Ontology Development Strategies and the Infectious Disease Ontology Ecosystem

Giacomo De Colle [1], Ali Hasanzadeh[1] and John C. Beverley [1,2]

[1]University at Buffalo, Buffalo, NY, USA
[2]National Center for Ontological Research, Buffalo, NY, USA

Abstract
After motivating a framework for evaluating top-down, middle-out, middle-in, and bottom-up ontology development strategies, we apply our framework to investigate whether infectious disease ontologies - specifically, the Virus Infectious Disease Ontology (VIDO) and the Coronavirus Infectious Disease Ontology (CIDO) - effectively promote semantic interoperability.

Keywords
Top-down, middle-out, bottom-up, middle-in, infectious disease ontologies, CIDO, VIDO

1. Introduction

Ontologies are developed using numerous strategies. Some follow a *top-down* strategy, in which classes or categories are devised to constrain lower-level ontology content extending from them [1]. *Bottom-up* strategies tend to begin creating ontology content reflecting a given domain of interest, representing that content with a high degree of fidelity [2]. Data used as a basis for a bottom-up ontology comes in many flavors, e.g. a SQL database, an Excel file, previously existing taxonomies, etc. An ontologist following the bottom-up strategy will attempt to uncover terms and relations implicit in the data, to represent them ontologically. The *middle-out* strategy aims at attaining the benefits of the preceding strategies, like proximity to the domain and ensuring consistency across lower-level ontologies [3,4]. Middle-out ontologies are developed at some level of abstraction above one or more domains to be modeled, but not as far removed as those that begin with the *top-down* strategy. Notice that these strategies are distinguished based on the starting point of their development [5]. They are thus distinct from ontology architectures distinguished in terms of their coverage, e.g. top-level ontologies such as the Basic Formal Ontology (BFO) [1,6], DOLCE [7], and YAMATO [8]; *mid-level* ontologies – such as the Common Core Ontologies (CCO) suite [9] or the Industrial Ontology Core [10].

While each of these strategies has been discussed in the literature, there has, as of yet, not been a rigorous, fair, comparison provided between them. One of our aims in this article is to provide such a comparison. Another of our aims is to apply the results of our comparison to representative ontologies which follow one of these strategies, specifically, the Coronavirus Infectious Disease Ontology (CIDO) [11,12] and the Virus Infectious Disease Ontology (VIDO) [13], designed according to the top-down strategy. Our evaluation will demonstrate the extent to which these ontologies effectively promote semantic interoperability, a primary goal of ontology development.

2. Criteria for Evaluating Ontologies

[1]Proceedings of the International Conference on Biomedical Ontologies 2023, August 28th-September 1st, 2023, Brasilia, Brazil
EMAIL: gdecolle@buffalo.edu (A. 1); ahasanza@buffalo.edu (A. 2); johnbeve@buffalo.edu (A. 3)
ORCID: 0000-0002-3600-6506 (A. 1); 0009-0003-6686-3319 (A. 2); 0000-0002-1118-1738 (A. 3)

We take as a starting point for the identification of evaluative criteria for ontologies, the work of Denny Vrandečić [14], itself originated from the work of Thomas Gruber. Vrandečić's evaluating criteria – displayed in Table 1 – have been used by numerous developers to improve the quality of their ontologies and knowledge graphs [15-7]. Note, no single ontology can perform well on all these metrics; indeed, some criteria appear in conflict, such as *completeness* and *conciseness*.

Table 1
Vrandečić's Ontology Evaluation Criteria

Criterion	Description
Accuracy	The extent to which an ontology accurately represents the domain within its stated scope.
Adaptability	The extent to which an ontology can be extended to represent entities in domains outside of its originally stated scope.
Clarity	The extent to which an ontology unambiguously, clearly, conveys the meanings of its terms and relations to users.
Completeness	The extent to which an ontology includes terms and relations that cover the entire domain within the intended scope of the ontology.
Conciseness	The extent to which an ontology is parsimonious, does not include redundant content, or irrelevant axioms.
Coherence	The extent to which an ontology is both logically consistent and semantically aligned with the intention of its creators.
Organizational fitness	The extent to which, within an organizational context, an ontology is integrated within the organization.

