealing with similar concepts) using the `cluster_disNet` functionality.

```
disNet <- build_disNet(term = "amyotrophic lateral sclerosis",
                      fields = c("label", "synonym"))
clDisNet <- cluster_disNet(disNet = disNet,
                          clusterOn = "xref")
clDisNet
```

```
## The setDisNet contains 29 disNet clusters
explore_disNet(clDisNet)
```

Instead of reviewing each node, the different cross-reference clusters can be reviewed to identify those of interest while using the relationships between nodes to handle equivalent nodes simultaneously without the need to review them separately. A summary of the clusters can be visualized using the `explore_disNet` function and the

Figure 11. Plot of a small disease network constructed around ‘Amyotrophic lateral sclerosis’ (MONDO:0004976). The colors refer to different databases. *Is_xref* and *is_related* edges are indicated in blue and orange respectively. The arrows on the edge refer to the backward ambiguity and show how an edge can be traversed. When no arrow is present, it can only be reached through the final step of conversion when no filtering is present.

output is shown in Table 4. For a list of disease networks created after clustering the *explore_disNet* functionality summarizes information on the different clusters, provides information on the size of each cluster and presents a tag identifier information. This tag identifier is identified as the node with the highest level in the ontology and a label available. If multiple identifiers have the same level, the tag one is picked on alphabetical order of the label. Summarizing disease networks using cluster of cross-reference edges also allows the revision of identifiers that have no label information attached.

Connecting to external resources. The aim of DODO is to facilitate the connection with external resources. As described above, DODO allows the creation of a structured disease network around diseases of interest and their relationships and information. The use of this disease network facilitates the connection and exploration of different biomedical knowledge resources as it also allows tracing their connections more easily.

It is exemplified below by connecting two different external resources: ClinVar (release 2020-03) and ChEMBL (release 25) using “amyotrophic lateral sclerosis” (ALS) as an example^{31,32}. We start by building a disease network around the term “amyotrophic lateral sclerosis” and extending through both cross-reference and parent/child relationships as described in the previous section. For comparison, ChEMBL and ClinVar are queried directly using the same search term. Each of these resources uses a different ontology as reference. ChEMBL uses both EFO and MeSH to annotate compounds to indication. ClinVar uses a variety of ontologies to connect variants and diseases, such as SNOMEDCT, MedGen, Orphanet or OMIM.

Through the use a disease network, 96 unique compounds were identified in ChEMBL for ALS connected to four different disease identifiers (Table 5). The same set of compounds is identified by querying the resource directly, demonstrating the performance of DODO to properly map those different ontologies. One associated

Table 4. Annotation of the different cross-reference clusters of nodes identified for a disNet around 'amyotrophic lateral sclerosis'.

cluster	clusterSize	id	label
1	150	ICD10CM:G12.21	Amyotrophic lateral sclerosis
2	22	MONDO:0017161	frontotemporal dementia with motor neuron disease
3	2	MedGen:C1862940	Amyotrophic Lateral Sclerosis, Autosomal Recessive
4	9	ORPHA:357043	Amyotrophic lateral sclerosis type 4
5	3	MedGen:C3542025	Amyotrophic lateral sclerosis 1, autosomal recessive
6	8	MONDO:0005145	sporadic amyotrophic lateral sclerosis
7	6	MONDO:0014640	FTDALS3
8	5	MONDO:0008781	juvenile amyotrophic lateral sclerosis with dementia
9	6	MONDO:0011632	amyotrophic lateral sclerosis type 21
10	2	MedGen:C4302169	Amyotrophic lateral sclerosis plus syndrome
11	3	UMLS:C2750729	Amyotrophic lateral sclerosis 6, autosomal recessive
12	3	UMLS:CN239196	Amyotrophic Lateral Sclerosis, Recessive
13	2	MedGen:CN260033	Amyotrophic lateral sclerosis 10, with or without FTD
14	4	ORPHA:52430	Inclusion body myopathy with Paget disease of bone and frontotemporal dementia
15	3	UMLS:C2931441	Infantile-onset ascending hereditary spastic paralysis
16	2	MedGen:C2931786	Amyotrophic lateral sclerosis, type 6
17	3	ORPHA:98756	Spinocerebellar ataxia type 2
18	2	MedGen:C3662062	Restrictive lung disease due to amyotrophic lateral sclerosis
19	3	MedGen:CN239175	Amyotrophic Lateral Sclerosis, Dominant
20	2	MedGen:C4551993	Amyotrophic Lateral Sclerosis, Familial
21	3	UMLS:CN239211	Amyotrophic Lateral Sclerosis/Frontotemporal Dementia
22	1	ClinVar:10103	Amyotrophic lateral sclerosis, typical
23	1	ClinVar:10438	Amyotrophic lateral sclerosis, susceptibility to
24	1	ClinVar:10387	Amyotrophic lateral sclerosis 13
25	1	ClinVar:32676	Amyotrophic lateral sclerosis 22 with frontotemporal dementia
26	1	MONDO:0008178	inclusion body myopathy with Paget disease of bone and frontotemporal dementia type 1
27	1	ClinVar:10104	Amyotrophic lateral sclerosis-parkinsonism/dementia complex 1, susceptibility to
28	1	ClinVar:16925	Amyotrophic lateral sclerosis 14 without frontotemporal dementia
29	1	HP:0007354	Amyotrophic lateral sclerosis

Table 5. Using the disNet to connect to ChEMBL results identifies compounds available for different disease identifiers listed here.

