Triggers and Bases: Extending BFO to Represent the Pahl–Beitz Framework for Working Principles

Ludger Jansen[1,2,*] and Dilek Yargan[2]

[1] *PTH Brixen College, Piazza Seminario 4, 39042 Bressanone, Italy*
[2] *University of Rostock, Institute of Philosophy, 18051 Rostock, Germany*

Abstract
It is an influential position in engineering design theory that one of the most important steps in conceptual design is the search for working principles. There is, however, no clear statement in the engineering literature about what a working principle is and how knowledge about working principles can be accounted for ontologically. We propose that dispositions are at the core of working principles, and thus that knowledge about working principles is knowledge about dispositions, their material bearers, and their triggers. Building on the engineering design framework developed by Pahl and Beitz, we suggest ontology design patterns for representing knowledge about working principles in a BFO-conformant manner. To represent trigger classes, we introduce the relation **has trigger** and its inverse **trigger of**, and discuss their limitations.

Keywords
working principles, engineering design, dispositions, Basic Formal Ontology, trigger, ontology design patterns

1. Introduction

Engineering design theory seeks to describe the creative process in engineering development projects. One of the most important textbooks is the monograph of the German engineering professors Gerhard Pahl (1925–2015) and Wolfgang Beitz (1935–1998). Their book originally appeared in 1977 [1], has been revised and enlarged throughout several editions, and has been translated into eight languages, including English [2], and is now continued by other researchers in the field [3]. According to the Pahl–Beitz account, one of the crucial steps of the construction process is to find "working principles" ("Wirkprinzipien") to implement the functions required for the intended goal. To implement the function to store energy, for example, a range of possible working principles are available: rotation of a flywheel, charging of an electrical battery, or heating water or even overheating water vapour ([2], p. 96). Pahl and Beitz distinguish between various kinds of knowledge that will be relevant to this purpose, among which are, most prominently, working material, working geometry, and working motions. It is not clear, however, how these can be represented in an ontology. The suggestions

Proceedings of the Joint Ontology Workshops (JOWO) - Episode XI: The Sicilian Summer under the Etna, co-located with the 15th International Conference on Formal Ontology in Information Systems (FOIS 2025), September 8-9, 2025, Catania, Italy.
* Corresponding author.
ludger.jansen@pthsta.it (L. Jansen); dilek.yargan@uni-rostock.de (D. Yargan)
0000-0002-0097-6359 (L. Jansen); 0000-0001-9618-6740 (D. Yargan)

in this paper rest on the assumption that dispositions are at the core of working principles, and that knowledge about working principles is knowledge about dispositions. The challenges involved in representing working principles as dispositions bring back, it seems, certain issues that have been discussed at length in philosophy. In the last decades, dispositions, tendencies, or powers have increasingly been acknowledged as explanatory entities in metaphysics and the philosophy of science [4–6]. In philosophy, dispositions are often described by means of ordered pairs of trigger types and realization types. Moreover, much discussion centred around the question of the categorical basis of dispositions.

While the issue of the material basis of dispositions has recently been accounted for in BFO in an insufficient way only, there is still no means to represent the triggers of a disposition. This paper suggests extensions to BFO to address these two issues and also suggests ontology design patterns for working principles based on the Pahl–Beitz account. Doing so, it also contributes to the representation of dispositions within BFO, evaluating and extending it in light of the philosophical discussions.

The remainder of the paper is organised as follows. Section 2 analyses working principles in light of the engineering design literature. Section 3 describes the BFO account of dispositions and the pattern suggested by Röhl and Jansen [7] for the representation of dispositions. Section 4 shortly describes our method, exploring the difference between introducing classes and relations into an ontology and using more complex patterns to describe certain phenomena. Section 5 then presents patterns for representing knowledge about working materials, working shape, working size, working motions, and combinations of these. In Section 6, we evaluate the suggested patterns using competency questions. Section 7 discusses the trigger relation used, and Section 8 concludes the paper by pointing out some limitations of our present results.

In this paper, class names are written in *italics* along with their corresponding namespaces, e.g., *BFO:material entity*. Upon the first mention of a class name, also its OBO ID is cited as a unique identifier within square brackets, e.g., *material entity* [BFO:0000040]. Additionally, the full IRI (Internationalized Resource Identifier) of a class, e.g., for *BFO:material entity*, http://purl.obolibrary.org/obo/BFO_0000040, is hyperlinked to the reference information within the square brackets. Relation names are written in bold, e.g., **BFO:bearer of**, when using the Web Ontology Language (OWL), but in italics, e.g., *base_of*, when using the legacy OBO format (see Section 3). We mostly refrain from citing the namespaces of **rdfs:subClassOf** and **owl:equivalentClass**. Lastly, the logical connectors are in small all-caps, e.g., NOT or SOME.