Accuracy follows naturally from the goals of most ontology development, namely, representing a given domain using a machine-readable controlled language. Accuracy may be determined when ontology developers and subject-matter experts interact during ontology development, in the interest of reaching consensus over ontology labels, definitions, logical relationships, etc. [18]. The most successful ontologies are reused, extended to new domains, and integrated with other ontologies and knowledge representation projects. An *adaptable* ontology is one which can be easily extended into a new domain distinct from that for which it was initially designed. *Clarity* can be achieved by proper definition development and documentation. In the absence of definitions, other annotation properties such as comments, citations, notes, labels, alternative labels, and preferred labels may be used to promote clarity. A *complete* ontology will include terms and relations needed to represent any terms within its scope. Completeness is related to accuracy, as evidenced by the impact completeness may have on whether competency questions for an ontology can be answered [19]. Adequately answering competency questions requires that the ontology has adequate coverage of the domain. A *concise* ontology will not include unnecessary elements or axioms, which promotes understanding by users and helps avoid the confusion that might emerge from the presence of, say, many unneeded classes. Related, a concise ontology will include only those terms and relations that are needed to represent the domain within its scope, i.e., the minimal set of terms and relations. A *coherent* ontology will be logically consistent and will entail as little beyond the intent of its creators as possible. Put another way, the intended interpretation of the ontology will match as closely as possible the semantic interpretation generated by model checkers [20] or OWL reasoners [21,22]. *Organizational Fitness* is not directly about a given ontology *per se* but also involves the organizational context in which the ontology is deployed. An ontology that scores high on organizational fitness is successfully and consistently deployed in an organizational context, is being maintained and developed, is accessible to members of the organization, and can be aligned with other ontologies within the organization.

The standard we use to adjudicate the importance of each of these criteria is, arguably, the primary aim towards which any good ontology aims, the promotion of *semantic interoperability* – the ability of computer systems to exchange data with unambiguous, shared meaning [1,23-4]. The most important criteria in this perspective seem to be accuracy, coherence, and adaptability. Notice that *coherence* cannot be simply reduced to the absence of contradictions in the ontology but is rather the absence of statements that are semantically, and often implicitly, in conflict. During the development of VIDO, such a coherence issue was identified with respect to the definition of "organism" used by the Ontology for Biomedical Investigations (OBI), which included viruses within purview [25]. "Organism", however, was defined as comprised of cellular entities, despite viruses being acellular. This is an example of a semantically incoherent statement that was originally not noticed and that might be more easily avoided by adopting one ontology development strategy over the others.

3. Evaluating Ontology Development Strategies

The three strategies in our focus are evaluated on a spectrum. The evaluation of these strategies was carried out by analyzing ontologies that explicitly adopt them, and then testing them against a set of questions devised to check compliance with the our criteria. In the future, we plan to test these questions and criteria on a larger scale using empirical methods, i.e. surveys and feedback group sessions with ontology developers, etc. In what follows, we do not conclude that ontologies borne from the one strategy necessarily perform better than others with respect to our criteria. Instead, we suggest that they stand a better chance of doing so than ontologies developed following one of the other strategies.

Organizational Fitness

- **Bottom-up:** This strategy promotes work on the same data set. However, different segments of this dataset, when used by various team members, might lack semantic uniformity. This disparity becomes apparent when generalizing the embedded knowledge.
- **Middle-out:** This strategy provides a common starting point for building classes that members of an ontology development team can use. Nevertheless, there may be a lack of shared understanding across ontologies borne out of this strategy within an organization, in particular with respect to higher-level terms and relations such as **part of**, **quality**, or **process**.
- **Top-down:** While starting from the most general classes and working your way down is a suitable way to promote consistency, it is not free of challenges. Deciding how best to define such classes across an enterprise requires time and skill that organizations may not have.

