The Phenoscape Knowledgebase: tools and APIs for computing across phenotypes from evolutionary diversity and model organisms

James P. Balhoff
RTI International
Research Triangle Park, NC, USA
jbalhoff@rti.org

Phenoscape project team
http://phenoscape.org/wiki/Acknowledgments

I. Introduction

The Phenoscape Knowledgebase (KB) is an ontology-driven database that combines existing phenotype annotations from model organism databases with new phenotype annotations from the evolutionary literature. Phenoscape curators have created phenotype annotations for more than 5,000 species and higher taxa, by defining computable phenotype concepts for more than 20,000 character states from over 160 published phylogenetic studies. These phenotype concepts are in the form of Entity–Quality (EQ) [1] compositions which incorporate terms from the Uberon anatomy ontology, the Biospatial Ontology (BSPO), and the Phenotype and Trait Ontology (PATO). Taxonomic concepts are drawn from the Vertebrate Taxonomy Ontology (VTO). This knowledge of comparative biodiversity is linked to potentially relevant developmental genetic mechanisms by importing associations of genes to phenotypic effects and gene expression locations from zebrafish (ZFIN [2]), mouse (MGI [3]), Xenopus (Xenbase [4]), and human (Human Phenotype Ontology project [5]). Thus far, the Phenoscape KB has been used to identify candidate genes for evolutionary phenotypes [6], to match profiles of ancestral evolutionary variation with gene phenotype profiles [7], and to combine data across many evolutionary studies by inferring indirectly asserted values within synthetic supermatrices [8]. Here we describe the software architecture of the Phenoscape KB, including data ingestion, integration of OWL reasoning, web service interface, and application features (Fig. 1).

II. Data ingestion and inference materialization

Phenoscape curators annotate character by taxon matrices associated with phylogenetic publications using the Phenex software [9], following the curation process described by Dahdul et al. [10]. The resulting NeXML data files, along with tab-delimited data dumps obtained from model organism databases, are translated into OWL models (including definitions of EQ phenotypes) using the Phenoscape data ingest pipeline (phenoscape-owl-tools, available with all other described software under an MIT open source license in the Phenoscape GitHub repository, https://github.com/phenoscape). OWL transformation code for individual data sources is kept concise and readable by using a Scala-based domain specific language for OWL axioms (Scowl) [11]. Next, all OWL data files, along with referenced ontologies, are processed to standardize property IRIs; in many cases community ontologies use non-standard or competing IRIs for the "same" property, e.g. our system accounts for ten variants of the *part_of* property. We rename these, rather than assert equivalence axioms, to simplify downstream reasoning and querying.

Based on the content of the data and reference ontologies, several "ontologies" are programmatically generated, for the purpose of precomputing inferred concept hierarchies supporting various Phenoscape use cases: 1) materialization of the transitive closure for selected properties; 2) generation of grouping phenotypes for semantic similarity queries; 3) generation of "absence" concepts for custom negation reasoning. The ELK reasoner [12] is then used to compute the inferred classification hierarchy, and inferred subclass axioms are materialized into concrete assertions. Because ELK does not support negation reasoning, we have also implemented a custom procedure to compute a class hierarchy for a predetermined set of negations [13]. To make reasoning on this large dataset feasible, we extract only the class axioms (Tbox) for input into ELK. This is sufficient for our purposes since most of the data is in the form of compositional class expressions; however it does restrict use cases that would rely on inference of property assertions or instance classification. All data, including asserted and inferred class axioms as well as instance data, are loaded into the Blazegraph RDF triple store [14], constituting approximately 100 million RDF triples. A separate OWL file including only class axioms (~870,000 logical axioms) is saved for later use in reasoner queries.

III. Semantic similarity precomputation

Phenotypic profiles for evolutionarily variable taxonomic nodes and model organism genes are computed as described in [7], and loaded into Blazegraph. The inferred phenotype class hierarchy is used to precompute semantic similarity scores between all pairs of evolutionary variation profiles and gene phenotype profiles, using an information content-based metric. The set of pair comparisons is broken into chunks and processed in parallel on a compute cluster to reduce computation time. The resulting set of similarity scores, along with computed statistical support, is loaded into Blazegraph, resulting in a final triple store totaling approximately 300 million RDF triples.

