A database structural improvement for efficient trait variation curation in Animal QTLdb and CorrDB

Z-L. Hu, C.A. Park, & J.M. Reecy*

Department of Animal Science, Iowa State University, 2255 Kildee Hall, Ames, Iowa, USA, *jreecy@iastate.edu

Abstract

In livestock genetics studies to improve production traits, the importance of accurately identifying and recording traits of interest cannot be overemphasized. However, it has been a challenge to consistently and unambiguously name and compare traits of interest that span time and geographic locations and are evaluated by different people. While the Animal Trait Ontology for Livestock, Clinical Measurement Ontology, Livestock Product Trait Ontology, and Vertebrate Trait Ontology, among others, provide good frameworks for using controlled vocabularies, building concept hierarchies, and defining relationships between terms, gaps still remain as to what level of granularity should be documented. This is important to efficiently use available resources for effective data curation. In the Animal QTLdb and CorrDB, we have introduced an approach to extend livestock trait ontologies for practical data curation that allows effective capture of trait variations while keeping complexities to a manageable level.

Introduction

The volume of genotype-to-phenotype data from livestock animal genomics studies has been increasing explosively over the past 20+ years, owing to continued progress in sequencing and genotyping technologies. For example, the cattle QTL and association data curated into the Animal QTLdb has increased 330-fold over the past 15 years (Figure 1). Trait ontology development using Vertebrate Trait Ontology (Park et al. 2013), Livestock Product Trait Ontology (Park et al., 2021), and Clinical Measurement Ontology (Shimoyama et al., 2012) has been an ongoing part of Animal QTLdb and CorrDB developments (Hu et al., 2007, 2013, 2016, 2019). The increase in incoming data presents challenges for Animal QTLdb and CorrDB data curation, not only due to an increase in workload, but also due to a need to

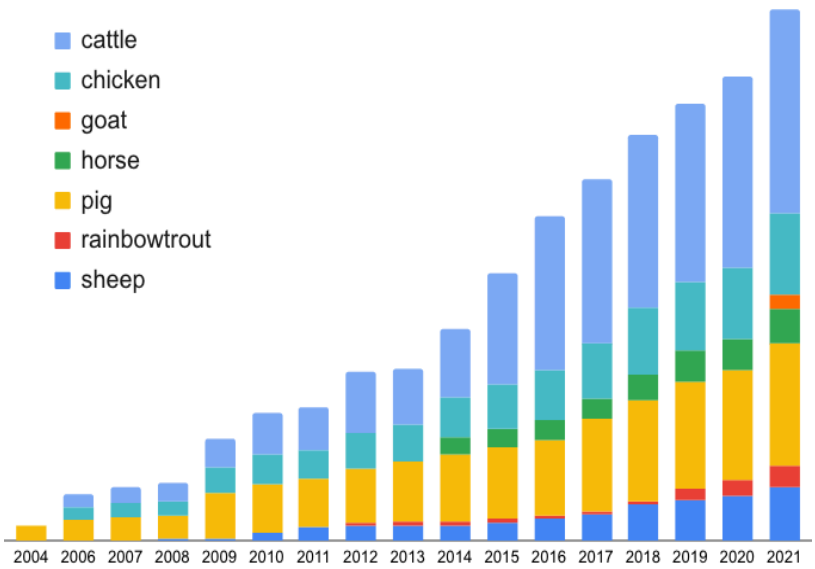

Figure 1. QTL/association data growth in the Animal QTLdb. Data counts were transformed for better visualization of scale. For the actual data counts please refer to https://www.animalgenome.org/QTLdb.

