Mapping the W3C Provenance Ontology (PROV-O) to the Basic Formal Ontology (BFO): Epistemological Considerations and Preliminary Implementation

First Author[1][0000-1111-2222-3333], Second Author[2] and Third Author[1]

[1] xxxx University, xxx xx xxxxx, USA
[2] xxx, xxx. xxx, xxx
a@a.com

Abstract. Large-scale, generative Artificial Intelligence (AI) constructs have proliferated rapidly since the introduction of the Transformer architecture in 2017, and the Generative Pre-trained Transformer (GPT) from OpenAI throughout the early 2020s. Now, more than ever, there is a dire need for machine-readable provenance in response to the deluge of AI data, as large AI constructs are black-box systems lacking transparency. The World Wide Web Consortium (W3C) spearheaded the Semantic Web movement in the late 1990s and early 2000s, which intended to make all data interconnected and interoperable. In association with the W3C, the Provenance Ontology (PROV-O) allows the representation of machine-readable artifact provenance. From the efforts of the Open Biomedical Ontologies Foundry, the Basic Formal Ontology (BFO) is a high-level, universally applicable ontology capable of representing any entity in reality within its Aristotelian taxonomy. As of 2021, BFO possesses the unique distinction of being the first ISO/IEC standardized top-level ontology. In contrast, PROV-O is over a decade old and, generally, relegated to small research projects, despite its foundational status in the W3C. A mapping of the seminal PROV-O to the ISO/IEC standardized BFO is beneficial, insofar as interest in provenance for software systems can be brought into the new era of large-scale generative AI with an internationally recognized representation format. This is a contribution towards the principles of FAIR data for large-scale generative AI: Findable, Accessible, Interoperable and Reusable.

Keywords: PROV-O, BFO, ontology mapping, FAIR data.

1 Introduction

The World Wide Web Consortium (W3C) defines provenance as the "… information about entities, activities, and people involved in producing a piece of data or thing, which can be used to form assessments about its quality, reliability or trustworthiness" [1]. Provenance is essential for digital systems, insofar as its proper maintenance facilitates data transparency, auditability, understandability, retention of semantics, and so on [2, 3]. The PROV standard of the W3C enables the "… inter-operable interchange of provenance information in heterogeneous environments such as the Web" [4]. Yet,

PROV, and the W3C's vision of the Semantic Web, never fully materialized in industry, due, in large part, to the complicated mathematics of the Web Ontology Language (OWL), which is a Description Logic (DL), a decidable subset of First-Order Logic (FOL). DLs find great success in research but lack adoptability in modeling for real-world software systems [5]. In any case, well-defined ontologies serialized in OWL, such as PROV-O, are useful as lexicons for the sharing of metadata between heterogeneous software.

The Basic Formal Ontology (BFO) is a general, Top-Level Ontology (TLO) following the Aristotelian notion of single inheritance, i.e., each class may be of type exactly one superclass. BFO is uniquely equipped for future work in that it is the first TLO to obtain International Organization for Standardization / International Electrotechnical Commission (ISO/IEC) standardization, in ISO/IEC series 21838 [6]. In contrast to PROV, BFO is recently maintained, rigorously validated and has seen success in the biomedical, defense and manufacturing industries. A preliminary mapping of the seminal PROV ontology with BFO may renew interest in the importance of provenance while maintaining an internationally recognized standard of representation.

In 2016, the FAIR (Findable, Accessible, Interoperable, Reusable) principles for scientific data were published [7]. The FAIR principles intend to facilitate data transparency, reusability, replicability and trust within research workflows. Faced with the black-box nature of large-scale generative Artificial Intelligence (AI) constructs on the Web, e.g., the Large Language Model (LLM) ChatGPT [8], structured knowledge forms, e.g., knowledge graphs predicated upon well-formed ontologies, have been shown to improve LLM reliability [9]. The present paper calls for a refreshing of the inspiration behind the Semantic Web movement of linked, structured knowledge, by mapping one of the most seminal ontologies, PROV-O, to the more recent and standardized BFO, in an effort to bring about FAIR data and data provenance.

2 Motivation

The principal motivation of the present paper is to engender thought, discussion and research around the topic of ontology standardization in the current year, which has seen a massive proliferation of large-scale generative AI constructs. Insofar as these constructs lack explainability due to the black-box nature of Machine Learning (ML), many researchers have posited the usefulness of structured knowledge being leveraged alongside large-scale generative AI, e.g., as knowledge graphs and LLMs have been demonstrated to possess synergy when incorporated [9].

Furthering this prime motivation are a number of other relevant concerns. The fulfillment of a proper mapping of the seminal PROV-O with the ISO/IEC standardized BFO is not only an incentive for researchers to continue work in the area of standardizing structured knowledge in the face of large-scale generative AI, but such a mapping may generally lend to:

1. Data integration and interoperability: mapped ontologies facilitate data integration from different sources, allowing systems to interoperate
2. Knowledge discovery and sharing: as many older projects utilized the PROV model, and newer industry work leverages BFO, a mapping of the two may facilitate greater discovery of past data

3. Standardization of semantics: provenance data are useful for retention of semantics; a common, standardized form, e.g., the ISO/IEC standardized BFO, keeps semantics uniform across heterogeneous systems
4. Reconciliation of old with new: the PROV project inspired and integrated with much scientific and industrial work; BFO is more recent and standardized for industry use; a reconciliation of the two may bridge the gap between old and new projects, academia and industry

3 Background

Knowledge exists in two types: implicit (tacit) and explicit [10]. Implicit knowledge is experience, stored in the mind of humans; explicit knowledge is serialized for re-use, represented through some medium. From Michael Bergman, a founder of the KBPedia project, Knowledge Representation (KR) is "… a field of artificial intelligence dedicated to representing information about the world in a form that a computer system can utilize to solve complex tasks" [11]. Central to KR is the classification of entities and their relations, which is the discipline of Ontology.