Definitions (Accuracy, Clarity, Coherence)

The Aristotelian scheme for definition writing is standardly employed by ontology developers [1,26], i.e., to define class A you identify its parent class B and describe the differentia that distinguishes instances of A from any other instances of B [27]. Adopting the Aristotelian model forces the ontologist to identify the relation of the class with the other classes in the hierarchy when building the definition, and provides a format that is human-readable, consistent, and understandable for the user if correctly applied.

- **Bottom-Up:** One cannot adopt the model of writing Aristotelian definitions if following this strategy, since there is no top-level ontology from which to identify parent classes. Consequently, those following the bottom-up strategy must develop some manner of schema to do so. Whatever that schema amounts to, it will not be extending from a top-level ontology, which seems a cost.
- **Middle-Out:** The middle-out strategy may adopt the Aristotelian schema for definitions to some extent, i.e., for classes on the lower level. Nevertheless, it is not possible to adopt this strategy for classes on the upper level of the hierarchy.
- **Top-Down:** Insofar as the top-down strategy allows for extensive application of Aristotelean definition schema, it scores highly with respect to definitions. By using the Aristotelian method, ontology developers can offer definitions by relating them to classes in a top-level ontology.

Axiomatization (Accuracy, Clarity, Consistency)

Formal axioms associated with terms and relations in an ontology promote accuracy, clarity, and consistency, as well as automated checking of such evaluative criteria. Axioms provide machineinterpretable enforcement of domain and range for relations, ensure disjoint classes do share instances in common, and in general connect parts of an ontology hierarchy to other parts of that hierarchy.

- **Bottom-Up:** Axiom development may be challenging when following the bottom-up strategy. Suppose a class **start time** should be related to a class **earlier than** that itself relates to any class representing entities having a duration, such as **phosphorylation** or **dimerization**. Intuitively, one might say such classes are all activities or processes of some sort. This is not an obvious option when following the bottom-up strategy.
- **Middle-Out:** The middle-out strategy fares better, since axioms can be written starting from a broader scoped architecture than that provided by the bottom-up strategy. This allows those following a middle-out strategy to leverage the space of possibilities already constrained by the placement of lower-level terms within the ontology taxonomy. Put another way, following the middle-out strategy might avoid some of the issues raised above by including natively a class-like **activity**, but will at some point lack a way to answer questions further up the taxonomy.
- **Top-Down:** The top-down strategy appears to fare better than either of the preceding strategies, with respect to axiom development. By populating ontology entities downward from existing classes and relations, ontology elements have an implicit formal structure inherited from the top-down strategy itself. Moreover, because there are an increasing number of logical constraints enforced as one proceeds down the taxonomy, the scope of possible axioms that can be applied to an ontology element is decreased.

Reinvention (Accuracy, Organizational Fitness, Clarity, Consistency)

It is important to determine the extent to which an ontology strategy encourages or discourages duplication of effort. Duplication is not only dangerous because it wastes time and resources. It also risks re-creating the mistakes that other ontologists have effectively amended in their efforts.

- **Bottom-Up:** The bottom-up strategy appears to encourage duplicative effort when viewed from the perspective of interoperability. Focusing solely on accurately modeling a domain runs the risk of missing the forest for the trees. Put another way, creating terms and relations highly specific to a domain, without reflection on how they might relate to existing ontologies, impedes interoperability with those other ontologies. This is, indeed, a recipe for creating data silos [1].
- **Middle-Out:** Middle-out ontologies avoid some of the issues plaguing the bottom-up strategy with respect to reinvention, by facilitating an upwards population of ontology content, as needed. Nevertheless, this strategy will ultimately run into the same issues at a higher level of generality.
- **Top-Down:** Top-down strategies clearly shine with respect to reinvention. This strategy creates a common architecture from which terms and relations extend, and consequently, can be reused in other ontologies employing the same top-level architecture. This is, of course, not to say that all ontologies designed according to this strategy thereby perform well with respect to reinvention. Several ontologies, for example, in the Open Biological and Biomedical Ontology (OBO) Foundry [28] extend from the top-level BFO but in the absence of collaboration across nearby efforts, do so by producing duplicative content.