Disease identifier	Number of compounds
EFO:0000253	73
EFO:0001356	1
MeSH:D000690	74
ORPHA:98756	2

disease is missing in ChEMBL, namely the identifier “ORPHA:98756”. This term was identified as a child term of ‘EFO:0001356’ with the extension of the *disNet* using DODO and providing more granularity (Figure 12). The disease can not be identified using a free-text query in ChEMBL directly as it’s labelled ‘Spinocerebellar ataxia type 2’. However, while different resources such as Monarch Initiative and EFO report ALS as a parent term, it is unclear whether this can be considered as ALS disease. OMIM does report a genetic overlap between spinocerebellar ataxia and amyotrophic lateral sclerosis type 13. The information integrated in the DODO graph database is based on the original information provided by ontologies such as EFO and Mondo without any additional curation. Errors in disease definitions and provided mappings will inherently be present in these ontologies and will therefore persist within DODO as well. While it was not within the initial scope of DODO, it may help to assess and identify underlying issues present in the ontologies.

For the ClinVar resource associating gene variants to diseases, all 105 unique Entrez gene variants returned querying ClinVar directly for “amyotrophic lateral sclerosis” are also identified using a network of diseases as a query start. However, an additional variant was reported for “Inclusion body myopathy with early-onset Paget disease with or without frontotemporal dementia 2” (ClinVar:18286). This identifier was found by extending through cross-references edges using transitivity mapping starting from “ORPHA:52430” (“Inclusion body myopathy with Paget disease of bone and frontotemporal dementia” and “Pagetoid amyotrophic lateral sclerosis” as a synonym (Figure 13)). This disease identifier is not an actual ALS disease but rather another neurodegenerative disorder related to frontotemporal dementia. This example highlights the necessity of reviewing the queried disease not only when using a network of diseases, but the same need remains when querying resources directly. Using DODO, it is possible to use underlying disease relationships to cluster diseases and identify groups of identifiers with similar or related concepts (Table 4). Particular clusters that are outside of the scope can be dropped and a more precise network of diseases returned to connect to external knowledgebases. The reviewed network of diseases no longer includes identifiers outside of the scope and identifies the same set of 105 unique Entrez gene identifiers compared to querying the ClinVar resource directly. For ChEMBL, the results remain the same. The ability to apply use and review easily disease networks should facilitate the integration of biomedical resources.

Figure 12. The figure shows the relations between the different diseases with compounds available in ChEMBL resource. The term ‘ORPHA:98756’ was identified as a child term of ‘EFO:0001356’ through extension using *is_a* edge (green edges). Cross-reference edges are indicated in blue. The arrows on edges refer to the direction an edge can be traversed.

Figure 13. The identifier 'ClinVar:18286' ('Inclusion body myopathy with early-onset Paget disease with or without frontotemporal dementia 2') connects to 'ORPHA:52430' through the use of transitivity on cross-reference edges (label: 'Inclusion body myopathy with Paget disease of bone and frontotemporal dementia' synonym: 'Pagetoid amyotrophic lateral sclerosis'). The color of the nodes refers to the ontology, the *is_xref* edges are indicated in blue and *is_related* edges indicated in orange. The arrows on the edge refer to the backward ambiguity and show how an edge can be traversed. When no arrow is present, it can only be reached through the final step of conversion when no filtering is present.

```
#####@
## reviewing disNet
clDisNet <- cluster_disNet(disNet = extendedDisNet,
                           clusterOn = "xref")
explore_disNet(clDisNet)
clDisNet <- clDisNet[c(1:2, 4:6, 8:9, 13:20, 22, 24:28)]
fedisNet <- merge_disNet(list = clDisNet)
```

Tracing connections. Understanding the relation between disease identifiers obtained when querying a resource directly through a search term is not a trivial thing. The question remains whether these identifiers are dealing with the same disease concepts. An additional feature from connecting resources using a network of diseases, is the possibility to identify if and how diseases returned from each resource are connected to each other. This does not only allow a better understanding of disease, but also facilitates downstream analyses. Figure 14 shows the original ALS extended and reviewed network with disease identifiers matching in ChEMBL (orange) and those matching in ClinVar (blue). As both resources use different ontologies as references, there is a necessity to use cross-reference to understand their relationship to each other. Indirect relationships are used and recorded when extending and can facilitate the understanding and integration of different biological resources.

Conclusions

Disease ontologies have allowed a more formal classification of diseases. They facilitate the integration of biological databases thereby increasing disease information usage and supporting the development of novel treatments. However, efforts to integrate biological databases or the ontologies directly are complicated by ontology-specific identifiers, heterogeneous decisions on disease definitions, and the inherent presence of errors. Despite ongoing integration efforts, we identified two remaining challenges that prevent seamless integration of different databases based on disease ontologies:

- Currently no resource provides a flexible and complete mapping across the multitude of disease ontologies
- There is no software available to comprehensively explore and interact with disease ontologies

Figure 14. The figure shows the extended network of ALS constructed in DODO. The colour of the nodes refers to the disease identifiers that are also identified through a direct query in ChEMBL (orange) and ClinVar (blue). The edges between the nodes capture the *is_xref* and *is_related* relationship in blue and orange respectively.

DODO aims to tackle these two challenges by constructing a meta-database containing information on disease identifiers and their relationships across different ontologies. Through well-defined and controlled transitivity mechanisms, the combined information across resources can be used dynamically to identify indirect cross-references. The R package contains several functions to build and interact with disease networks or convert concept identifiers between ontologies. The workflow to construct a custom, local DODO database is provided with the intent to allow adaptation. A docker image with the presented ontologies is provided for convenience.