In computer science, an ontology (lowercase, with the indeterminate article preceding) is a serialized artifact which represents some knowledge [12]. From Gruber (1992), an ontology is an "explicit specification of a conceptualization"; Borst (1997) states that an ontology is a "formal specification of a shared conceptualization" [13, 14]. Arp, Smith and Spear (authors of the BFO handbook) define an ontology as "… a representational artifact, comprising a taxonomy as proper part, whose representations are intended to designate some combination of universals, defined classes, and certain relations between them" [15].

Ontologies exist in a spectrum of formality, e.g., from term lists to lexical thesauri, to database schema, to hierarchies, to DLs and, finally, to FOL [12, 16, 17]. The taxonomy, or hierarchy, is the canonical substrate of most ontologies, as most knowledge is hierarchical in nature [18]. To facilitate the vision of the Semantic Web, the W3C prescribed the Resource Description Framework (RDF), allowing the creation of knowledge graphs [19]. Building upon RDF, the Resource Description Framework Description Language, or RDF Schema (RDFS), allows for the creation of subsumption, i.e., subclass, hierarchies [20]. The final important piece of the Semantic Web technology stack is OWL, which allows the assertion of logical constraints on classes and properties, as well as inferencing with DL reasoners [21]. The combination of these three technologies, RDF, RDFS and OWL, are the form of the ontologies mapped in the present paper.

3.1 Mapping, Alignment, Merging

Ontology mapping may be considered the first step in the total process of merging two ontologies; e.g., two ontologies, O_1 and O_2 are first mapped, then aligned and finally, merged into a single, unified interpretation of some vocabulary in some domain of discourse [22]. I.e., ontology mapping, which is the focus of the present paper, is the first step in wholly reconciling two ontologies. An ontology mapping is a set of correspondences between the elements of two ontologies, where such correspondences may be equivalence relationships, sub-class or super-class relationships, transformation rules, etc. [23].

There are various approaches to ontology mapping, from the fields of machine learning, formal mathematics, linguistics and more; and there are various reasons for doing so, e.g., for academic purposes, or data interoperability in industry settings. In any case, ontology mapping is, at best, a semi-automated process, requiring significant expert insight and reasoning [22, 24].

3.2 The Provenance Ontology (PROV-O)

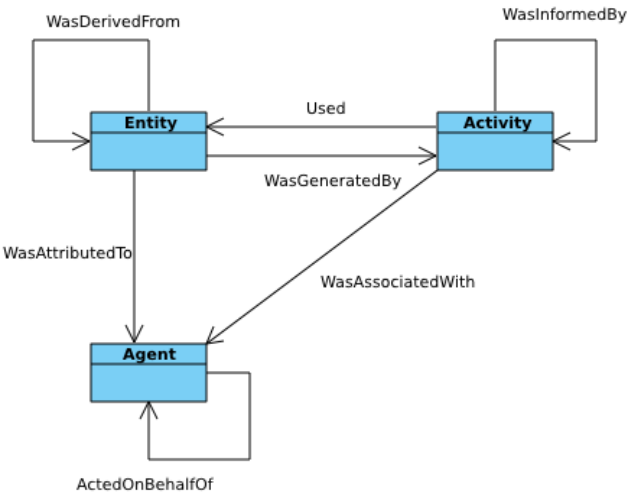

Fig. 1. The three Core, "Starting Point" classes of PROV, and their properties (from [1]).

The Open Provenance Model (OPM) was proposed in 2007 [25]. In 2011, its specification defines the OPM as an annotated causality graph, including its core nodes and edges, examples, constraint rules and implementations, e.g., RDF/RDFS/OWL [26]. In 2013, the PROV (short for PROVenance) working group, under the auspices of the W3C, completed its work on the PROV Data Model (PROV-DM), producing an OWL serialization of the Provenance Ontology called PROV-O, along with XML and textual serializations [27, 28]. See Fig. 1 for its Core classes and properties.

3.3 Basic Formal Ontology

Meant for extension by lower-level domain, task and application ontologies, TLOs represent very general entities, e.g., time, matter, space, events, etc. [29]. Three notable TLOs are the Unified Foundational Ontology (UFO) [30], KBPedia Knowledge Ontology (KKO) [11] and the Basic Formal Ontology (BFO), although others exist [31]. BFO is inspired by Aristotle, as its immediate dichotomy is between Continuant and Occurrent (very roughly, "thing" and "process") [15]. See Fig. 2. BFO follows the principle of single inheritance, i.e., that each entity can be of only one type of higher entity, such that each entity is traceable up to the most general entity through one path of inheritance.

The first ISO/IEC standardized TLO, BFO is the foundational ontology of the Open Biological and Biomedical Ontology (OBO) Foundry and Industrial Ontologies Foundry (IOF) [32], where dozens of ontologies from various researchers are hosted publicly for use in academic and industry work. The Descriptive Ontology for Linguis-