2. Working Principles in Engineering Design

There is a wide consensus in engineering design that the central step in the conceptual phase of the design process is the search for working principles. This is witnessed by the influence of the Pahl–Beitz approach on engineering guidelines in Germany on engineering design (VDI 2221 [8, 9], VDI 2222-2 [10]) and biomimetics (VDI 6220-1 [11]), and by the continued international success of the textbook. Terminology is not uniform, though [12], and there are approaches like design thinking where it is not required to find working principles at all [13].

According to the Pahl–Beitz approach, engineering design starts with a functional analysis, breaking down the overall goal to be achieved into a system of subfunctions. In a second step, the engineer needs to figure out how these subfunctions can be set to work; this is the step that is described as the search for working principles. Working principles are to be sought for each subfunction and then combined with the overall working structure of the device to be constructed. Normally, there might be several working principles available, which then have to be evaluated by the engineer for their compatibility and how well they match the requirements in terms of their causal efficiency, energy efficiency, cost efficiency, side-effects, and environmental friendliness. For instance, the function to store energy can be fulfilled by diverse working principles, such as charging a battery or stretching a metal string. There have been attempts to collect and systematise working principles in "solution catalogues", which were, however, limited by the linear logic of the printed book [14, 15].

Despite widespread use of the term, there is hardly any ontological analysis of working principles that can be found in the literature. According to Pahl et al. [2], working principles are the causal principles that bring about the intended effects. Obviously, the term "principle" does not help an ontological analysis. More promising is to take up the causal aspect linked to working principles. In an ontological framework, dispositions are a typical locus for causal relations: an event e_1 causing an event e_2 can be analysed in terms of a mediating disposition *d,* in terms of e_1 triggering *d* to realise itself in e_2. Against this background, it looks promising to understand knowledge about working principles as knowledge about dispositions. Two remarks to clarify this suggestion are in order here. First, this does not mean that we want to equate working principles with a knowledge item, as working principles are something causally efficient in the world, and biomimetic researchers, for example, study living nature to discover new working principles to be transferred to technical solutions [16]. Second, we do not simply equate working principles and dispositions either, as knowing a working principle can also involve knowledge about other entities, like a shape, a material, or a movement. Our dispositional account thus also involves non-dispositional entities.

Jansen [12] shows that the dispositional account allows to integrate the various ways that Pahl et al. [2] use to describe working principles: sometimes they use names, sometimes they give formulae for the laws of nature employed, and nearly always they use drawings or diagrams to describe the working principles in question. Often, they use all three methods in combination. Except for the proper names of working principles (like lever effect, the electromagnetic effect, the hydraulic effect, etc.), these ways to describe working principles can be translated into talking about dispositions [12]. References to laws of nature, for example, can in a first approximation be transformed into a characterisation of a disposition in terms of trigger and realisation processes. In the simplest case of a law that describes the dependence of the value of a certain dimension (say, y) on exactly one variable only (say, x), it is a law of the form "$y = f(x)$". An object that is subjected to this law is, then, the bearer of the disposition to display a y of the value $f(x)$ if triggered by x.

In this paper, we will develop patterns for the representation of working principles knowledge based on this dispositional approach. Pahl et al. mention three "classifying criteria" or "facets" for working principles: working material, working geometry, and working movements ([2], p. 94). Knowing that copper conducts electricity is to know a working material, knowing that balls roll is to know a working geometry; and knowing that pushing a billiard ball with the stick makes the ball roll is to know a working movement.