DODO helps clarifying and defining conditions of interest in addition to help in the understanding of relationships between disease concepts. It improves accessibility of disease ontologies for a standard user. In addition, connecting different biomedical knowledge resources through a disease network facilitates the integration of all this information. It also ensures these resources are queried transparently using equivalent identifiers of the disease of interest. In addition, it also allows visualizing the connection between these resources directly.

Through the aggregation of different ontologies and their mappings, DODO facilitates the generation of exhaustive descriptions of disease landscapes. The code to build and query DODO is provided under open source license to allow further improvement by other developers.

Data availability

Underlying data

All data underlying the results are available as part of the article and no additional source data are required.

Extended data

Zenodo: Extended data for publication “Dictionary of disease ontologies (DODO): a graph database to facilitate access and interaction with disease and phenotype ontologies”, <https://doi.org/10.5281/zenodo.3922210>¹⁶

This project contains the following extended data:

- **Table 1:** List of all functions available in DODO R package with description and scope details.
- **Table 2:** List of ontologies among which the cross-reference relations are encoded as is_xref

Data are available under the terms of the [Creative Commons Attribution 4.0 International license](#) (CC-BY 4.0).

Software availability

Source code for DODO is available at: <https://github.com/Elysheba/DODO>

Archived source code as at time of publication: <https://doi.org/10.5281/zenodo.3922143>³³

License: GPL-3 license

Docker image of the DODO Neo4j instance is available at: <https://hub.docker.com/repository/docker/elysheba/dodo> (tag:02.04.2020)

Archived database docker image as at time of publication: <http://doi.org/10.5281/zenodo.3921874>³⁴

Source code (and archived source code) for parsing disease ontologies are available in [Table 1](#).

References

- Gruber TR: **A Translation Approach to Portable Ontology Specifications.** *Knowl Acquis.* 1993; 5(2): 199–220.
[Publisher Full Text](#)
- Haendel MA, Mcmurry JA, Relevo R, *et al.*: **A Census of Disease Ontologies.** *Annu Rev Biomed Data Sci.* 2018; 1: 305–331.
[Publisher Full Text](#)
- Hoehndorf R, Dumontier M, Gkoutos GV: **Evaluation of research in biomedical ontologies.** *Brief Bioinform.* 2013; 14(6): 696–712.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Hasnain A, Kamdar MR, Hasapis P, *et al.*: **Linked biomedical dataspace: Lessons learned integrating data for drug discovery.** *Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics).* 2014; 8796: 114–130.
[Publisher Full Text](#)
- Kibbe WA, Arze C, Felix V, *et al.*: **Disease Ontology 2015 update: An expanded and updated database of Human diseases for linking biomedical knowledge through disease data.** *Nucleic Acids Res.* 2015; 43(Database issue): D1071–D1078.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Livingston KM, Bada M, Baumgartner WA, *et al.*: **KaBOB: ontology-based semantic integration of biomedical databases.** *BMC Bioinformatics.* 2015; 16(1): 126.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Malone J, Holloway E, Adamusiak T, *et al.*: **Modeling sample variables with an Experimental Factor Ontology.** *Bioinformatics.* 2010; 26(8): 1112–1118.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Rappaport N, Nativ N, Stelzer G, *et al.*: **MalaCards: An integrated compendium for diseases and their annotation.** *Database (Oxford).* 2013; 2013: bat018.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Hu W, Qiu H, Huang J, *et al.*: **BioSearch: a semantic search engine for Bio2RDF.** *Database (Oxford).* 2017; 2017: bax059.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Mungall CJ, McMurry JA, Kohler S, *et al.*: **The Monarch Initiative: An integrative data and analytic platform connecting phenotypes to genotypes across species.** *Nucleic Acids Res.* 2017; 45(D1): D712–D722.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Shefchek KA, Harris NL, Gargano M, *et al.*: **The Monarch Initiative in 2019: an integrative data and analytic platform connecting phenotypes to genotypes across species.** *Nucleic Acids Res.* 2020; 48(D1): D704–D715.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Cheng L, Wang G, Li J, *et al.*: **SIDD: A Semantically Integrated Database towards a Global View of Human Disease.** *PLoS One.* 2013; 8(10): e75504.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Schriml LM, Mitraka E: **The Disease Ontology: fostering interoperability between biological and clinical human disease-related data.** *Mamm Genome.* 2015; 26(9–10): 584–589.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- Yu G, Wang LG, Yan GR, *et al.*: **DOSE: An R/Bioconductor package for disease ontology semantic and enrichment analysis.** *Bioinformatics.* 2015; 31(4): 608–609.
[PubMed Abstract](#) | [Publisher Full Text](#)
- Saqi M, Lysenko A, Guo YK, *et al.*: **Navigating the disease landscape: Knowledge representations for contextualizing molecular signatures.** *Brief Bioinform.* 2019; 20(2): 609–623.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
- François L: **Extended data for publication “Dictionary of disease ontologies (DODO): a graph database to facilitate access and interaction with disease and phenotype ontologies” [Data set].** Zenodo. 2020.
<http://www.doi.org/10.5281/zenodo.3922210>
- Docker Inc: **Docker Community Edition.** 2019.
[Reference Source](#)
- Neo4J Inc: **Neo4j Community Edition.** 2020.
[Reference Source](#)
- R Core Team: **A Language and Environment for Statistical**