accommodate more diverse data formats, methods of data detection and analysis, and degree of trait data granularity in each new experiment. For example, for the most common trait, Average Daily Gain (ADG), many use cases have been identified in different species, such as in cattle (ADG in cows, heifers, on test, pre-weaning, post-weaning), pigs (120-210d, 120-240d, 21-120d, 21-210d, 21-240d, 21-46d, 210-240d, 30-100kg, 46-120d, 46-210d, 46-240d, BirthTo30kg, pre-weaning, post-weaning), and sheep (3-6Mos, 6-12Mos, 6-9Mos, 9-12Mos, BirthTo3Mos, BirthTo6Mos, pre-weaning, WeaningTo6Mos, WeaningTo9Mos). Clearly, numerous additional variants are possible, depending on study criteria such as age (time), weight, and/or production management stage. The combined use of these factors can vastly increase the number of different ADG terms. Previously, we developed a scheme to accommodate 'sibling traits' (Götz et al., 2012) modified from their base trait (e.g., ADG) in order for reported QTL/associations to have a proper trait name to link to (Hu et al., 2019). This conceptual design was implemented in the QTLdb curator environment and it worked fine, except that the number of such sibling traits quickly exceeded what could be easily managed. The problem was more obvious when a curator needed to choose a trait from a daunting lengthy list of (sibling) traits. When it came to recording the sibling traits as part of a trait ontology hierarchy, concerns arose, such as continually growing trait lists, possibilities of sibling traits as children of base traits, etc. (Figure 2a). To solve these problems, we have envisioned a 'trait variant' structure for practical use in the Animal QTLdb and CorrDB curator tools environment. Here we report the initial steps toward this effort.

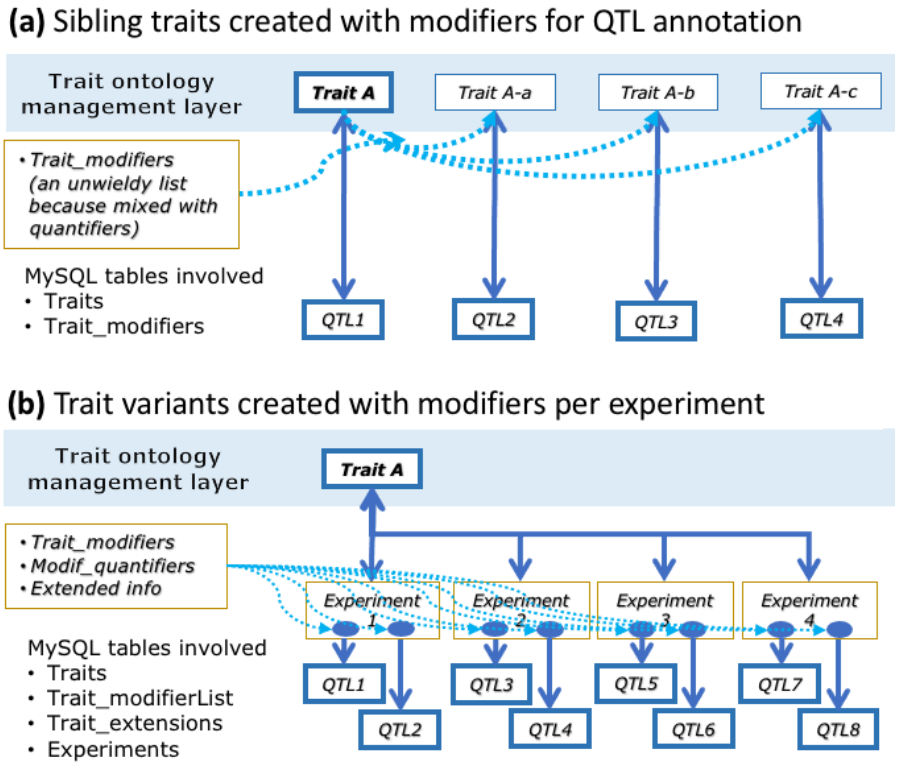

Figure 2. Conceptual data structure differences between `sibling traits` ["modified"] as part of ontology extensions (a), and `variant traits` [modified with extended info] created at the experiment level (b), and their use in QTL data annotations.

Results

Approach. A trait term may be 'modified' by another term, as a property, to produce an extension of the trait term as a new term. Examples of such extensions include muscle pH (pH) measured at 24hr or 48hr post-mortem, or in different muscles, such as biceps femoris or semimembranosus;

milk yield (MY) measured daily or cumulatively up to 305 days of calf age, or MY compared between the first and second parities, etc. In these cases, if we call pH or MY the base trait, then base traits with additional properties are modified traits. In order to bring the modifier terms to a manageable level in the trait data curation environment, we use additional terms to extend the context of a modifier. Examples of such terms can be number of days (time) in ADG measurements, parity number in MY measurements, etc. The extended names of traits incorporating the base trait in addition to the relevant modifiers are called trait variants (Figure 2b).