Bochtler [17] seems to conceive of these as three subtypes of working principles. This does not do justice to the complexity of some engineering solutions, as sometimes all three facets have to be combined to describe a given working principle. Also, it does not seem to be promising to think of "facets" itself as a coherent ontological category. Instead, we hypothesize that they have to be construed as three aspects of working principles, i.e., as different types of knowledge about the dispositions involved. In a given case, it might be advisable to combine these varieties of knowledge with each other. As Pahl et al. put it, "a working interrelationship comes into existence through physical effects in combination with the chosen geometric and material characteristics" ([2], p. 38). In fact, they need to be combined, as Pahl et al. point out: "Only the combination of the physical effect with the geometric and material characteristics (working surfaces, working motions and materials) allows the principle of the solution to emerge" ([2], p. 40). For example, for some cases it might be sufficient to know that copper is an electrical conductor in order to design a certain machine, i.e., it is a certain working material that is used. This is based on a certain disposition of copper, namely the disposition to allow the flow of electricity. However, if a high voltage current is to be transmitted, the copper cable in question has to have a certain minimal diameter, i.e., the working material has to be combined with a certain working size. This is because only copper cables with a certain diameter have the disposition to conduct high voltage without being destroyed. In both cases, it might also be pointed out that the lengthy form of a cable connecting the correct contacts is also presupposing a certain working geometry. Again, only such an apparatus has the disposition to transmit an electric current between the two contacts. Hence, various combinations of these aspects are needed to sufficiently describe a working principle. Each aspect requires a specific pattern for representing the knowledge involved. In the following, we will discuss the means to represent disposition patterns, which will be presented to represent these three aspects and their combinations (Section 4).

3. Representing Dispositions

The Basic Formal Ontology (BFO) is the top-level ontology that has been adopted by the Open Biological and Biomedical Ontology (OBO) Foundry ([18]; obofoundry.org). BFO contains *BFO:disposition* [BFO:0000016] as a subclass of *BFO:realizable entity* [BFO:0000017], which is, in turn, a subclass of *BFO:specifically dependent continuant* [BFO:0000020]. A sibling class of *BFO:disposition* is *BFO:role* [BFO:0000023]. In contrast to roles, dispositions are described by BFO as realizable entities that are 'internally grounded', i.e., grounded in the physical structure of their bearers [19]. This implies that gaining and losing a disposition requires a physical change on the side of the bearers. Realizable entities are connected via the relation **has realization** with types of processes.[2]

To improve the representation of dispositions in earlier versions of BFO, Röhl and Jansen [7] suggested a pattern for the representation of non-probabilistic single-track dispositions.[3] Inspired by the philosophical literature on dispositions, they use four relations. In addition to *inheres_in*, *has_realization,* and *has_participant*, which all have counterparts in BFO, they introduce the relation *base_of*, which relates a disposition to the quality that grounds that disposition, as well as two versions for the trigger relation (Figure 1).

[2] *BFO:function* [BFO:0000034] is the only asserted subclass of *BFO:disposition.* This subsumption is questioned in [20] and has been defended in [21]. As we do not discuss the ontological representation of the functional analysis in this paper, we can stay undecided on this matter here.

[3] Multi-track dispositions have since been discussed by [22] and [23] as special types of mereologically complex dispositions.

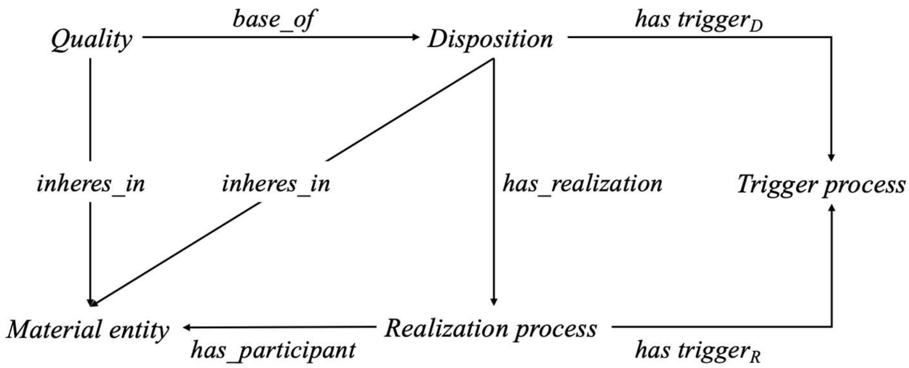

Figure 1: The relations between the classes *disposition*, *material entity*, *quality*, and *realization* and *trigger* processes in the Röhl–Jansen pattern (our diagram based on [7]).

Arp et al. acknowledge this idea of a physical base. They write that a disposition "exists because of certain features of the physical makeup of the independent continuant that is its bearer", and continue that one "can think of the latter as the material basis of the disposition in question" ([18], p. 178). Some have identified the material basis mentioned by BFO with what philosophers have called the categorical basis of a disposition, i.e., as some "characteristics" or "some attribute or attributes of their material bearers that is non-dispositional in nature" [24]. The implementation in the current version of BFO, in contrast, follows a different approach. It contains a relation **BFO:has material basis,** but its range is *BFO:material entity.* As Toyoshima and Barton explain, "The material basis of a disposition is some material part(s) of the disposition bearer in virtue of which the disposition exists." [25]. This approach is obviously motivated by examples like the receptor molecule as part of a cell membrane, or the disordered body structure that bases some disease. It is, however, not viable for all cases. The reason why a ball can roll is not a certain material part of it, but its shape; the reason why a snowman melts is its material. As engineers will have to deal with such cases, too, we will provide patterns for representing such cases in this paper.