Unused Content (Conciseness, Clarity)

In some cases, ontologies will be developed with placeholder classes or relations, intended to be connected to data, but which never are. The result is there may be ontology classes that are not used, potential points of confusion, or perplexity for ontology developers unfamiliar with the intent behind creating such content.

- **Bottom-Up:** Bottom-up strategies perform best with respect to this criterion, as ontology content is developed on demand, directly from relevant domain data. Bottom-up ontology development encourages creating ontology content that is representative of the domain, and so makes it unlikely that ontology developers following this strategy will create empty content.
- **Middle-Out:** Middle-out strategy followers arguably perform as well as bottom-up strategy followers with respect to this criterion. By keeping an eye on the domain-level and the upper-level, this strategy tends to result in ontologies that populate classes which are integrated into analyses of the domain in question.
- **Top-Down:** Top-level strategy results perform worst with respect to this criterion. Constructing ontologies from the most general content down runs the risk of including plausible content that does not end up being used by domain ontologies extending from the top-level.

Abstract Representations (Clarity, Conciseness, Accuracy)

Ontologies sometimes include content that is so general that may lead to confusion by subject-matter experts, ontology developers, or other users. Such abstract representations may be challenging for users to understand, which may, in turn, undermine the use of the ontology, or support perceptions that a given ontology is too difficult to be used for practical purposes.

- **Bottom-Up:** More than any of the alternative strategies, bottom-up strategies typically avoid the inclusion of abstract representations within ontology output. Working so closely at the domain level encourages the creation of ontology content understood by relevant subject-matter experts.
- **Middle-Out:** Similarly, the middle-out strategy promises to avoid the inclusion of many abstract representations, given its emphasis on domain-level content while attending to top-level architecture. Expanding upwards only when needed keeps abstract representations at a minimum.
- **Top-Down:** Abstract representations are often found in ontologies designed following a top-down strategy, as top-level content must often be rather general. Terms such as **continuant** or **predicate** are often divorced from the experiences of subject-matter experts, employment of them leads to confusion at best and misuse at worst. Consequently, it becomes challenging to link these abstract representations to domain-level content, which is indeed one of the main purposes of developing ontologies.

Adaptability (Adaptability, Organizational Fitness)

Ontologies should be extendable to new domains, reflecting new discoveries, scientific advancements, or novel ways of understanding existing knowledge. We must take care here, however, when reflecting on what it means for an ontology development strategy to promote adaptability. Ontologies may be extended vertically - by upward or downward population of content – or horizontally – by covering new domains not yet represented. The upward and downward population of content places constraints on the ontology content that can be created consistently. For example, asserting that instances of **process** must have some **temporal part** requires that any subclasses of **process** also have some **temporal part**, thereby narrowing the space of possible extensions. A horizontal extension is, however, sometimes entirely unconstrained, as when an ontology is developed outside of existing ontologies.

- **Bottom-Up:** Strictly speaking, this strategy promotes adaptability solely in the sense of horizontal extension since it encourages the creation of ontology silos, entirely disconnected from other potential ontologies within an enterprise. New terms and relations can be introduced outside the scope of existing ontologies developed following this strategy, without encountering inconsistency. This, of course, comes at the cost of interoperability.
- **Middle-Out:** This strategy fares poorly when adaptability is understood as a vertical extension since constraints are applied during the upward and downward population of ontology content. Ontology content developed in the upward direction must also bear constraints from the lower-level ontology content, and vice versa. The upward and downward aspects of this strategy lead it thus compromised on two fronts. Consider, from the Credential Transparency Description Language [29]: an **address** is defined as "particulars describing the location of the place", whereas 'particular' refers to details of a description. Such use of 'particular' is, however, wildly different from the use of 'particular' in, say, DOLCE [5], where the term denotes an individual. Extending Credential Engine to a domain covered by a DOLCE-based ontology would thereby generate inconsistency. On the other hand, where adaptability is understood as horizontal extension the middle-out strategy permits the development of ontologies with few constraints, with the caveat simply being that at some point such new ontologies will need to connect with a top-level. In this respect, the middle-out strategy provides flexibility in ontology design.
- **Top-Down:** The top-down strategy encourages constraints on ontology content extended from the top-level, and so constrains the range of possible ontology extensions. Put another way, when adaptability is understood as a vertical extension, the top-down strategy places constraints on a downward population of ontology content and does not permit an upward population. In this respect, the top-down strategy fares worse than the bottom-up strategy with respect to permitting extensions without generating inconsistency, since any downward populated content