- Computing. 2019.
[Reference Source](#)
20. Wickham H, François R, Henry L, *et al.*: **dplyr: a grammar of data manipulation**. 2019.
[Reference Source](#)
 21. Müller K, Wickham H: **tibble: simple data frames**. 2019.
[Reference Source](#)
 22. Godard P, van Eyll J: **BED: A Biological Entity Dictionary based on a graph data model [version 3; peer review: 2 approved]**. *F1000Res*. 2018; 7: 195.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
 23. Ren KK: **rlist: a toolbox from non-tabular data manipulation**. 2016.
[Reference Source](#)
 24. Wickham H: **stringr: simple, consistent wrappers for common string operations**. 2019.
[Reference Source](#)
 25. Wickham H, Hester J, François R: **readr: Read Rectangular Text Data**. 2018.
[Reference Source](#)
 26. Almende BV, Thieurmél B, Robert T: **visNetwork: network visualization using vis.js library**. 2019.
[Reference Source](#)
 27. Chang W: **shinythemes: themes for shiny**. 2018.
[Reference Source](#)
 28. Xie Y, Cheng J, Tan X: **DT: a wrapper for the JavaScript Library "DataTables"**. 2019.
[Reference Source](#)
 29. Csardi G, Nepusz T: **The igraph software package for complex network research**. *InterJournal. Complex Systems*, 1695. 2006.
[Reference Source](#)
 30. Chang W, Cheng J, Allaire JJ, *et al.*: **shiny: Web Application Framework for R**. 2019.
[Reference Source](#)
 31. Landrum MJ, Lee JM, Benson M, *et al.*: **ClinVar: Improving access to variant interpretations and supporting evidence**. *Nucleic Acids Res*. 2018; 46(D1): D1062–D1067.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
 32. Mendez D, Gaulton A, Bento AP, *et al.*: **ChEMBL: Towards direct deposition of bioassay data**. *Nucleic Acids Res*. 2019; 47(D1): D930–D940.
[PubMed Abstract](#) | [Publisher Full Text](#) | [Free Full Text](#)
 33. François L: **Elysheba/DODO: publication (v1) release**. *Zenodo*. 2020.
 34. François L: **docker-ucb-public-dodo-20.04.2020 (version 20/04/2020)**. *Zenodo*. 2020.

Open Peer Review

Current Peer Review Status:

Version 1

Reviewer Report 14 October 2022

<https://doi.org/10.5256/f1000research.27748.r152033>

© 2022 Frey J et al. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Jeremy G. Frey

School of Chemistry, University of Southampton, Southampton, UK

Samantha Kanza

Department of Chemistry, Faculty of Engineering and Physical Sciences, University of Southampton, Southampton, UK

We like the paper from a technical perspective. The main feedback would be that it lacks some of the real world application and needs a bit more explanation around that.

The paper gives you a lot of info about the system technically does but doesn't really explain why this will be useful in practice.

Overall this is an interesting well written paper, with some well documented data and code sources. I would suggest accept with minor revisions which would add some additional value.

As this is a cross domain paper, it is particularly important to make it accessible to researchers from these different domains. As such, it would be useful to include a simple explanation of how transitive mappings work. It would also be interesting to either see an example of a similar approach to this paper, or if this does not exist state that, and explain how and why you took this approach.

This paper goes into a lot of good technical detail about how DODO was created, but it could use a bit more of an explanation about the why earlier on in the paper. You've covered this from a technical perspective, in that there is no software available to explore these ontologies as a whole, this would create a more flexible landscape etc but haven't noted why this is beneficial outside of this technical remit. What could a researcher potentially do with your tool that they couldn't do before? How will this benefit the disease community from a real world impact scenario? You mention this briefly right at the end in the conclusion, but it would be good to see these motivations stated up front.

On a similar note, I like that you have included use cases to demonstrate how researchers would

use your application. However, it would be useful to see these linked back to more specific real world examples. If you convert disease and phenotype identifiers between ontologies WHY is this useful? HOW will this enable researchers to develop more novel treatments in a way that they couldn't do before?

There are a few areas that need some minor grammatical changes / tense changes

1. DODO combines the information provided by the different ontologies and allows connecting ontologies to one another, even if they don't have direct cross-references — weird sentence, would suggest changing allows to enables?
2. Structuring and harmonize the information derived from each ontology - changed tense within the sentence. Structure and harmonize.
3. This is exemplified in Figure 3, where transitivity mappings is needed to connect the initial MONDO identifier - mappings ARE needed.

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?

Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Digital chemistry, Semnatic Web

We confirm that we have read this submission and believe that we have an appropriate level of expertise to confirm that it is of an acceptable scientific standard.

Reviewer Report 20 November 2020

<https://doi.org/10.5256/f1000research.27748.r73679>

© 2020 Vasilevsky N. This is an open access peer review report distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nicole Vasilevsky

Oregon Clinical & Translational Research Institute, Oregon Health & Science University, Portland, OR, USA

The authors describe the development of the Dictionary of Disease Ontology database that provides mappings across disease ontologies and an R package that allows users to interact with the data.

Introduction:

- Mondo is now formatted as Mondo, not MonDO (see <https://mondo.monarchinitiative.org/>). You should define the abbreviations: OMIM, NCIt, etc. (Note: NCIt is formatted NCIt, not NCIT).
- References 12 and 14 are not appropriate references for the Disease Ontology.
- This sentence "While efforts such as MonDO and EFO try to integrate different disease ontologies through semantic learning and manual curation, these resources, like the different disease ontologies themselves, are currently not providing a complete mapping across all diseases ontologies" references papers about MalaCards and BioSearch, not Mondo or EFO.