BFO documentation also acknowledges the idea of dispositions being triggered. In fact, the ISO document describing the BFO describes "processes triggering dispositions" as the basal causal relation ([26], p. 11). Röhl and Jansen [7] introduce two versions of the trigger relation: $has\_trigger_D$ (relating a disposition type with a trigger type) and $has\_trigger_R$ (relating a realisation type with a trigger type). Originally, the Röhl–Jansen pattern is based on the now-deprecated OBO format, where relations directly connect classes. They define the two trigger relations as follows:

$$D\ has\_trigger_D\ T := \forall x\ (x\ \mathbf{instance\_of}\ D \rightarrow \forall y\ (x\ \mathbf{has\_trigger_D}\ y \rightarrow y\ \mathbf{instance\_of}\ T))$$

$$R\ has\_trigger_R\ T := \forall x\ (x\ \mathbf{instance\_of}\ R \rightarrow \exists y\ y\ \mathbf{instance\_of}\ T\ \&\ x\ \mathbf{has\_trigger_R}\ y)$$

In contrast, the syntax of the Web Ontology Language (OWL) demands a quantifier (e.g., SOME or ONLY) before the second-class term. Note that **BFO:has realization** and **has trigger** may not be used with the SOME quantifier, but only with the quantifier ONLY, because there will probably be instances of the respective type of disposition that are not triggered and thus not realised.[4]

The logical properties of these relations are not very informative, mainly due to the fact that their domain and range are disjoint. Therefore, they are all trivially irreflexive and asymmetric. Also, they are all trivially transitive, as the antecedents condition for transitivity (i.e., Rab & Rbc) is always wrong, and thus the criterial condition for transitivity (i.e., Rab & Rbc → Rac) is always true, although there are no transitive cases at all.

4. Methods

The ontology design patterns suggested here aim to comply with the Open Biological and Biomedical (OBO) Foundry practice rules (obofoundry.org). As suggested by the OBO Foundry, the Basic Formal Ontology (BFO) is adopted as a foundational ontology, providing both top-level classes and formal-ontological relations. Two ways are employed to represent the Pahl–Beitz framework for working principles. First, we will introduce new classes and relations according to the OBO Foundry rules, or import these from other OBO Foundry ontologies. This implies in particular that all new classes are to be related to a BFO top-level class. In general, we do not cover the domain classes that will be needed for the representation of working principles here, but use placeholders for these.

While the introduction of new classes even in large numbers is a welcome way to enrich an ontological representation, new relations should be kept to a minimum and instead be represented by means of classes plus the relations available in BFO and the OBO Foundry ontologies. The only new relation introduced will be the relation **has trigger** and its inverse, **trigger of**.

The second way, which we take for working principles here, is to describe Ontology Design Patterns (ODP). According to Gangemi and Presutti [28], ontology design patterns are standardised solutions for recurring representation challenges. They go beyond the introduction of new classes, as they normally involve several classes and typical relations between them, and are thus presented in a number of formal statements that have to be used together. In principle, these statements could be merged into a complex conjunctive statement, but for ease of implementation, we split up this conjunctive statement into less complex statements that can be added to an OWL file.

5. Patterns for Representing Knowledge about Working Principles

In the following, patterns are suggested for representing knowledge about working principles. As we said before (Section 2), stating a working principle may involve detailing materials, shapes, sizes, and working movements. We start with patterns for those cases, where information about one of these aspects is sufficient to describe the working principle, and then proceed to more complex cases where we have to combine various of all three of these aspects. This order is primarily due to didactic reasons, starting from the simple and proceeding to the

---

[4] A trigger relation is in fact included in the Unified Formal Ontology (UFO), where the domain of the relation is the class of situations (nemo-ufes.github.io/gufo, [27]). This is, however, not of much help for users of BFO, as BFO does not acknowledge situations as a kind of entities.

complex. Once the complex patterns are known, the simple patterns can be derived from the complex ones. Each pattern consists of a number of assertions to be included in a respective ontology. These assertions are presented in the form of schemata, including terms containing variables like “*x*”, “*y*”, “*z*”, “*R*”, or “*T*” to form placeholders for classes of the respective types. We implemented the patterns in the OWL using the ontology editor Protégé, and evaluated the patterns using competency questions (see Section 6).[5]