must remain consistent with the content from which it extends. The top-down strategy fares worse, moreover, than either of the other strategies when considering adaptability from the horizontal perspective. This is because any ontology designed following the top-down strategy must find parent terms and relations that ultimately have roots in a top-level ontology. In other words, when attempting to cover a new domain, ontology developers must *find* constraints to apply to ontology content for this new domain.

Implicitness (Accuracy, Consistency, Coherence)

It is important to be able to discern the extent to which an ontology design strategy encourages or discourages the creation of implicit incoherence.

- **Bottom-Up**: While the bottom-up strategy offers developers greater flexibility, this approach has its challenges when it comes to inviting implicit incoherence. The significant dependence on data often results in classes that mirror imperfections in the data. Consequently, terms and relations falling out of a bottom-up strategy tend to reflect inherent flaws in the data.
- **Middle-Out**: Without an explicit structure guiding expansion, ontology developers following this strategy may overlook the "implicit rules" essential for coherence. Additionally, the absence of a shared foundational understanding can render much of the stored knowledge inconsistent, without obvious methods for detecting such inconsistency. Moreover, transitioning the ontology to different domains can be challenging for middle-out strategy ontologies, especially when the abstract classes it has developed are too domain-specific. As discussed, the term **address** in the Credential Transparency Description Language [29] is defined differently than in other ontologies, which can lead to confusion when seeking alignment.
- **Top-Down**: This strategy exhibits advantages and challenges. The top-down strategy equips the ontologist with the tools needed to craft robust and rigorous axioms, paired with clear class definitions, reducing the risk of implicit inconsistencies. However, the rigidity of the top-level classes can be a limitation. If incoherence arises, rectifying it can be challenging, especially if it necessitates changes to foundational classes.

4. Discussion

This section offers a comprehensive analysis of three ontological strategies, summarizing their strengths, weaknesses, and unique characteristics with respect to the preceding evaluative criteria. We aim to provide a holistic view, facilitating a deeper understanding of each method's applicability and limitations.

The bottom-up strategy excels in clarity and conciseness, primarily because the ontologies it produces are more closely aligned with the terminology associated with a given domain. When conducted effectively, this strategy accurately mirrors domain data structure, ensuring a direct correlation with the original data's knowledge. However, such close reliance on original datasets can compromise adaptability and accuracy when applied to slightly varied domains. The bottom-up strategy's biggest strength - its proximity to the original data – is also its biggest weakness.

The middle-out strategy generally outperforms the bottom-up method. Ontology developers following this strategy tend to avoid getting mired in data details. Nevertheless, the absence of a top-level set of common terms can lead to implicit commitments in an ontology developed following the middle-out strategy. Moreover, developers will ultimately be pressed in this strategy to create local, domain-dependent top-level terminology, that will quite likely impede clarity and adaptability, and lead to challenges in semantic homogeneity and external interoperability.

The top-down strategy stands out in terms of coherence, clarity, and completeness. It offers adaptability, especially when using top-level classes to ensure coherence across domains. However, its inherent ontological commitments may make it fragile, necessitating careful adaptation to new domains, and rigorous attention to how ontologies are extended from it. While the approach promotes ontology alignment and avoids redundancy, it demands a centralized organizational structure and skilled individuals adept at handling abstract classes.

The top-down strategy appears to perform best overall with respect to our evaluative criteria, with the middle-out strategy coming second, and the bottom-up strategy third. While challenges exist, the top-down method's proactive approach to axiomatization and class building during production can also mitigate issues that might arise post-deployment. That said, it is important to emphasize that applying top-down techniques often requires more effort in the development of an ontology. Compliance to a top-level is not cheap, and the benefits gained are not always obvious when an ontology is being developed for a specific domain-level modeling task. The tradeoff is – we claim – best characterized as a difference between short-term and long-term costs. Effort spent upfront following a top-down strategy saves effort downstream, a cost that artifacts resulting from middle-out and bottom-up strategies will ultimately have to pay later – when it is more expensive - if they hope to promote semantic interoperability.