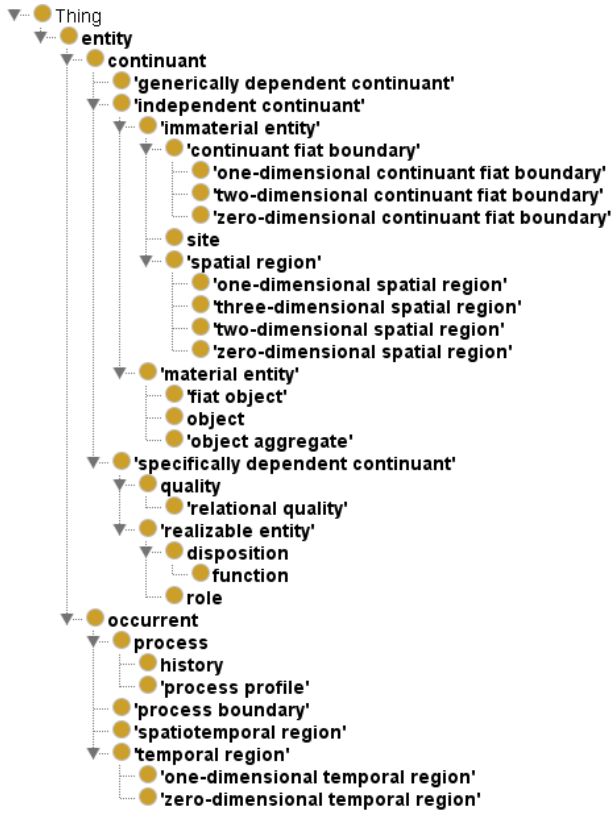

Fig. 2. The BFO mono-hierarchy (from Protégé).

tic and Cognitive Engineering (DOLCE) is also ISO/IEC standardized in series 21838-3, but it is rather dated and lacking modern adoption, as it was first published in 2009 [33]. A Top Level Ontology within Standards (TUpper) is also ISO/IEC standardized in series 21838-4, but it is too recent and lacking proof of efficacy in industry, being first published in 2022 [34].

So, in the present paper, BFO is the TLO chosen as the basis for mapping PROV because (1) it possesses the unique distinction of being the first ISO/IEC standardized TLO, (2) it is recently updated and maintained publicly, and (3) it has been successfully used in several industrial settings, e.g., the biomedical domain, defense and in manufacturing.

3.4 The Common Core Ontologies

Developed by the defense contractor, Calspan-University of Buffalo Research Center (CUBRC), the Common Core Ontologies (CCO) are a mid-level extension of the BFO taxonomy designed to represent generic classes and relations across many domains of interest, with a focus on military entities, e.g., aircraft, armies, cyberspace, operations, planning, geography, etc. [35]. The CCO are relevant to the present paper insofar as they prescribe mid-level terms that may allow more specific mappings from

PROV, as BFO is extremely general, covering only 35 classes; in contrast, the CCO covers several hundred classes, properties and individuals across its 13 ontologies. In particular, CCO's extensions of BFO's Generically Independent Continuant and Material Entity classes are highly applicable to provenance representation.

4 Mapping

PROV is to be mapped to, or extended from, the BFO/CCO taxonomy, for the purpose of elevating provenance representation out of the old form, and into an ISO/IEC compliant one.

Both the PROV model and BFO can be had in various serializations, e.g., XML, OWL, etc. BFO is also serialized in Common Logic (CL), which possesses the full expressivity of FOL [36]. For the purpose of the present paper, the OWL serializations of PROV and BFO are used, because the rendering of selected statements in the Terse Triple Language (TTL) format is amenable for both understanding and brevity; moreover, the CCO are only found in TTL format; so, OWL is most useful here. Following are the versions of each ontology utilized in this mapping:

- PROV-O: recommendation version 2013-04-30 [28]
- BFO: 2020, from 2024-01-29[1]
- CCO: 1.5, from 2024-02-14[2]

4.1 Mapping Technique

Bergman maintains a list of active ontology mapping tools as of 2018 [37]. In the present paper, the mapping from PROV to BFO is not performed with any specific software tool, or any purpose-written code. As PROV-O and BFO have essentially no lexical overlap, background knowledge is useful to a mapping between them [38]. The present mapping, as it is preliminary for the purpose of renewing interest in provenance in the face of large-scale generative AI, is conducted manually, using lexical analysis of ontology terms and their definitions, as well as the authors' experience with both ontologies as a form of background knowledge. Lexico-semantic reasoning for every mapping is given so that the method maintains some measure of replicability. Given the subjectivity of this method, the mapping is not declared as prescriptive, but rather, preliminary. The complete mapping, with an OWL file, is also hosted publicly in a GitHub repository[3].

PROV-O terms are mapped in sub-sections of the present paper, attendant with summarizing tables, following the canonical division of PROV-O categories:

- Core (Starting Point) terms
- Expanded terms
- Qualified terms

[1] https://github.com/BFO-ontology/BFO-2020/releases/tag/release-2024-01-29

[2] https://github.com/CommonCoreOntology/CommonCoreOntologies/releases/tag/v1.5-2024-02-14

[3] https://github.com/Semantic-Science/PROV-O_BFO_Mapping

PROV-O object properties are mapped in a final sub-section separately, as properties are not as easily mapped as classes. PROV-O data properties are ignored for the sake of simplicity.

In mapping, there are three properties used to express equivalence between classes and properties (respectively) in OWL: **`owl:equivalentClass`** and **`owl:equivalentProperty`** [23]. In the text, reference to these properties is in Protégé Manchester syntax, with the term **`equivalentTo`**. References to specific terms from PROV-O, BFO and CCO within the text are either directly, e.g., "the PROV class, Agent", or in TTL format, e.g., **`prov:Agent`**. These two forms are considered normative in the present paper and are therefore used interchangeably. The used prefixes, and their resolved URIs, are given below:

- **`owl:`** **`<http://www.w3.org/2002/07/owl#>`**
- **`prov:`** **`<http://www.w3.org/ns/prov#>`**
- **`bfo:`** **`<http://purl.obolibrary.org/obo/bfo#>`**
- **`cco:`** **`<http://www.ontologyrepository.com/CommonCoreOntologies/>`**

It is common in mapping to a broad TLO such as the CCO provide, that the ontology being mapped from (i.e., PROV-O), has certain classes or properties which are best introduced as new sub-classes in the extant hierarchy of the new ontology. Where applicable, this is denoted by asserting that the mapping should introduce a new subclass of some class, where introducing the new class may mean doing so under a new URI, or by directly copying the PROV-O class of interest into the new hierarchy.