We start with a list of **general declarations** to introduce our placeholder classes that are used in the following patterns and connect them to their respective top-level classes:

portion of material **subClassOf** *BFO:material entity*

portion of material x **subClassOf** *portion of material*

process type R **subClassOf** *BFO:process*

process type T **subClassOf** *BFO:process*

disposition to R **subClassOf** *BFO:disposition*

disposition to R **BFO:has realization** ONLY *process type R*

disposition to R when T **subClassOf** *BFO:disposition*

disposition to R when T **BFO:has realization** ONLY *process type R*

PATO:shape **subClassOf** *BFO*:*quality*

shape y **subClassOf** *PATO:shape*

PATO:size **subClassOf** *BFO*:*quality*

size z **subClassOf** *PATO:size*

If one states a **working material**, this should be based on the knowledge that this material comes with the disposition searched for. As Pahl et al. write, “we need a general idea of the *type of material* [...], for example, whether it is solid, liquid or gaseous; rigid or flexible; elastic or plastic; stiff, hard or tough; or corrosion-resistant” ([2],p. 40, italics in the original). A challenge is posed by the normal way to talk about materials, because they are normally referred to using so-called mass nouns. However, BFO does not contain any top-level category for stuff, and the OBO Foundry naming conventions explicitly state a preference for singular nominal forms ([29], table 1, entry 2.3). To solve this problem, we use the prefix “portion of” to turn a mass noun into a count noun in the singular, e.g., “copper” and “water” into “portion of copper” and “portion of water”.[6] We introduce a class *portion of matter* to collect these classes, which we introduce as a subclass of *BFO:material entity.* Alternatively, the material copper could be represented as an aggregate of copper atoms, but most materials in the real world are not pure aggregates of one kind of atom or molecule only; they will most certainly also contain some

[5] The OWL files for the patterns are available at https://github.com/BiomimeticsOntologies/WorkingPrinciple/.

[6] Regarding the class label, the prefix policy is followed by *SDP:portion of organism substance* [SPD:0000008]. In contrast, neither *ENVO:environmental material* [ENVO:00010483] nor its subclasses comply with the naming convention to use the prefix “portion” in their labels, yet it does describe its instances as portions of materials.

impurities. Hence, while a piece of copper does not need to contain only atoms that are copper atoms, a piece of copper needs to contain at least some copper atoms. In contrast, introducing stuff qualities like *copper quality* and then defining *piece of copper* as a piece of copper in which an instance of the copper quality inheres does not seem to bring with it any surplus for expressivity or reasoning power.

The following pattern can be used in such a situation; all six subclauses have to be asserted in the respective ontology:

(Wmat) *portion of material x* **subClassOf BFO:bearer of** SOME *disposition to R*

If one states a **working geometry** (e.g., a shape or surface structure), this should be based on the knowledge that material objects of this shape have the disposition searched for. According to Pahl et al., the working geometry, i.e., "the arrangement of working surfaces (or working spaces)" ([2], p. 39), can be described with reference to its dimensionality (point, line, body, etc.), its shape (curve, circle, octagon, asymmetrical, etc.), its position (axial, parallel, vertical, etc.), its size (small, large, tall, low, etc.), and its number (undivided, divided, simple, double, etc.). For lack of better alternatives, we decided to use the respective Phenotype and Trait Ontology (PATO; [30]) classes for shape and size, namely *PATO:shape* [PATO:0000052] and *PATO:size* [PATO:0000117], which are imported with their subclasses under *BFO:quality*. While both PATO and Pahl et al. [2] use relative sizes, the biological perspective of PATO (increased, decreased, normal) does not easily fit to the technical perspective of Pahl et al. (small, large). In addition, in the technical domain it will often be important to represent absolute sizes (measured in meters or some other unit). The dimensionality is taken care of by the respective subclasses of *PATO:shape*. What Pahl et al. [2] call "number" seems to be a mixed bag of shape qualifiers (a divided sphere is a specific shape) and aggregate builders (two spheres are an object aggregate consisting of two parts with a specific shape). The following patterns can be used for these:

(Wshape) *object of shape y* **equivalentClass**
(*BFO:material entity* AND **BFO:bearer of** SOME *shape y*)

object of shape y **subClassOf BFO:bearer of** SOME *disposition to R*

(Wsize) *object of size z* **equivalentClass**
(*BFO:material entity* AND **BFO:bearer of** SOME *size z*)

object of size z **subClassOf BFO:bearer of** SOME *disposition to R*

These patterns may be used in combination with each other, if, e.g., a certain working principle requires combining a certain material with a certain shape, in a certain size. This would require using the following combined pattern:

(Wcomb1) *object with xyz* **equivalentClass** (*portion of material x* AND
BFO:bearer of SOME *shape y* AND **BFO:bearer of** SOME *size z*)

object with xyz **subClassOf BFO:bearer of** SOME *disposition to R*

Alternatively, without using the defined class *object with xyz*, the pattern can be laid out in the following condensed way:

(Wcomb1*) (*portion of material x* AND **BFO:bearer of** SOME *shape y*
AND **BFO:bearer of** SOME *size z*)
subClassOf BFO:bearer of SOME *disposition to R*

The third and last aspect is the **working movement** (also called "working motion"). Working movements are, of course, instances of *BFO:process*. According to Pahl et al. ([2], fig. 3.18, p. 95), they can be characterized according to their type (translation, rotation, etc.), their 'nature' (regular–irregular, etc.), their direction (stated by means of the three Cartesian dimensions), their magnitude (velocity, etc.), and their number (one, several, etc.). According to our dispositional account, specifying a working movement means specifying a certain type of cause that will yield a certain intended effect. In our dispositional framework, working movements are best seen as types of triggers for dispositions that are realized (if they are realized) in the intended effects. We here present two patterns for this, which will be compared in Section 6:

(Wmov) *disposition to R when T* **subClassOf has trigger** ONLY *process type T*

(Wmov*) *process type T* **subClassOf trigger of** ONLY *disposition to R when T*

Again, there might be a need to combine these patterns with the other ones. If, for example, a certain piece of material with a certain shape and size needs to be moved in a certain way, we can build either on (Wmov) or (Wmov*). Building on (Wmov), we arrive at the following combined pattern:

(Wcomb2) *process type T* **subClassOf BFO:has participant** SOME
(*portion of material x* AND **BFO:bearer of** SOME *shape y* AND
BFO:bearer of SOME *size z*)

disposition to R when T **subClassOf has trigger** ONLY *process type T*

For an analogue pattern building on (Wmov*), we arrive at the following:

(Wcomb2*) *process type T* **subClassOf BFO:has participant** SOME
(*portion of material x* AND **BFO:bearer of** SOME
shape y AND **BFO:bearer of** SOME *size z*)

process type T **subClassOf trigger of** ONLY *disposition to R when T*

6. Evaluation

We evaluated the patterns using the competency questions displayed in Table 1. Entering (CQ1) and (CQ2) as DL queries returned the correct result, thus validating (Wmat), (Wshape), and (Wsize). (CQ3) delivered the intended results when used with (Wmov*). However, (CQ3) failed to return any results when used together with (Wmov), because the logical strength of the ONLY assertions is too weak. The (Wmov) axiom only says that if there is a trigger process, then it is of type *T*. This does not imply the converse, namely that, given an instance of *T*, that it does actually trigger a disposition being realised in a process of type *R*.

Given the result of the evaluation by means of these competency questions, it would thus be advisable to prefer (Wmov*) over (Wmov). There is, however, a serious problem with (Wmov*): It requires that disposition classes have unique trigger classes. It is, however, likely that there will be many cases where instances of one and the same trigger class can trigger instances of different classes of dispositions. This is a serious limitation of (Wmov*).

Table 1: Competency questions used to evaluate the ontology design patterns

	Natural language question	DL query (Text in the `Consolas font` can be pasted into the DL Query tab in Protégé)
CQ1	Which materials have the disposition to R?	*portion of material* AND **BFO:bearer of** SOME *disposition to R* `portion_of_material and 'bearer of' some disposition_to_R`
CQ2	Which material entities have the disposition to R?	*BFO:material entity* AND **BFO:bearer of** SOME *disposition to R* `'material entity' and ('bearer of' some disposition_to_R)`
CQ3	Which motion leads to R?	*BFO:process* AND **trigger of** ONLY (*BFO:disposition* AND **has realization** ONLY *process type R*) `process and (trigger_of only (disposition and ('has realization' only process_type_R)))`