Implementation. By limiting modifiers to some major terms, we were able to create a relatively short list of modifiers (Analysis, Anatomy location, Environment, Herd, Instrument, Measurement, Parity, Pedigree, Stage, Time) for a pull-down menu in the curator tool (Figure 3). Each of the modifier terms has its own list of relational terms to further define its context (e.g., age, before, after, by, during, etc. for 'time'; above, anterior, at, below, dorsal, posterior, by, in, etc. for 'anatomy locations'; and so on). In the QTLdb data curation environment, multiple modifiers can be applied to a base trait. (An example with multiple modifiers is meat color lightness measured 24hr post-mortem on the semimembranosus using the CIELAB method.) This has helped to cover most, if not all, trait variants we have encountered so far. Under this management scheme, our goal is to use a set of terms over time for consistency. This implementation has improved ease of use

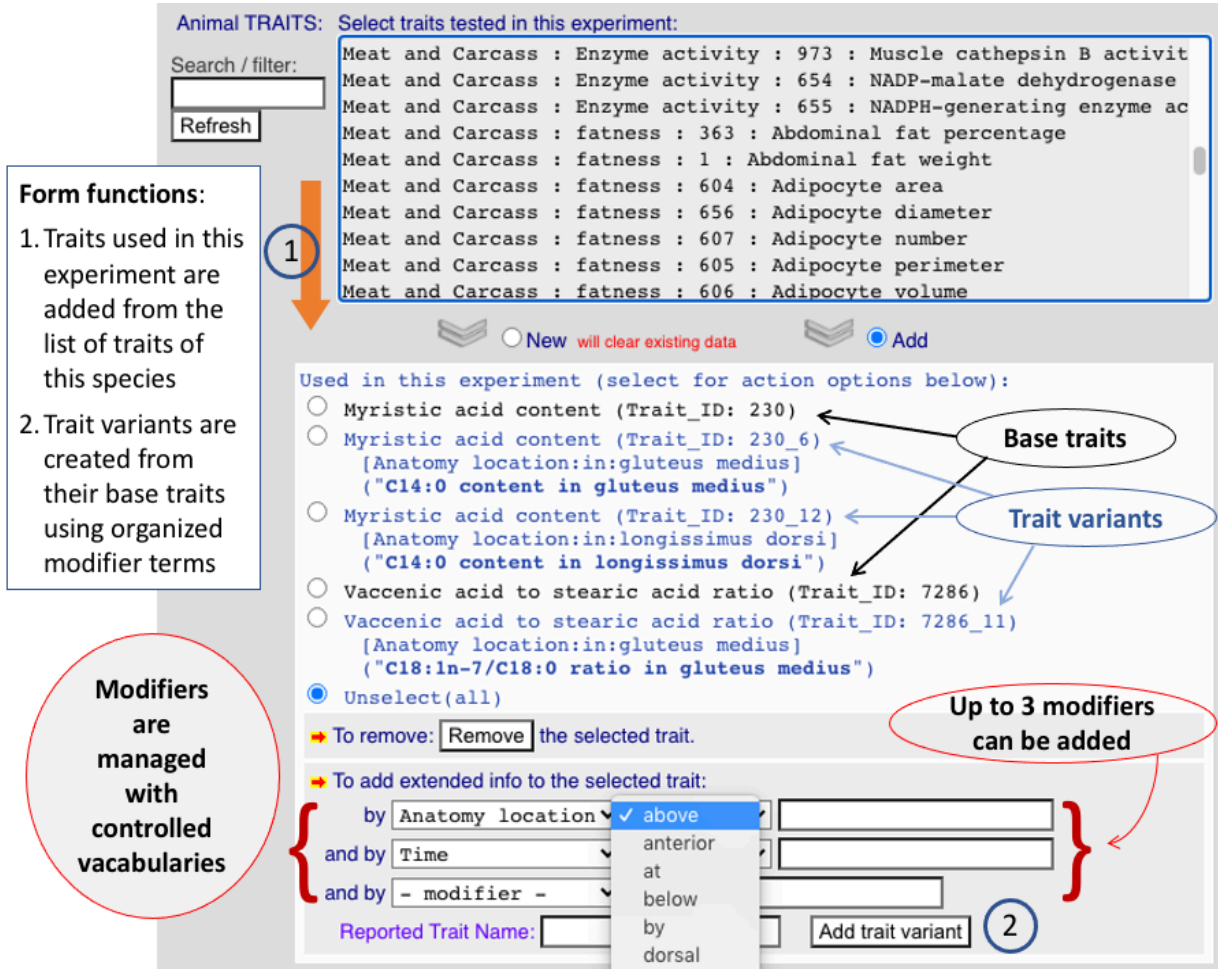

Figure 3. A web form from the implementation of the trait-variant concept showing how trait variants are created from their base traits using modifiers as controlled vocabularies.

for QTLdb/CorrDB curators by limiting the long list of sibling traits. Since we implemented the creation and management of trait variants at the experiment level (details entered once for each publication; Figure 3), it essentially moved the trait variant management out of the trait ontology term management realm (for the database as a whole; Figure 2b). The use of more MySQL tables in this implementation (Figure 2a versus Figure 2b) reflects our emphasis on the concept partitioning, compartmentalization, and relationship building, which directly result in a reduction in human effort while leaving the computer/database to handle the complexities.

Discussion

Our approach effectively helped reduce the lengthy list of 'modifiers' for practical use in the data curator environment, while increasing the trait term modifier system, which can be handled in the database backend once computer programs are developed. As such, this study provided a path for ontology developments by using proper concept partitioning approaches.