4.2 Assumptions and Constraints

Four constraints on the present mapping are given below:

- All PROV-O classes will be mapped to the closest equivalent BFO/CCO class, asserted as sub-classes of the closest parent BFO/CCO class, or otherwise described as a transformation rule
- All PROV-O object and data properties (i.e., relations or predicates) will be mapped to the closest equivalent BFO/CCO property, or otherwise asserted as sub-properties of the closest parent BFO/CCO property
- Instances (**`owl:NamedIndividuals`**) are ignored for the sake of simplicity
- Annotation properties are ignored for the sake of simplicity
- No OWL reasoning is performed, as the mapping is preliminary and evolving

A principal issue in mapping PROV-O to BFO is that PROV-O allows multiple class inheritance, i.e., a class may be child, or type, of more than one class. This directly violates the Aristotelian notion which BFO is predicated upon, i.e., that each thing is of type exactly one other thing, such that all things are traceable up to the most general thing through one path of inheritance. BFO circumvents the tendency of the Linked Data specialist to allow multiple inheritance by providing the class Role, which allows entities to possess one or more roles, while still maintaining a single type.

4.3 PROV-O Core Class Mapping

The mapping begins with the three Core, or "Starting Point" classes of PROV-O. The Core classes are the basis of PROV-O, allowing for the creation of simple provenance descriptions.

A PROV Activity is defined as "something that occurs over a period of time and acts upon or with entities; it may include consuming, processing, transforming, modifying, relocating, using, or generating entities". With this definition, it is imprudent to prescribe the CCO class, Act, as the most appropriate term to map to, as it is defined as a "process in which at least one Agent plays a causative role". Although most uses of PROV Activity may, indeed, imply the role of some Agent, PROV's definition is general enough to include simple processes devoid of the influence of an Agent, e.g., deterioration. So, BFO's term, Process, is posited to be the most appropriate term for mapping **`prov:Activity`**, i.e., they are considered equivalent.

Table 1. Mapping of the Core PROV-O classes to BFO/CCO.

Core PROV-O Class	Equivalence or Transformation Rule
Activity	**`equivalentTo bfo:0000015 (Process)`**
Agent	Introduce as new subclass of **`bfo:0000023 (Role)`**
Entity	**`equivalentTo bfo:0000002 (Continuant)`**

A PROV Agent is defined as "something that bears some form of responsibility for an activity taking place, for the existence of an entity, or for another agent's activity". A CCO Agent is defined as "A Material Entity that is capable of performing Planned Acts". Notably, **`prov:Agent`** has three subclasses: **`prov:Organization`**, **`prov:Person`** and **`prov:SoftwareAgent`**. It is not appropriate to assert that **`prov:Agent`** is equivalent to **`cco:Agent`** because, although every **`prov:Agent`** is either a Person, Organization or some piece of Software, not every Person, Organization or piece of Software is an Agent at every point in time. Additionally, asserting this implies that the sub-classes of **`prov:Agent`** are also sub-classes of **`cco:Agent`**, which would place, for instance, **`prov:Person`** in conflict with **`cco:Person`**, which exists in a separate sub-tree. Granting these concerns, the most appropriate mapping is to assert that PROV's conception of Agent is a new sub-class of BFO Role, e.g., AgentRole, similarly to the extant **`cco:OperatorRole`**. The remaining issue is that **`cco:Agent`** exists along with some new AgentRole, i.e., it may be considered redundant. A possible reason for simply ignoring **`cco:Agent`** is that it is defined as the DL constraint **`equivalent to bfo:000040 and (cco:agent_in some bfo:000015)`**, where **`bfo:000040`** is a BFO Material Entity and **`bfo:000015`** is a BFO Process. A **`cco:Agent`** is logically the manifestation of some Material Entity engaged in some Process, as some Organization, Person or piece of Software engaged in a process while bearing some AgentRole would be.

A PROV Entity is defined as a "physical, digital, conceptual, or other kind of thing with some fixed aspects; entities may be real or imaginary". This is not to be confused with a BFO Entity, which is the most general class in BFO. The closest mappable

class, given the added "real or imaginary" suffix in the definition, is simply BFO Continuant. It is possible that, in most cases, the subclass of Continuant, BFO Independent Continuant, is applicable, but the words "conceptual" and "imaginary" imply the possibility of BFO Generically Dependent Continuant, so the subsuming class, BFO Continuant, is most appropriate.

4.4 PROV-O Expanded Class Mapping

Built atop the Core category of terms, PROV-O Expanded terms allow for more fine-grained provenance descriptions.

A PROV Organization is defined as a "a social or legal institution such as a company, society, etc.", where a CCO Organization is defined as "A Group of Agents which can be the bearer of roles, has members, and has a set of organization rules". Given these definitions, it can be reasonably posited that the two terms are equivalent. A PROV Person is defined as "Person agents are people", where a CCO Person is defined as "An Animal that is a member of the species Homo sapiens". The obvious contention with simply asserting that **`prov:Person`** is equivalent to **`cco:Person`** is that **`prov:Person`** is sub-class of **`prov:Agent`**, where **`cco:Person`** is sub-class to **`cco:Animal`**, then **`cco:Organism`**, then BFO Object, and on. The same reasoning can be applied with asserting that **`prov:Organization`** is equivalent to **`cco:Organization`**. Notwithstanding, **`prov:Person`** is asserted as equivalent to **`cco:Person`** because the **`prov:Agent`** super-class contention has been addressed in the Core class mapping, by mapping **`prov:Agent`** to a new sub-class of BFO Role.

Table 2. Mapping of the Expanded PROV-O classes to BFO/CCO.