Methods:

- "Most mappings are unambiguous (one concept in an ontology is related to only one concept in another ontology); however, some concepts map to many similar concepts within the same ontology. This is conceptually visualized in Figure 3 where the MONDO identifier maps to 2 ORPHA identifiers."
An important feature of Mondo is that it contains semantics with the database cross references (dbxrefs), that include precise inter-ontology relationships in the form of OWL equivalence axioms, and these relationships have the properties of symmetry and transitivity. The dbxrefs in Mondo have annotations including MONDO:equivalentTo or MONDO:relatedTo, which indicate the relationship of the referenced term to the Mondo term. If the dbxref lacks an equivalence or relatedTo axiom, it is not intended to be interpreted as such.
- In your example in Figure 3, you state that Mondo maps to 2 Orphanet identifiers. There are many cases in Mondo where terms cross-reference more than one Orphanet term, but the semantics are defined. In general, Mondo does not have equivalence axioms to 2 Orphanet terms (with the rare exception of proxy merges, where we decide that two concepts in an external resource mean the same thing. In this case, we work with the source ontology (such as Orphanet) to resolve this as best as possible.) More commonly, there are dbxrefs to more than one Orphanet term where one term is equivalent, and another defined as a subclassOf or superClassOf. For example, MONDO_0011822 'Bartter disease type 3' (Orphanet:112 has the axiom annotation MONDO:subclassOf and Orphanet:93605 has the

axiom annotation MONDO:equivalentTo). Further description of axiom annotation is available here: <https://mondo.readthedocs.io/en/latest/editors-guide/f-entities/#axiom-annotations>

- In your example in Figure 4, ICD:87.0 is not an equivalent mapping to MONDO_0010561 Coffin-Lowry syndrome, and is annotated with the source Orphanet:192, meaning this mapping came from Orphanet. The semantics of the Orphanet mappings are not defined, and should not be interpreted as equivalence mappings.
- Figure 4:
You are missing some xrefs from Mondo to other sources, see: <https://www.ebi.ac.uk/ols/ontologies/mondo/terms?iri=http%3A%2F%2Fpurl.obolibrary.org%2Fobo%2FM>
Please see this publication on the algorithm used to develop Mondo: <https://www.biorxiv.org/content/10.1101/048843v3>
- Figure 6: The title should say ontology = Mondo (Monarch is not an ontology).

Use Cases:

- Use case 2: Strict indirect conversion
"As there is no direct mapping provided by the resources between Mondo and Orphanet, transitive mapping provided by DODO needs to be applied."
This is because Orphanet does not have a class for 'epilepsy'. ORPHA:166463 is 'epilepsy syndrome', which is mapped (MONDO:equivalentTo) to a more granular class (subclass) in Mondo: MONDO_0015650 'epilepsy syndrome'.
- ORPHA:101993 does not return any results in Orphanet:
<https://www.ebi.ac.uk/ols/search?q=orphanet%3A101993&groupField=iri&start=0&ontology=ordo>
(or via searching here: https://www.orpha.net/consor/cgi-bin/Disease_Search_Simple.php?lng=EN). I believe this is meant to be ORPHA:101998 (not ORPHA:101993). We obsoleted this term in Mondo and merged it with MONDO:0005027, as Mondo does not include "rare" disease terms (see: <https://github.com/monarch-initiative/mondo/issues/254>).
- "...the majority of Mondo identifiers cannot be converted to EFO or MeSH identifiers." EFO and MESH are broader than Mondo and Mondo never intended to fully map to these two terminologies. EFO does not import the entirety of Mondo, EFO only imports relevant disease terms from Mondo into their disease hierarchy that are needed for their applications.
- Figure 9:
"Looking into this identifier in more detail shows that it only has six direct cross-references among which one DO identifier (DOID:0050564) (Figure 9)."
Mondo only xrefs 4 sources as equivalent terms for this term, see <https://www.ebi.ac.uk/ols/ontologies/mondo/terms?iri=http%3A%2F%2Fpurl.obolibrary.org%2Fobo%2FM>
Your figure shows relationships to 7 terms, not 6.
MONDO:0019587 does not xref OMIM:618533, there is a superclassOf relationship between those two terms.

- Figure 10:
KEGG:05014 is equivalent to MONDO:0004976 in Mondo.
- Figure 12:
The labels are cut off, please revise this figure so the labels are displayed properly.
It is my understanding that you are saying Mondo does not classify 'spinocerebellar ataxia type 2' as a child of 'familial amyotrophic lateral sclerosis' and 'familial amyotrophic lateral sclerosis' as child of 'amyotrophic lateral sclerosis', but this is not the case. This hierarchy is inferred in Mondo, I recommend you view the OWL file with the reasoner turned on and you can see this hierarchy (The Mondo OWL file is available here: <https://github.com/monarch-initiative/mondo>).
- Figure 13:
The labels are cut off or missing, please review this figure so the labels are displayed properly.
- Conclusions
Is this resource currently being used by the community? Please include any description of community use and evaluation.

Data availability

- The link to the docker hub returns a 404 error:
<https://hub.docker.com/repository/docker/elysheba/dodo>
- I was able to access the Docker image via Zenodo, minor comments:
I believe you are referring to the Mondo disease ontology, not the Monarch Ontology.
- HPO is not a disease ontology per se, it is an ontology of abnormal human phenotypes encountered in human diseases.