As with the development of trait ontologies, the trait variant system may also be subject to future developmental improvements. Because a trait term modifier system will be used well into the future and across species, its stability is important in order to ensure consistent and stable development down the road. For example, properly defining 'base traits' and 'modifiers' is important at the onset. For instance, 'MY' and '305-day MY' may be nearly interchangeable concepts, since 305-day MY is typically the default measurement standard in dairy production throughout the world. In situations like these, careful consideration is needed to weigh the advantages and disadvantages and in order to establish the most appropriate basic trait and its modifiers.

Because we implemented the creation of trait variants at the experiment level (Figure 3), curators must repeat the creation of a trait variant for each and every experiment in which it is used. This may necessitate future improvements regarding how trait variants created in one experiment may be reused in another experiment when traits are defined similarly in both experiments.

References

Götz F., Wächter T., and Schroeder M. (2012) Bioinformatics 28(12): i292–i300. https://doi.org/10.1093/bioinformatics/bts215.

Hu Z-L, Fritz E.R and Reecy J.M. (2007) Nucleic Acids Res 35(Database issue):D604-D609. https://doi.org/10.1093/nar/gkl946.

Hu Z-L, Park C.A., and Reecy J.M. (2016) Nucleic Acids Res 44(D1): D827-D833. https://doi.org/10.1093/nar/gkv1233.

Hu Z-L, Park C.A., and Reecy J.M. (2019) Nucleic Acids Res 47(D1):D701–D710. https://doi.org/10.1093/nar/gky1084.

Hu Z-L, Park C.A., Wu X.L. and Reecy J.M (2013) Nucleic Acids Res 41(D1): D871-D879. https://doi.org/10.1093/nar/gks1150.

Park C.A., Bello S.M., Smith C.L., Hu Z-L., Munzenmaier D.H., et al. (2013) J Biomed Semantics 4(1):13. https://doi.org/10.1186/2041-1480-4-13.

Park C.A., Hu Z-L, and Reecy J.M. (2021) Livestock Product Trait (LPT) Ontology. Available at: https://www.animalgenome.org/bioinfo/projects/lpt/. Last updated: December 16, 2021.

Shimoyama M., Nigam R., McIntosh L.S., Nagarajan R., Rice T., et al. (2012) Front Genet 3:87. https://doi.org/10.3389/fgene.2012.00087.

Hu Z-L, Park C.A., and Reecy J.M. (2019) Nucleic Acids Res. 47(D1):D701–D710. https://doi.org/10.1093/nar/gky1084.