Expanded PROV-O Class	Equivalence or Transformation Rule
Organization	**`equivalentTo cco:Organization`**
Person	**`equivalentTo cco:Person`**
SoftwareAgent	Introduce as new sub-class of **`cco:InformationProcessingArtifact`**
Bundle	Introduce as new sub-class of **`bfo: 0000002 (Continuant)`**
Collection	**`equivalentTo bfo:0000115 min 1 bfo:0000002 (Continuant)`**
EmptyCollection	**`equivalentTo bfo:0000115 max 0 bfo:0000002 (Continuant)`**
Location	Introduce as new sub-class of **`cco:DesignativeInformationContentEntity`**

The term **`prov:SoftwareAgent`** presents a more complicated mapping case. This term is defined as "A software agent is running software". Considering the "running software" fragment, the CCO term, Algorithm, under Directive Information Content Entity (ICE), is not appropriate, as ICEs are not material, because they are just the informational content of Information Bearing Entities/Artifacts, which are material. In any case, PROV considers SoftwareAgent as something material, as the software is defined as "running". The closest subsuming term is, therefore, **`cco:InformationProcessingArtifact`**, under which a new sub-class, e.g., Software, may be introduced. The sibling class, **`cco:InformationBearingArtifact`**, may also be considered; but, as software is

executable, it is more reasonable to assert that it is an Artifact of processing, as opposed to bearing, information.

A PROV Bundle is defined as "a named set of provenance descriptions, and is itself an Entity, so allowing provenance of provenance to be expressed". The PROV-O recommendation document also states that "there are kinds of bundles (e.g. handwritten letters, audio recordings, etc.) that are not expressed in PROV-O, but can be still be described by PROV-O" [28]. Although provenance is itself metadata, a PROV Bundle is distinct from a PROV Collection insofar as a Bundle is a metadata term, meant for organizing, or packaging, PROV descriptions, including Agents, Activities and Entities, that can also contain other Bundles; i.e., Bundles can be recursive. A PROV Collection is also distinct in that it is described specifically with the property **prov:hadMember**, where a Bundle can be described more loosely, with the PROV-O document not specifying any particular properties. So, an appropriate mapping is to introduce a new subclass of BFO Continuant, e.g., Bundle (but, ideally, something with a more descriptive name, e.g., ProvenanceBundle), asserting it as equivalent to **prov:Bundle**. In the terms of Arp, Smith and Spear, a PROV Bundle is a very fiat[4] term, insofar as it is specifically used to designate a contrived thing [15]; but it is mapped regardless to maintain as complete a mapping as possible.

A PROV Collection is defined as "an entity that provides a structure to some constituents, which are themselves entities. These constituents are said to be member of the collections". The closest term is BFO Object Aggregate, but this term is for an aggregate of Material Entities, where PROV Entities can be material or immaterial. So, the most appropriate mapping for **prov:Collection** is to introduce a new subclass of BFO Continuant, e.g., Collection, and assert it as the DL constraint **equivalentTo bfo:0000115 min 1 bfo:0000002** where the property **bfo:0000115** is labelled as **has member part** and the class **bfo:0000002** is BFO Continuant. A PROV EmptyCollection is defined as "a collection without members". Similar to a PROV Collection, the most appropriate mapping is to assert it as the DL constraint **equivalentTo bfo:0000115 max 0 bfo:0000002**.

PROV states that a Location can be "[A]n identifiable geographic place (ISO 19112), but it can also be a non-geographic place such as a directory, row, or column. As such, there are numerous ways in which location can be expressed, such as by a coordinate, address, landmark, and so forth". The CCO has terms Artifact Location and Geospatial Location, which are defined as "A Site that is the location of some Artifact" and "A Geospatial Region at which an Entity or Event is located", respectively. Artifact Location may seem to be the appropriate term to map to, but **prov:Entity** was previously mapped to BFO Continuant, so the definition of Artifact Location, which specifies "some Artifact", would conflict with this earlier mapping, as it precludes immaterial entities from being located. Clearly, a Location is not a BFO Occurrent, as it does not unfold in time, so that half of the hierarchy can be ignored. A Location cannot be a BFO Specifically Dependent Continuant, since it is not a quality, or something realizable, of other entities. Furthermore, a PROV Location is not intended to *be* some location, but to represent it with an identifier, such as a name; i.e., it is a designator, not the location itself, so BFO Independent Continuant

[4] "Fiat", translated from Latin, means "let it be done". In terms of Ontology, a fiat term is one declared by human decision or societal convention rather than by natural boundaries.

can also be ignored. Therefore, a PROV Location is best mapped to a BFO Generically Dependent Continuant, specifically, the CCO extension, Designative Information Content Entity, by introducing a new sub-class of Designative Information Content Entity, e.g., Location.

4.5 PROV-O Property Mapping

PROV-O data and object properties are mapped separately from classes, as BFO and the CCO prescribe very few properties with a limited set of disconnected hierarchies, so mapping is not so direct. Additionally, one must consider the domains and ranges of properties, which affect their use with respect to the newly mapped classes; and, once mapped, the domains and ranges must be mapped as well. Granting these constraints, many of the properties are simply "binned" under extant CCO properties for organizational purposes.

Many CCO and PROV-O super-properties, e.g., **`prov:wasInfluencedBy`**, both note that they should not be used directly when modeling; instead, their more specific sub-properties should be used. I.e., many of the top-level properties in both the CCO and PROV-O exist only for organizational purposes, acting as bins for groups of related sub-properties, so, the mapping of properties is not so formal as with classes.

Moreover, many of the properties cannot be mapped to anything prescribed by BFO/CCO, so they are, at best, simply re-used with type **`owl:ObjectProperty`** or **`owl:DataProperty`**, i.e., they would not be placed within any property hierarchy, save for those already extant in PROV-O, e.g., as **`prov:wasInfluencedBy`** has several child properties. These are indicated with the phrase "Import from PROV-O".

Table 3. Mapping of the Core PROV-O properties to BFO/CCO.