7. The Limits of the Trigger Relation

The patterns introduced can be used to represent knowledge about the so-called working principles. They make a special relation that relates a disposition to its so-called categorical basis redundant. Hence, there is no need for the *base_of* relation suggested by Röhl and Jansen [7]. However, the pattern for the representation of a working movement (Wmov) requires the introduction of new relations, namely, **has trigger** and its inverse **trigger of**. The trigger is what directly, in an unmediated way, brings about the realisation of the disposition; it is the proximate cause. E.g., the breakability of the wine glass is realised, because a considerable force was exerted on it when it hit the floor. This is, in turn, being caused by other processes – the glass falling, Jim's dropping the glass, Jim's being clumsy, and so on, reaching back, possibly, till the Big Bang. These are, however, indirect, mediated, and non-proximate causes. The two relations can then be elucidated as follows:

- A disposition instance *d* is triggered by a process instance *p*, if and only if *p* is the proximate cause of the occurrence of the realization of *d*.
- A disposition type *D* has a trigger type *T* if and only if instances of *T* are the proximate causes of the occurrence of the realization of the instances of *D*.
- A process type *T* is the trigger type of a certain disposition type *D*, if and only if instances of *D* are triggered only by instances of *T*.

The definition from Röhl and Jansen [7] can be adopted to elucidate the OWL relations:

D **has trigger** ONLY *T* := ∀x (x instance of *D* ⟶ ∀y (x has trigger y ⟶ y instance of *T*))

T **trigger of** ONLY *D* := ∀x (x instance of *T* ⟶ ∀y (x trigger of y ⟶ y instance of *D*))

So far, the introduction of **has trigger** and **trigger of** is unproblematic. The fact that they are standardly to be used with the ONLY quantifier restricts their power in reasoning. Moreover, it can be difficult to determine informative classes of trigger processes for which this relation can be stated. If too large classes are chosen, the resulting statement might no longer be informative. Consider the following sample statements:

(Trig)	a.	*disposition type 1* **subClassOf has trigger** ONLY *BFO:process*
	b.	*disposition type 1* **subClassOf has trigger** SOME *process type 1*
	c.	*disposition type 1* **subClassOf has trigger** ONLY *process type 1*
	d.	*process type 1* **subClassOf trigger of** ONLY *BFO:disposition*
	e.	*BFO:process* **subClassOf trigger of** ONLY *disposition type 1*
	f.	*process type 1* **subClassOf trigger of** ONLY *disposition type 1*

Of these, statement (Trig.a) is trivially true, but quite uninformative. It would be apt for a top-level ontology, but not for a domain ontology, for example, for engineering. In contrast, (Trig.b) is probably false. For it states that, for all instances *x* of *disposition type 1*, there is at least one instance *y* of *process type 1*, such that *x* **has trigger** *y*. As already mentioned in Section 2, it is most likely that some instances of *disposition type 1* are not and might never be triggered. Hence, a statement of the form (Trig.c) is most appropriate to inform about the trigger class of a given disposition type. It states: For all instances x of *disposition type 1*, if x is triggered, then it is only triggered by an instance y of *process type 1*. The developer of a domain ontology using this pattern should thus search for the smallest complete class of triggers, i.e., the process class such that all possible trigger processes of the dispositions type in question fall into this process type.

For **trigger of**, the problems are even worse. (Trig.d) is trivially true for any process type, and therefore uninformative. In contrast, (Trig.e) is false, as processes of different types can trigger a wide range of dispositions of various types. For the same reason, also (Trig.f) is most certainly false, for even instances of one and the same process type are likely to trigger various types of dispositions. A more adequate representation would thus be to relate the trigger class to a (possibly long) alternation of disposition types (e.g., *disposition type 1* OR *disposition type 2* OR *disposition type 3,* etc.).

8. Conclusion

In this paper, patterns have been presented for the representation of knowledge about working principles in a BFO-conformant way. For this, patterns have been introduced that allow for representing the various varieties of knowledge about working principles distinguished by Pahl and Beitz: working materials, working shapes, working size, and working movement. It has not been necessary to introduce a new relation for representing the material basis of dispositions. Only the representation of working movements required a new relation, namely the relation **has trigger**, along with its inverse **trigger of**.

Pahl and Beitz have been our guides in the development of the patterns, and we derived our requirements from their framework. Their framework for engineering design is, however, heavily oriented on mechanical engineering. Possibly, other areas of engineering require different ways to describe their working principles. In this case, our collection of patterns might need to be extended.