5. The Middle-In Strategy

Most researchers appear to develop ontologies according to one of the three developmental strategies discussed thus far, but our analysis of the complexity of ontology development suggests there is a promising strategy not yet discussed in the literature. The *middle-in* strategy combines aspects of the top-down and bottom-up strategies, in a manner distinct from the middle-out strategy. According to this strategy, ontologists begin with a top-level ontology which is then used as a guide when exploring data to develop domain-level ontology content. This strategy takes its name from starting at both the top and bottom levels, then developing ontology content to meet in the middle.

During the writing of this manuscript, it was discovered that some ontology developers have – often unknowingly – employed such a strategy or have hinted towards such a strategy. For example, in 2010 Enrico Francesconi and his team published a paper where they describe a similar method to build DALOS, a multilingual ontology for the legal domain [30]. More recently, CIDO and VIDO are the result of the middle-in strategy, as they each use BFO as a top-level ontology, while employing bottom-up, data-focused, design. For example, they re-use reference ontologies and mid-level ontologies like CHEBI [31] and OBI [25]. Moreover, CIDO was built starting from the classifications of real-life, already existing data coming from GISAID, NextStrain, and DrugBank, as well as data coming from domain-specific literature [32], and VIDO was related to taxonomies such as the NCBITaxon [33] and based on the Baltimore Classification [34]. Finally, both made use of connections with domain experts to maintain their terminology and ontological commitments grounded within the domain. The perks of employing top-down methods are still visible in the quality of employed definitions and axioms, as well as in the number of external ontologies and domains that are referred to, suggesting the ontologies are adaptable and extensible.

CIDO and VIDO score well on all criteria most relevant for semantic interoperability. Starting with accuracy, the precision of the content of the two stems from the quality and variety of the data and domain knowledge used as a development basis. Clarity is favored by the terminology respecting these sources, as well by the terms being defined using the Aristotelian schema. Coherence is respected through the use of axioms taken or developed starting from the top-level layer provided by BFO. Adaptability is preserved thanks to native integration with a set of BFO-based ontologies and to potential coherent integration with all other ontologies that use the same upper-level architecture. These virtues make CIDO and VIDO stable hubs for long-term development of interoperable terminologies in the realm of infectious disease representation, and provide confirmation for our evaluation of middle-in ontologies as best suited to promote semantic interoperability.

6. Conclusion

Each of the four methodologies presents distinct advantages and challenges. The bottom-up approach, with its implicit semantics, is apt for smaller projects where team members share a common understanding and where rapid access to existing well-defined datasets is needed. Conversely, for larger projects demanding explicit semantics to maintain coherence across multiple contributors, extract implicit information, or promote reasoning capabilities, the middle-out and top-down approaches are more appropriate. Both, however, are less appropriate than the middle-in strategy which concluded our discussion. In the context of infectious disease ontologies, CIDO and VIDO represent successful endeavors in providing a basis for structuring data. We argue that the quality of the two ontologies is in part a function of their adoption of a middle-in development strategy. While the middle-in strategy is akin to the top-down strategy, the former may yield more fragile ontologies due to its rigorous nature. Proper implementation demands meticulous effort from ontologists in crafting precise mappings and alignments. As we understand it the primary impediment to interoperability arises from inconsistent terminology and axioms across domains; this is an issue best addressed by the middle-in strategy.

Acknowledgements

The authors wish to acknowledge the help and comments of Alexander Diehl, Asiyah Lin, the members of the Spring 2023 UB Logic for Ontology seminar, the PROVO-BFO mapping group, the 2023 ICBO Infectious Disease workshop audience and especially of one of its participants, who pointed out that empirical tests should accompany our current methodology for testing adherence to the criteria.