Core PROV-O Property	Equivalence or Transformation Rule
wasGeneratedBy	Introduce as new sub-property of **`bfo:0000056`** (**`participates in`**)
wasDerivedFrom	Import from PROV-O
wasAttributedTo	Not directly mappable
startedAtTime	Introduce as new sub-property of **`cco:has_datetime_value`**
used	Import from PROV-O
wasInformedBy	Import from PROV-O
endedAtTime	Introduce as new sub-property of **`cco:has_datetime_value`**
wasAssociatedWith	Introduce as new sub-property of **`bfo:0000056`** (**`participates in`**)
actedOnBehalfOf	Import from PROV-O

PROV-O Core Property Mapping. The closest property to **`prov:wasDerivedFrom`** is **`cco:is_successor_of`**, but the domain and range of this property are both BFO Independent Continuant, which precludes all other Continuants (or Entities, in the nomenclature of PROV). **`prov:wasDerivedFrom`** therefore cannot be directly mapped, and the same is true for **`prov:wasAttributedTo`**. **`prov:used`** may seem close to **`cco:uses`**, but the former has a domain of prov:Activity and a range of **`prov:Entity`**, where the latter has a domain of

cco:Agent and a range of Material Entity, so the two cannot be directly mapped. **prov:wasInformedBy** has a note, "An activity a2 is dependent on or informed by another activity a1, by way of some unspecified entity that is generated by a1 and used by a2". This places the term similarly to **cco:is_cause_of**, but **prov:wasInformedBy** is not so strong as to imply cause, but simply influence. Therefore, the term cannot be directly mapped. **prov:wasAssociatedWith** has a domain of **prov:Activity** and a range of **prov:Agent**, placing it close to **cco:agent_in**; but, the definition of **cco:agent** is too specific to allow this assertion, so the most appropriate mapping is to introduce a new sub-property of its super-property, **cco:participates_in**. **prov:actedOnBehalfOf** is used to demarcate where a subordinate Agent acts at the behest of a responsible Agent. There is no close BFO/CCO property, so **prov:actedOnBehalfOf** cannot be directly mapped.

Table 4. Mapping of the Expanded PROV-O properties to BFO/CCO.

Expanded PROV-O Property	Equivalence or Transformation Rule
alternateOf	Import from PROV-O
specializationOf	Import from PROV-O
generatedAtTime	Introduce new sub-property of **cco:has_datetime_value**
hadPrimarySource	Import from PROV-O
value	Introduce new data property and set all CCO has_x_value as its sub-properties
wasQuotedFrom	Import from PROV-O
wasRevisionOf	Import from PROV-O
invalidatedAtTime	Introduce new sub-property of **cco:has_datetime_value**
wasInvalidatedBy	Import from PROV-O
hadMember	**owl:equivalentProperty bfo:0000115 (has member part)**
wasStartedBy	Import from PROV-O
wasEndedBy	Import from PROV-O
invalidated	Import from PROV-O
influenced	Import from PROV-O
atLocation	Import from PROV-O
generated	Introduce new sub-property of **bfo:0000057** (**has participant**)

PROV-O Expanded Property Mapping. Many of the expanded PROV-O properties cannot be directly mapped to, or under, any BFO/CCO properties, so they can be imported directly. The two properties dealing with time, **prov:generatedAtTime** and **prov:invalidatedAtTime**, can simply be placed as sub-properties of **cco:has_datetime_value**. The CCO have eleven data properties for data values, e.g., **cco:hasAltitudeValue**, **cco:hasBooleanValue**, etc. To map **prov:value**, which is semantically a subsuming term to these CCO has_x_value properties, it is appropriate to introduce a new property, e.g., value, and set all CCO has_x_value properties to its sub-properties. **prov:hadMember** is directly mappable

to the BFO property labelled **has member part**. **prov:generated** is mappable as a sub-property of the BFO property labelled **has participant**.

prov:atLocation reads similarly to **cco:occurs_at**, which has a domain of BFO Process and a range of BFO Site; it is also similar to **bfo:0000171**, which is labelled **located in**. But these two properties are in terms of Material Entities and Independent Continuants, where the PROV-O notion of a Location allows for entities not necessarily material, e.g., database rows. So, **prov:atLocation** is mapped simplest by importing it directly.

4.6 PROV-O Qualified Terms Mapping

The Qualified classes and properties of PROV-O are used to provide elaborated information about asserted relations between Core and Expanded terms, using a different design pattern, the Qualified Relation. Given the complexity inherent in this pattern [39], and the large quantity of Qualified terms, a mapping of these terms is out of the scope of the present paper. Notwithstanding, all PROV-O terms, including the qualified terms, have been mapped and are placed within a public GitHub repository[5].

5 Discussion and Limitations

The present mapping is preliminary, serving as an introduction to the idea of renewing the Semantic Web ideal of machine-readable provenance, by elevating the W3C PROV standard to the more recently ISO/IEC standardized BFO. As argued throughout the present paper, such a mapping may renew scientific interest in machine readable provenance amid the current zeitgeist of large-scale generative AI and its massive deluge of data. What researchers do with this mapping remains to be seen, but it is hoped that this early effort will spawn continued research in the domain of TLO mapping for FAIR provenance of large AI systems.

In any case, there are several limitations to the present mapping. First, the mapping is insular, insofar as no input, save for the authors', contributed to the reasoning underlying the chosen mappings. The lack of codified background knowledge between the two ontologies, PROV-O and BFO/CCO, resulted in the subjective use of the authors' expertise as background knowledge, which is the most basic starting point for mapping [38]. Ideally, a greater diversity of thought, and an iterative life cycle incorporating more ontologists, would be employed to arrive at a more correct and complete mapping. The use of an ontology matching tool, such as one detailed by Bergman, may also provide a more replicable baseline for the mapping [37]. Next, the mapping is not exhaustive, nor is it complete. Instances in both PROV-O and BFO/CCO are ignored, as well as annotation properties; and the DL constraints asserted in the present paper to map particular classes have not been validated with any OWL reasoning. Ontology mapping efforts have a dearth of evaluation metrics, typically employing standard ML metrics like precision and recall [38]. There is no evaluation of the present mapping provided, as it is preliminary.