IV. Web service interfaces

Access to data in the Phenoscape KB is provided by two web service applications, implemented in Scala using the Spray HTTP toolkit [15]. The first, Owlery, provides a generic JSON-

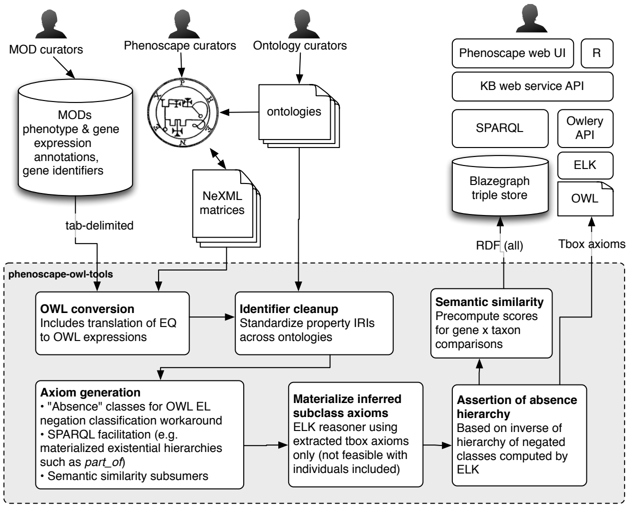
MOD curators
Phenoscape curators
Ontology curators
MODs phenotype & gene expression annotations, gene identifiers
ontologies
NeXML matrices
tab-delimited
Phenoscape web UI
R
KB web service API
SPARQL
Owlery API
ELK
OWL
Blazegraph triple store
RDF (all)
Tbox axioms
phenoscape-owl-tools
OWL conversion
Includes translation of EQ to OWL expressions
Identifier cleanup
Standardize property IRIs across ontologies
Semantic similarity
Precompute scores for gene x taxon comparisons
Axiom generation
• "Absence" classes for OWL EL negation classification workaround
• SPARQL facilitation (e.g. materialized existential hierarchies such as part_of)
• Semantic similarity subsumers
Materialize inferred subclass axioms
ELK reasoner using extracted tbox axioms only (not feasible with individuals included)
Assertion of absence hierarchy
Based on inverse of hierarchy of negated classes computed by ELK

Fig. 1. Phenoscape KB software architecture and data flow.

format API to an any OWL API-based reasoner; for the Phenoscape KB we use ELK, loaded with extracted class axioms from the ontologies and data. We use Owlery to support web application queries that require reasoning on arbitrary OWL class expressions. Owlet provides web services for description logic queries (obtaining subclasses and superclasses), and also supports reasoner-based query expansion using our Owlet package. The Owlery API is documented at http://docs.owlery.apiary.io/. The second web service application, phenoscape-kb-services, is the primary public Phenoscape API and provides Phenoscape application-specific services such as annotation query, semantic similarity, annotation support for presence/absence inference, and term info. Phenoscape-kb-services obtains most results via SPARQL query to the Blazegraph triple store. For use cases requiring computation by the ELK reasoner, SPARQL queries with embedded OWL expressions are first expanded using the Owlet service provided by the Owlery API before being submitted to Blazegraph. The phenoscape-kb-services API returns most results in both JSON and tab-delimited text formats. Documentation for the phenoscape-kb-services API can be found at http://docs.phenoscapekb.apiary.io/.

V. Applications

A public web user interface for the Phenoscape Knowledgebase is available at http://kb.phenoscape.org/. The web interface is a client-side browser application developed in the AngularJS JavaScript framework [16]. The application is implemented entirely upon web service calls to the phenoscape-kb-services API, ensuring that all functionality is available via a documented, public API. Using the web application, researchers can query evolutionary descriptions and gene phenotypes relevant to particular structures and qualities, search for taxonomic groups exhibiting variation similar to the phenotypic effects of a gene of interest [7] (and vice versa), and export synthetic presence/absence supermatrices using the OntoTrace system [8]. Building upon the same web service API, we have also implemented an R package, rPhenoscape, which makes some of the functionality of the KB available within the R statistical computing environment.