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

Yes

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?

Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?

Partly

Competing Interests: I am the primary curator for the Mondo Disease Ontology.

Reviewer Expertise: Mondo disease ontology, biomedical ontology, biocuration

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

Author Response 17 Dec 2020

Liesbeth François

Thank you for providing your comments to the manuscript "Dictionary of disease ontologies (DODO): a graph database to facilitate access and interaction with disease and phenotype ontologies", please find our replies to your comments and questions below. We would like to address your different comments and ask if you have any additional comments or thoughts. We will adapt our manuscript as described below with the receipt of the second review so that we can address both reviews equally.

We would also precise more generally that DODO aims to connect the information from different disease ontologies. The information from these different sources is considered equally without any priority among them. In contrast with the disease ontologies themselves that aim to harmonize and structure diseases themselves, DODO aims to facilitate the connection of different biomedical resources by relying on these different ontologies to support downstream analyses. DODO provide several parameters to flexibly obtain the most relevant identifiers for the use in mind.

Introduction:

- **Mondo is now formatted as Mondo, not MonDO (see <https://mondo.monarchinitiative.org/>). You should define the abbreviations: OMIM, NCIt, etc. (Note: NCIt is formatted NCIt, not NCiT).**

Thanks you for your comment, the abbreviations used within the document will be verified and corrected. We will also include a definitions of the abbreviations.

- **References 12 and 14 are not appropriate references for the Disease Ontology.**

Thanks for you comment, we will remove these references.

- **This sentence "While efforts such as MonDO and EFO try to integrate different disease ontologies through semantic learning and manual curation, these resources, like the different disease ontologies themselves, are currently not providing a complete mapping across all diseases ontologies" references papers about MalaCards and BioSearch, not Mondo or EFO.**

We apologize for this error, F1000 Research request weblinks and URLs to be added

as a hyperlink to the manuscript (<https://f1000research.com/for-authors/article-guidelines/software-tool-articles>, section 13). However, during the manuscript preparation the weblinks for Mondo and EFO were removed from the manuscript by accident. We will amend this error and link to webpage of Mondo Disease Ontology (<https://mondo.monarchinitiative.org/>) and EFO (<http://blog.opentargets.org/2019/12/19/efo3-a-community-driven-ontology-to-advance-clinical-discoveries/>) which lists the resources that are currently included into these resources. The references to Malacards (Rappaport et al. 2013) and BioSearch (Hu et al. 2017) highlight more generally the issues in the completeness of mappings arising across the multitude of disease ontologies that exist. To address this potential confusion, we will change this sentence as follows: "The connections provided by existing disease ontologies are not a complete mapping across different ontologies, complicating the integration of biomedical resources (Hu et al. 2017, Rappaport et al. 2013). Similarly, ongoing efforts such as **Mondo** and **EFO** that aim to integrate different disease ontologies through semantic learning and manual curation, are currently not yet providing a complete mapping across all diseases ontologies."

Methods:

- **"Most mappings are unambiguous (one concept in an ontology is related to only one concept in another ontology); however, some concepts map to many similar concepts within the same ontology. This is conceptually visualized in Figure 3 where the MONDO identifier maps to 2 ORPHA identifiers."**

An important feature of Mondo is that it contains semantics with the database cross references (dbxrefs), that include precise inter-ontology relationships in the form of OWL equivalence axioms, and these relationships have the properties of symmetry and transitivity. The dbxrefs in Mondo have annotations including MONDO:equivalentTo or MONDO:relatedTo, which indicate the relationship of the referenced term to the Mondo term. If the dbxref lacks an equivalence or relatedTo axiom, it is not intended to be interpreted as such.

Thanks for your clarification. This figure was created to conceptually explain the notion of ambiguity and does not capture a true example. We will adapt the figure to avoid the implication that these semantics are not considered in Mondo.

Conceptually, it is sometimes possible that multiple nodes of one ontology are related through a cross-reference edge to one identifier in another ontology. The reason is that DODO uses the information provided by different ontologies which may provide additional mappings between identifiers. This is captured by the ambiguity property in the data model.

- **In your example in Figure 3, you state that Mondo maps to 2 Orphanet identifiers. There are many cases in Mondo where terms cross-reference more than one Orphanet term, but the semantics are defined. In general, Mondo does not have equivalence axioms to 2 Orphanet terms (with the rare exception of proxy merges, where we decide that two concepts in an external resource mean the same thing. In this case, we work with the source ontology (such as Orphanet) to resolve this as best as possible.) More commonly, there are dbxrefs to more than one Orphanet term where one term is equivalent, and another**

defined as a subClassOf or superClassOf. For example, MONDO_0011822 'Bartter disease type 3' (Orphanet:112 has the axiom annotation MONDO:subClassOf and Orphanet:93605 has the axiom annotation MONDO:equivalentTo). Further description of axiom annotation is available here:

<https://mondo.readthedocs.io/en/latest/editors-guide/f-entities/#axiom-annotations>

Figure 3 is meant to provide a conceptual example to introduce and explain the ambiguity property and does not necessarily capture a true example. We will adapt the figure to avoid the implication that these semantics are not considered in Mondo. Additional cross-reference edges to a node may be derived from other ontologies as DODO processes and contains the information provided by different ontologies. DODO aims to facilitate access and exploration of many different disease ontologies. It doesn't aim to curate these resources but rather build upon the extensive efforts already performed by the ontologies themselves. However, this may result in additional mappings between identifiers which is captured by the ambiguity property in the data model.