[5] https://github.com/Semantic-Science/PROV-O_BFO_Mapping

Mapping PROV-O to BFO/CCO is an effort in lexical matching and semantic disambiguation, primarily by referencing the stated definitions and examples from each ontology. Save for the three core PROV-O classes, there is very little terminological overlap between the two ontologies, so much of the mapping manifests in DL equivalence rules, transformation rules, or the introduction of PROV-O terms as sub-classes or sub-properties of extant BFO/CCO terms. Also, PROV-O, as it is an ontology arising from the early stages of the Linked Data and Semantic Web movement, allows for multiple type inheritance, where BFO is Aristotelian, allowing mono-inheritance. This is addressed in a few cases, e.g., with the Core term, Agent, by advocating for the use of BFO's Role class, but discourse between future researchers is necessary to arrive at a unanimously accepted mapping.

Future research in this area should include a more rigorous mapping of the two ontologies, using what is reported here as a baseline epistemological framework from which to refute, modify and advance the mapping, to the eventual extent that the seminal PROV-O can be accurately represented within the ISO/IEC standardized model which BFO provides. This is important, as, for instance, both BFO and PROV-O have been used in tandem for scientific research provenance documentation, without any deep reconciliation of the two, i.e., without a proper mapping, by following the Linked Data specialist tendency to simply use terms as deemed necessary for some particular project model [40]. Some have called for the integration of large-scale generative AI constructs with the scientific research workflow, which is a contentious position, but nevertheless likely as such constructs demonstrate greater capabilities [41]. A philosophical and technical reconciliation of PROV-O with BFO would more appropriately place all future provenance work with PROV-O into ISO/IEC standardization, allowing for a FAIR-er scientific research landscape.

6 Conclusion

The present paper delivers a preliminary mapping of the W3C's seminal PROV-O to the more recent, ISO/IEC standardized OBO Foundry's BFO. The authors believe that such a mapping may renew scientific interest in the important factor of machine-readable data provenance, in response to the recent explosion of large-scale generative AI and its attendant deluge of data. The mapping of PROV-O to BFO is preliminary and intentionally open-ended, as it is hoped other researchers may build upon the work introduced here with philosophical debate and scientific discourse, to bring about standardized, FAIR data and well-represented data provenance in the current zeitgeist of massive AI.

Supplemental Material Statement: A full mapping, including a serialized OWL file and the PROV-O Qualified terms, which are out of the scope of the preliminary mapping provided in the present paper, can be found publicly on a GitHub repository: https://github.com/Semantic-Science/PROV-O_BFO_Mapping

References

[1] L. Moreau, P. Missier, K. Belhajjame, R. B'Far, J. Cheney, S. Coppens, S. Cresswell,

Y. Gil, P. Groth, G. Klyne, T. Lebo, J. McCusker, S. Miles, J. Myers, S. Sahoo and C. Tilmes, "PROV-DM: The PROV Data Model," W3C, 2013. [Online]. Available: https://www.w3.org/TR/2013/REC-prov-dm-20130430/. [Accessed April 2024].

[2] M. Herschel, R. Diestelkämper and H. Ben Lahmar, "A survey on provenance: What for? What form? What from?," *The VLDB Journal*, vol. 26, pp. 881-906, 2017.

[3] S. N. Mitchell, A. Lahiff, N. Cummings, J. Hollocombe, B. Boskamp, R. Field, D. Reddyhoff, K. Zarebski, A. Wilson, B. Viola, M. Burke, B. Archibald, P. Bessell and R. Blackwell, "FAIR data pipeline: provenance-driven data management for traceable scientific workflows," *Philosophical Transactions of the Royal Society A*, vol. 380, no. 2233, 2022.

[4] P. Groth and L. Moreau, "PROV-Overview," W3C, 30 April 2013. [Online]. Available: https://www.w3.org/TR/2013/NOTE-prov-overview-20130430/. [Accessed April 2024].

[5] I. Polikoff, "Why I Don't Use OWL Anymore," TopQuadrant, [Online]. Available: https://archive.topquadrant.com/owl-blog/. [Accessed April 2024].

[6] "ISO/IEC 21838-2:2021 Information technology — Top-level ontologies (TLO) — Part 2: Basic Formal Ontology (BFO)," International Organization for Standardization, November 2021. [Online]. Available: https://www.iso.org/standard/74572.html. [Accessed April 2024].

[7] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L. B. da Silva Santos, P. E. Bourne, J. Bouwman, A. J. Brookes and T. Clark, "The FAIR Guiding Principles for scientific data management and stewardship," *Scientific Data*, vol. 3, no. 1, pp. 1-9, 2016.

[8] OpenAI, "GPT-4 Technical Report," *arXiv:2303.08774*, 2023.

[9] S. Pan, L. Luo, Y. Wang, C. Chen, J. Wang and X. Wu, "Unifying large language models and knowledge graphs: A roadmap," *IEEE Transactions on Knowledge and Data Engineering*, 2024.

[10] C. McInerney, "Knowledge Management and the Dynamic Nature of Knowledge," *Journal of the American Society for Information Science and Technology*, vol. 53, no. 12, pp. 1009-1018, 2002.

[11] M. K. Bergman, A Knowledge Representation Practionary: Guidelines, Springer International Publishing, 2018.

[12] N. Guarino, D. Oberle and S. Staab, "What is an Ontology?," in *Handbook on Ontologies*, 2009, pp. 1-17.