VI. Conclusion

The Phenoscape Knowledgebase architecture illustrates one approach to integration of multiple datasets in a rich ontological framework. Deriving the full benefit of the sophisticated knowledge represented in OBO library ontologies will often require application of automated reasoners and programmatic generation of axioms and concepts to facilitate particular use cases. Here we have provided an overview of the mix of reusable components and special purpose code we have developed to support Phenoscape; it is our hope that continued evolution of the Phenoscape KB architecture will result in further identification and development of reusable tools which can support similar efforts.

References

1. Mungall CJ, Gkoutos GV, Smith CL, Haendel MA, Lewis SE, Ashburner M. Integrating phenotype ontologies across multiple species. Genome Biol. 2010;11: R2. doi:10.1186/gb-2010-11-1-r2
2. Bradford Y, Conlin T, Dunn N, Fashena D, Frazer K, Howe DG, et al. ZFIN: enhancements and updates to the Zebrafish Model Organism Database. Nucl Acids Res. 2011;39: D822–9. doi:10.1093/nar/gkq1077
3. Eppig JT, Blake JA, Bult CJ, Kadin JA, Richardson JE, Mouse Genome Database Group. The Mouse Genome Database (MGD): facilitating mouse as a model for human biology and disease. Nucleic Acids Res. 2015;43: D726–36. doi:10.1093/nar/gku967
4. Karpinka JB, Fortriede JD, Burns KA, James-Zorn C, Ponferrada VG, Lee J, et al. Xenbase, the Xenopus model organism database; new virtualized system, data types and genomes. Nucleic Acids Res. 2015;43: D756–63. doi:10.1093/nar/gku956
5. Köhler S, Doelken SC, Mungall CJ, Bauer S, Firth HV, Bailleul-Forestier I, et al. The Human Phenotype Ontology project: linking molecular biology and disease through phenotype data. Nucleic Acids Res. 2014;42: D966–74. doi:10.1093/nar/gkt1026
6. Edmunds RC, Su B, Balhoff JP, Eames BF, Dahdul WM, Lapp H, et al. Phenoscape: Identifying Candidate Genes for Evolutionary Phenotypes. Mol Biol Evol. 2016;33: 13–24. doi:10.1093/molbev/msv223
7. Manda P, Balhoff JP, Lapp H, Mabee P, Vision TJ. Using the Phenoscape Knowledgebase to relate genetic perturbations to phenotypic evolution. Genesis. 2015;53: 561–571. doi:10.1002/dvg.22878
8. Dececchi TA, Balhoff JP, Lapp H, Mabee PM. Toward Synthesizing Our Knowledge of Morphology: Using Ontologies and Machine Reasoning to Extract Presence/Absence Evolutionary Phenotypes across Studies. Syst Biol. 2015; doi:10.1093/sysbio/syv031
9. Balhoff JP, Dahdul WM, Kothari CR, Lapp H, Lundberg JG, Mabee P, et al. Phenex: ontological annotation of phenotypic diversity. PLoS One. 2010;5: e10500. doi:10.1371/journal.pone.0010500
10. Dahdul WM, Balhoff JP, Engeman J, Grande T, Hilton EJ, Kothari C, et al. Evolutionary characters, phenotypes and ontologies: curating data from the systematic biology literature. PLoS One. 2010;5: e10708. doi:10.1371/journal.pone.0010708
11. Balhoff JP. Scowl: a Scala DSL for programming with the OWL API. The Journal of Open Source Software. 2016. doi:10.21105/joss.00023
12. Kazakov Y, Krötzsch M, Simančík F. The Incredible ELK. J Automat Reason. Springer Netherlands; 2013;53: 1–61. doi:10.1007/s10817-013-9296-3
13. Balhoff JP, Dececchi TA, Mabee PM, Lapp H. Presence-absence reasoning for evolutionary phenotypes. Proceedings of Phenotype Day of the Bio-ontologies SIG at ISMB 2014. 2014. pp. 1–4. Available: http://phenoday2014.bio-lark.org/pdf/11.pdf
14. Blazegraph. [Internet]. [cited 29 May 2016]. Available: https://www.blazegraph.com
15. spray | REST/HTTP for your Akka/Scala Actors [Internet]. [cited 29 May 2016]. Available: http://spray.io
16. AngularJS — Superheroic JavaScript MVW Framework [Internet]. [cited 29 May 2016]. Available: https://angularjs.org

Funded by National Science Foundation grants DBI-1062404, DBI-1062542.