- **In your example in Figure 4, ICD:87.0 is not an equivalent mapping to MONDO_0010561 Coffin-Lowry syndrome, and is annotated with the source Orphanet:192, meaning this mapping came from Orphanet. The semantics of the Orphanet mappings are not defined, and should not be interpreted as equivalence mappings.**

DODO processes multiple ontologies as a way to enrich mappings with the aim to facilitate connecting various biomedical databases. Figure 4 aimed to highlight the ambiguity derived from some mappings, in this case the ICD10 identifier Q87.0 in this disease network around Coffin-Lowry syndrome. Using transitivity mappings, the property of ambiguity is required to carefully consider which mappings to include. This ambiguity in mappings derived from different ontologies (DODO processes information from 9 different resources) and it is related to the different way disease definition are defined and the precision of the different cross-reference edges.

- **Figure 4:**

You are missing some xrefs from Mondo to other sources, see:

<https://www.ebi.ac.uk/ols/ontologies/mondo/terms?iri=http%3A%2F%2Fpurl.obolibrary.org%2F>

Please see this publication on the algorithm used to develop Mondo:

<https://www.biorxiv.org/content/10.1101/048843v3>

For visibility, only a part of this disease network is shown in Figure 4. This figure aimed to introduce an example to show the issue of ambiguity of some cross-references edges (ICD10:Q87.0 in this figure) that related to a high number of mappings. We will clarify in the manuscript that the Figure 4 represent only a part of the disease network for visibility reasons.

- **Figure 6: The title should say ontology = Mondo (Monarch is not an ontology).**

Thanks you for your comment, we will correct the title to represent to correct ontology name.

Use Cases:

- **Use case 2: Strict indirect conversion**

"As there is no direct mapping provided by the resources between Mondo and Orphanet, transitive mapping provided by DODO needs to be applied."
This is because Orphanet does not have a class for 'epilepsy'. ORPHA:166463 is 'epilepsy syndrome', which is mapped (MONDO:equivalentTo) to a more granular class (subclass) in Mondo: MONDO_0015650 'epilepsy syndrome'.

Indeed, there is no direct cross-reference relationship as Orphanet relates to rare disorders. However, dependent on the use case, DODO allows to only use direct cross-references (to stay close to the original level) or include indirect relationships to connect various biomedical databases. These databases may not consider disease in the same way but connecting these resources may be required to perform downstream analyses. This conversion or extension is not trivial and we aimed to showcase the different possibilities within the use cases in the manuscript. Important to note is that while ontologies, such as Mondo, aim to harmonize disease definitions, DODO wants to connect the information from different ontologies with the aim to provide a flexible way to connect multiple biomedical resources that can be used for downstream analyses. There are different parameters available within the conversion and extension functions to obtain the relevant disease network or conversion that can be adapted based on the need. An example is using transitivity to connect biomedical resources that are not connected using first level cross-reference edges (different use cases for indirect conversion). However, while DODO provides a default setting, the definition of these parameters is left to the user's requirements.

- **ORPHA:101993 does not return any results in Orphanet:**

<https://www.ebi.ac.uk/ols/search?q=orphanet%3A101993&groupField=iri&start=0&ontology=or>
(or via searching here: https://www.orpha.net/consor/cgi-bin/Disease_Search_Simple.php?lng=EN). I believe this is meant to be ORPHA:101998 (not ORPHA:101993). We obsoleted this term in Mondo and merged it with MONDO:0005027, as Mondo does not include "rare" disease terms (see: <https://github.com/monarch-initiative/mondo/issues/254>).

Indeed, it should refer to ORPHA:101998 instead of ORPHA:101993, we will adapt this in the manuscript. The two Orphanet identifiers were not directly derived from Mondo ontology, rather they are still retained within the MedGen ontology which provides mappings between UMLS:C0014544 and ORPHA:101998 and ORPHA:166463, as can be seen in Figure 7. This example additionally shows that different ontologies provide different definitions, curation steps and rules for defining cross-reference mappings. DODO doesn't aim to perform any curation steps on the mapping themselves, but rather build onto the extensive efforts performed by each ontology. While disease ontologies aim to provide a harmonization of disease definitions, DODO uses this information to build a more enriched disease mapping and facilitate the connection of different biomedical databases that can be used for downstream

analyses. In addition, the exploration and visualization of disease networks may highlight potential errors in the original ontologies and their provided mappings. In such case, this potential error is reported to the ontology for revision.

- **"...the majority of MonDO identifiers cannot be converted to EFO or MeSH identifiers." EFO and MESH are broader than Mondo and Mondo never intended to fully map to these two terminologies. EFO does not import the entirety of Mondo, EFO only imports relevant disease terms from Mondo into their disease hierarchy that are needed for their applications.**

The ontologies of EFO and MeSH indeed extend beyond diseases only. However, the information processed from these resources in the scope of DODO only considers identifiers related to disease from EFO and MeSH. We will add this clarification to the manuscript, by adding the following sentence to the section discussion the feeding of the database: "Some ontologies such as [EFO](<https://www.ebi.ac.uk/efo/faq.html#whatisefo>) and [MeSH](https://www.nlm.nih.gov/mesh/qualifiers_scopenotes.html) extend their ontology to include information like anatomy, disease and chemical compounds. In the scope of DODO only the relevant information on disease identifiers is extracted from these ontologies.". However, converting all MonDO identifiers to EFO/MeSH (disease) ontology using the different possible parameters available for conversion, only a smaller subset of Mondo identifiers could be converted to a EFO/MeSH disease term. This may be a result of different ontologies that use different disease definitions, curation steps and rules for defining cross-reference mappings.