[13] T. R. Gruber, "A Translation Approach to Portable Ontology Specifications," *Knowledge Acquisition*, vol. 5, no. 2, pp. 199-220, 1993.

[14] W. Borst, "Construction of Engineering Ontologies," *Institute for Telematica and Information Technology*, 1997.

[15] R. Arp, B. Smith and A. D. Spear, Building Ontologies with Basic Formal Ontology, Cambridge: The MIT Press, 2015.

[16] O. Lassila and D. McGuinness, "The role of frame-based representation on the

semantic web," 2001.

[17] M. Uschold and M. Gruninger, "Ontologies and Semantics for Seamless Connectivity," *SIGMOD Record*, vol. 33, no. 4, pp. 58-64, 2004.

[18] T. T. Procko, T. Elvira and O. Ochoa, "GPT-4: A Stochastic Parrot or Ontological Craftsman? Discovering Implicit Knowledge Structures in Large Language Models," *2023 Fifth International Conference on Transdisciplinary AI (TransAI)*, pp. 147-154, 2023.

[19] O. Lassila and R. Swick, "Resource Description Framework (RDF) Model and Syntax Specification, W3C Recommendation," World Wide Web Consortium, 1999. [Online]. Available: https://www.w3.org/TR/REC-rdf-syntax/. [Accessed 2023].

[20] D. Brickley, R. V. Guha and A. Layman, "Resource description framework (RDF) schema specification," *W3C*, 1999.

[21] "OWL 2 web ontology language," W3C OWL Working Group, 11 December 2012. [Online]. Available: https://www.w3.org/TR/owl2-overview/. [Accessed 2021].

[22] Y. Kalfoglou and M. Schorlemmer, "Ontology mapping: the state of the art," *The knowledge engineering review*, vol. 18, no. 1, pp. 1-31, 2003.

[23] N. F. Noy, "Ontology Mapping," in *Handbook on Ontologies*, Berlin, Springer, 2009, pp. 573-590.

[24] M. Ehrig and Y. Sure, "Ontology Mapping - An Integrated Approach," in *European semantic web symposium*, vol. 3053, Berlin, Springer, 2004, pp. 76-91.

[25] L. Moreau, J. Freire, J. Futrelle, R. E. McGrath, J. Myers and P. Paulson, "The Open Provenance Model," 2007.

[26] L. Moreau, B. Clifford, J. Freire, J. Futrelle, Y. Gil, P. Groth, N. Kwasnikowska, S. Miles, P. Missier, J. Myers, B. Plale, Y. Simmhan, E. Stephan and J. Van den Bussche, "The Open Provenance Model Core Specification (v1.1)," *Future Generation Computer Systems*, 2010.

[27] "Provenance Working Group," W3C, 19 June 2013. [Online]. Available: https://www.w3.org/2011/prov/wiki/Main_Page. [Accessed April 2024].

[28] T. Lebo, S. Sahoo, D. McGuinness, K. Belhajjame, J. Cheney, D. Corsar, D. Garijo, S. Soiland-Reyes, S. Zednik and J. Zhao, "PROV-O: The PROV Ontology," W3C, 30 April 2013. [Online]. Available: https://www.w3.org/TR/prov-o/. [Accessed April 2024].

[29] N. Guarino, "Formal Ontology in Information Systems," *Proceedings of the First International Conference (FOIS '98)*, June 1998.

[30] G. Guizzardi, G. Wagner, J. Paulo Andrade Almeida and R. SS Guizzardi, "Towards ontological foundations for conceptual modeling: The unified foundational ontology (UFO) story," *Applied ontology*, vol. 10, no. 3, pp. 259-271, 2015.

[31] C. Partridge, A. Mitchell, A. Cook, J. Sullivan and M. West, "A Survey of Top-Level Ontologies: To Inform the Ontological Choices for a Foundation Data Model," *Centre for Digital Built Brain*, 2020.

[32] S. Borgo, A. Galton and O. Kutz, "Foundational ontologies in action," *Applied*

Ontology, vol. 17, no. 1, pp. 1-16, 2022.

[33] S. Borgo and C. Masolo, "Foundational choices in DOLCE," in *Handbook on Ontologies*, Berlin, Springer, 2009, pp. 361-381.

[34] M. Grüninger, Y. Ru and J. Thai, "TUpper: A top level ontology within standards," *Applied Ontology*, vol. 17, no. 1, pp. 143-165, 2022.

[35] CUBRC, "An Overview of the Common Core Ontologies," *nist.gov*, 2019.

[36] A. Ruttenberg, "First-Order Logic Based Implementation," Basic Formal Ontology, 7 April 2020. [Online]. Available: https://basic-formal-ontology.org/fol.html. [Accessed April 2024].

[37] M. K. Bergman, "30 Active Ontology Alignment Tools," 22 January 2018. [Online]. Available: https://www.mkbergman.com/2129/30-active-ontology-alignment-tools/. [Accessed April 2024].

[38] K. Ramar and G. Gurunathan, "Technical review on ontology mapping techniques," *Asian Journal of Information Technology*, vol. 15, no. 4, pp. 676-688, 2016.

[39] L. Dodds and I. Davis, "Qualified Relation," 31 May 2012. [Online]. Available: http://patterns.dataincubator.org/book/qualified-relation.html. [Accessed April 2024].

[40] M. Schröder, S. Staehlke, P. Groth, J. B. Nebe, S. Spors and F. Krüger, "Structure-based knowledge acquisition from electronic lab notebooks for research data provenance documentation," *Journal of Biomedical Semantics*, vol. 13, no. 4, 2022.

[41] T. T. Procko, A. Davidoff, T. Elvira and O. Ochoa, "Towards Improved Scientific Knowledge Proliferation: Leveraging Large Language Models on the Traditional Scientific Writing Workflow," *Available at SSRN 4594836*, 2023.