References

[1] Arp Robert, Spear Andrew D., Smith Barry, “Building Ontologies with Basic Formal Ontology”, MIT Press, 2015.
[2] Keet, Maria C. “Introduction to Ontology Engineering”, College Publications, 2020.
[3] Belhoucine, K., Mourchid, M., Mouloudi, A., Mbarki, S., "A Middle-out Approach for Building a Legal domain ontology in Arabic," 2020 6th IEEE Congress on Information Science and Technology (CiSt), Agadir - Essaouira, Morocco, (2020): 290-295.
[4] Uschold Mike, Grüninger Michael. “Ontologies: Principles, Methods and Applications.”, The Knowledge Engineering Review, vol. 11, no. 2, pp. 93–136 (1996)
[5] Raad Joe, Cruz Christophe, “A survey on ontology evaluation methods”, Proceedings of the International Conference on Knowledge Engineering and Ontology Development, part of the 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, (2018).
[6] BFO 2020, 2020, URL: https://github.com/BFO-ontology/BFO-2020
[7] Borgo Stefano, Claudio Masolo. “Ontological Foundations of Dolce”, Theory and Applications of Ontology: Computer Applications, pp. 279–295, (2022).
[8] Mizoguchi Riichiro, Borgo Stefano, “Yamato: Yet-Another More Advanced Top-Level Ontology”, Applied Ontology, vol. 17, no. 1, pp. 211–232, (2022).
[9] CCO, Common Core Ontologies 2023, URL : https://github.com/CommonCoreOntology/
[10] Kulvatunyou, B. , Drobnjakovic, M. , Ameri, F. , Will, C. and Smith, B. (2022), “The Industrial Ontologies Foundry (IOF) Core Ontology”, Formal Ontologies Meet Industry (FOMI) 2022, Tarbes, FR, (2022).
[11] Yongqun He, Hong Yu, Edison Ong, Yang Wang, Yingtong Liu, anthony Huffman, Hsin-hui Huang, John Beverley, Junguk Hur, Xiaolin Yang, Luonan Chen, Gilbert S. Omenn, Brian Athey and Smith Barry, “CIDO, a community-based ontology for coronavirus disease knowledge and data integration, sharing, and analysis”, Scientific Data Nature, (2020).
[12] He Yongqun, Yu Hong, Hufman Anthony, Yu Lin Asiyah, Natale Darren A., Beverley John, Zheng Ling, Perl Yehoshua, Wang Zhigang, Liu Yingtong, Ong Edison, Wang Yang, Huang Philip, Tran Long, Du Jinyang, Shah Zalan, Shah Easheta, Desai Roshan, Huang Hsin-hui, Tian Yujia, Merrell Eric, Duncan William D., Arabandi Sivaram, Schriml Lynn M., Zheng Jie, Masci Anna Maria, Wang Liwei, Liu Hongfang, Smaili Fatima Zohra, Hoehndorf Robert, Pendlington Zoë May, Roncaglia Paola, Ye Xianwei, Xie Jiangan, Tang Yi-Wei, Yang Xiaolin, Peng Suyuan, Zhang Luxia, Chen Luonan, Hur Junguk, Omenn Gilbert S., Athey Brian and Smith Barry, “A comprehensive update on CIDO: the community-based coronavirus infectious disease ontology”, Journal of Biomedical Semantics, (2022).
[13] Beverley, John ; Babcock, Shane ; Smith, Barry ; He, Yongqun ; Merrell, Eric ; Cowell, Lindsay ; Hurley, Regina & Duesing, Sebastian (2022). Coordinating Coronavirus Research: The COVID-19 Infectious Disease Ontology. Proceedings of the International Conference on Biomedical Ontologies.
[14] Vrandečić Denny, “Ontology Evaluation.” in: Steffen Staab, Rudi Studer (Ed.), Handbook on Ontologies, 200, pp. 293–313.
[15] Diehl Alexander D., Lee Jamie A., Scheuermann Richard H., Blake Judith A., “Ontology development for biological systems: immunology”, Bioinformatics Applications Note, Vol.23 no.7 (2007): 913-915.
[16] Hogan, Aidan, et al. "An empirical survey of linked data conformance.", Journal of Web Semantics 14, pp. 14-44, (2012).