- **Figure 9:**
"Looking into this identifier in more detail shows that it only has six direct cross-references among which one DO identifier (DOID:0050564) (Figure 9)."
Mondo only xrefs 4 sources as equivalent terms for this term, see
<https://www.ebi.ac.uk/ols/ontologies/mondo/terms?iri=http%3A%2F%2Fpurl.obolibrary.org%2F>
Your figure shows relationships to 7 terms, not 6.
MONDO:0019587 does not xref OMIM:618533, there is a superclassOf
relationship between those two terms.

The Mondo ontology indeed only reports 5 cross-reference mappings (including Orphanet:90635). However, the mappings between MONDO:0019587 and OMIM:618533 is in this case provided by the EFO ontology, which adds this additional mapping for the MONDO:0019587 that they have integrated into the EFO ontology (<https://www.ebi.ac.uk/ols/ontologies/efo/terms?iri=http%3A%2F%2Fpurl.obolibrary.org%2Fobo%2F>). DODO processed 9 different resources equally and their provided cross-reference information. It uses this information to build a more enriched disease mapping. The primary aim is to facilitate the connection of different biomedical databases that can be used for downstream analyses.

- **Figure 10:**
KEGG:05014 is equivalent to MONDO:0004976 in Mondo.

Thanks for your comment, it is possible that you refer to Figure 11 instead? This disease network indeed shows the cross-reference mappings between MONDO:0004976 and KEGG:05014. In addition to the ambiguity property, two types of cross-reference edges are defined: *is_xref* and *is_related*. Briefly, an *is_xref* edges is used to define equal cross-reference relationships between ontologies are more or less similar in concept definition. The exploration of this threshold is based on 14 different ontologies that are frequently used and trusted by the authors. This type of cross-reference edge is used in obtaining all transitive mappings. An *is_related* edges is used for all other cross-reference edges and this relationship is not used in transitivity mappings directly. Rather, it is only used in the final step of conversion after transitivity mapping to obtain the direct relationships of the converted nodes or when applying the direct conversion (use case 1). As KEGG ontology is not considered when exploring the *is_xref* threshold setting, all cross-reference relationships which consider a KEGG identifier, are classified as *is_related* edges.

- **Figure 12:**
The labels are cut off, please revise this figure so the labels are displayed properly.
It is my understanding that you are saying Mondo does not classify 'spinocerebellar ataxia type 2' as a child of 'familial amyotrophic lateral sclerosis' and 'familial amyotrophic lateral sclerosis' as child of 'amyotrophic lateral sclerosis', but this is not the case. This hierarchy is inferred in Mondo, I recommend you view the OWL file with the reasoner turned on and you can see this hierarchy (The Mondo OWL file is available here: <https://github.com/monarch-initiative/mondo>).

Thanks for your remark, we will revise the figure to display the full label. We recognize that the explanation in the manuscript can be confusing as indeed "spinocerebellar ataxia type 2" (SCA2) is classified as a child of "familial amyotrophic lateral sclerosis" which in term is a child of "amyotrophic lateral sclerosis". This relationship in is indeed reported in EFO. As we focus on integrating information from ClinVar and ChEMBL, we only consider the relevant identifiers in this context. While this relationship is reported in Mondo, this ontology is not used to integrate the information from ChEMBL and ClinVar. We will amend the sentence to better clarify our meaning to "However, while EFO reports 'familial amyotrophic lateral sclerosis' as a parent term, it is unclear whether this can be considered as ALS disease clinically".

- **Figure 13:**
The labels are cut off or missing, please review this figure so the labels are displayed properly.

Thanks for your remark, we will revise the figure to display the full label.

- **Conclusions**
Is this resource currently being used by the community? Please include any description of community use and evaluation.

This publication coincided with the first release of the database and R package for the community. We hope that the open access sharing of resource will allow and facilitate access and usage of the disease ontology resources by the community. As of yet, we have not seen any publications referring to the usage of DODO by the community as DODO has only recently been released for public usage. However, DODO is being used internally to integrating different biomedical resources and with this publication we wanted to take the opportunity to share our work with the community.

Data availability

- **The link to the docker hub returns a 404 error:**
<https://hub.docker.com/repository/docker/elysheba/dodo>

We apologize for the incorrect link, the link to docker hub was corrected (<https://hub.docker.com/repository/docker/elysheba/public-dodo>) in the manuscript.

- **I was able to access the Docker image via Zenodo, minor comments:**
I believe you are referring to the Mondo disease ontology, not the Monarch Ontology.

Thanks for your comment, we will correct the ontology name to Mondo disease ontology in both the docker hub (<https://hub.docker.com/repository/docker/elysheba/public-dodo>) and zenodo archive ([10.5281/zenodo.3921874](https://zenodo.org/record/3921874))

- **HPO is not a disease ontology per se, it is an ontology of abnormal human phenotypes encountered in human diseases.**

Indeed, we have specified this distinction more clearly in both the docker hub repository and the Zenodo archive that HPO refers to a phenotype ontology and is not a disease ontology as such.

Competing Interests: No competing interests were disclosed.

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

F1000Research