[17] Zaveri Amrapali, Rula Anisa, Maurino Andrea, Pietrobon Ricardo, Lehmann Jens, Auer Sören, "Quality assessment for linked data: A survey", Semantic Web 7.1 pp. 63-93, (2016).
[18] Neuhaus Fabian, Hastings Janna, “Ontology is Consensus Creation, Not (Merely) Representation, Applied Ontology 17, (2022).
[19] Fox Mark, Grüninger Michael, “The Role of Competency Questions in Enterprise Engineering”, Proceedings of the IFIP WG5.7 Workshop on Benchmarking - Theory and Practice, (1994).
[20] McCune William, “Mace4 Reference Manual and Guide”, Argonne National Laboratory Technical Memorandum n.264, (2003).
[21] Hermit, an OWL 2 reasoner, 2023. URL: http://www.hermit-reasoner.com/
[22] Pellet, an OWL 2 reasoner, 2023. URL: https://github.com/stardog-union/pellet
[23] Abukwaik H, Taibi D, Rombach D (2014) Interoperability-related architectural problems and solutions in information systems: a scoping study. In: Avgeriou P, Zdun U (eds) Software architecture. ECSA 2014. Lecture notes in computer science, vol 8627. Springer, Cham. https://doi.org/10.1007/978-3-319-09970-5_27
[24] Justine Flore Tchouanguem, Mohamed Hedi Karray, Bernard Kamsu Foguem, Camille Magniont, F. Henry Abanda, and Barry Smith. 2021. BFO-based ontology enhancement to promote interoperability in BIM. Appl. Ontol. 16, 4 (2021), 453–479. https://doi.org/10.3233/AO-210254
[25] Bandrowski A, et al. (2016) *The Ontology for Biomedical Investigations*. PLoS ONE 11 (4): e0154556. doi: 10.1371/journal.pone.0154556.
[26] Otte NJ, Beverley J, Ruttenberg A. (2022) *BFO: Basic Formal Ontology*. Applied Ontology. 17-43.
[27] Seppälä Selja, Ruttenberg Alan, Schreiber Yonatan, Smith Barry, “Definitions in ontologies”, Cahiers de Lexicologie 109 (2), pp.175-207, (2016).
[28] Smith B, et al. (2007) *The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration.* Nat Biotechnol. 25:1251–1255. doi:10.1038/nbt1346.
[29] Credential Transparency Description Language (CTDL), 2023, Accessed 8-14-23 from https://credentialengine.org/credential-transparency/ctdl/
[30] Francesconi Enrico, Montemagni Simonetta, Peters Wim, Tiscornia Daniela, “Integrating a Bottom–UP and Top–down Methodology for Building Semantic Resources for the Multilingual Legal Domain”, Semantic Processing of Legal Texts, pp. 95–121, (2010).
[31] Hastings J. et al. *ChEBI in 2016: Improved services and an expanding collection of metabolites*. Nucleic acids research 44, D1214-1219, doi:10.1093/nar/gkv1031 (2016).
[32] Yu Hong, Li Li, Huffman Anthony, Beverley John, Hur Junguk, Merrell Eric, Huang Hsin-hui, Wang Yang, Liu Yingtong, Ong Edison, Cheng Liang, Zeng Tao, Zhang Jingsong, Li Pengpai, Liu Zhiping, Wang Zhigang, Zhang Xiangyan, Ye Xianwei, Handelman Samuel K., Sexton Jonathan, Eaton Kathryn, Higgins Gerry, Omenn Gilbert S., Athey Brian, Smith Barry, Chen Luonan and He Yongqun, “A new framework for host- pathogen interaction research”, Frontiers in Immunology, (2022)
[33] Federhen S. (2012) *The NCBI Taxonomy Database*. Nucleic Acids Res. 40:D136-D143. doi:10.1093/nar/gkr1178.
[34] Baltimore D. (1971). *Expression of Animal Virus Genomes*. Bacteriological Reviews. 35, 235-41.