The patterns still need to be tested against actual data, which might come with more and different search requirements. However, even if the present CQs are not data-driven, they have already pointed to a serious restriction of their performance. Another serious restriction is that, so far, we only have a core structure for the patterns. In order to be useful, these patterns have to be combined with knowledge about materials, shapes, sizes, and processes, which requires specific and sufficiently detailed reference ontologies for these domains. For shapes and sizes, PATO contains a considerable branch that might be helpful to represent knowledge about working geometries, but these classes have been developed to meet the needs of biology, not of engineering. This is especially urgent for *PATO:size*, as its classes describe relative sizes ("increased", "decreased"), and not the specifications necessary for technical parts. In general, these domain ontologies are not yet developed, or not yet sufficiently developed, and require more work in the future.

Acknowledgements

Research for this paper has been conducted under the auspices of the project "Learning from Nature" funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) (nr. 492191929). Part of the work of Ludger Jansen on this paper has been conducted during a research stay funded by the DFG (nr. 449836922) at the Centre for Advanced Study "Access to cultural goods in digital change: art historical, curatorial, and ethical aspects" (KFG 33) at the University of Münster, and has benefited much from the hospitality and discussions in Münster. We thank the anonymous reviewers for their inspiring feedback.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] G. Pahl, W. Beitz, Konstruktionslehre. Handbuch für Studium und Praxis. Berlin, Heidelberg, New York: Springer, 1977.

[2] G. Pahl, W. Beitz, J. Feldhusen J, K.H. Grote, Engineering Design: A Systematic Approach. 3rd. ed. London: Springer London, 2007. doi:10.1007/978-1-84628-319-2.

[3] B. Bender, K. Gericke (Eds), Pahl/Beitz Konstruktionslehre. Methoden und Anwendung erfolgreicher Produktentwicklung. 9th. ed., Berlin, Heidelberg: Springer Vieweg, 2021.

[4] N. Cartwright, Nature's Capacities and Their Measurement. Oxford University Press, 1994. doi:10.1093/0198235070.001.0001.

[5] J.D. Jacobs (ed), Causal Powers, Oxford, New York: Oxford University Press, 2017.

[6] S. Mumford, R.L. Anjum, Getting Causes from Powers, Oxford University Press, 2011. doi:10.1093/acprof:oso/9780199695614.001.0001.

[7] J. Röhl, L. Jansen, Representing dispositions, Journal of Biomedical Semantics, 2011; 2: S4. doi:10.1186/2041-1480-2-S4-S4
[8] VDI 2221, Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte, Beuth, Berlin, 1993.
[9] VDI 2221-1, Entwicklung technischer Produkte und Systeme: Modell der Produktentwicklung, Beuth, Berlin, 2019.
[10] VDI 2222-2, Konstruktionsmethodik; Erstellung und Anwendung von Konstruktionskatalogen. Design engineering methodics; setting up and use of design catalogues, Beuth, Berlin, 1982.
[11] VDI 6220-1, Bionik: Konzeption und Strategie. Abgrenzung zwischen bionischen und konventionellen Verfahren/Produkten. Biomimetics: Conception and strategy, Differences between biomimetic and conventional methods/products, Beuth, Berlin, 2012.
[12] L. Jansen, Von der bildlichen Darstellung zur ontologischen Analyse: Wirkprinzipien als komplexe Dispositionen, in: M. te Vrugt, O. R. Scholz (eds) Wissenschaft und Metaphysik. Nomos, forthcoming.
[13] K. Gericke, C. Beinke, P. Gemmer, L. Blessing, Entwicklungsmethodik nach Pahl und Beitz und Design Thinking – Vergleich und Einordnung, in: DFX 2010: Proceedings of the 21st Symposium on Design for X, Buchholz/Hamburg, Germany, 23-24.09.2010, 2010; pp.117–130.
[14] R. Koller, N. Kastrup, Prinziplösungen zur Konstruktion technischer Produkte, Berlin, Heidelberg: Springer, 1998. doi:10.1007/978-3-642-58755-9.
[15] K. Roth, Konstruieren mit Konstruktionskatalogen. Band. 1: Konstruktionslehre, 3rd. ed., Springer-Verlag, 2000; Band. 2: Konstruktionskataloge, Springer-Verlag, 2012; Band. 3: Verbindungen und Verschlüsse, Lösungsfindung. Springer-Verlag, 2013.
[16] M. Drack, M. Limpinsel, G. De Bruyn, J. H. Nebelsick, O. Betz, Towards a theoretical clarification of biomimetics using conceptual tools from engineering design, Bioinspiration & Biomimetics 2018; 13: 016007. doi:10.1088/1748-3190/aa967c.
[17] W. Bochtler, Modellbasierte Methodik für eine integrierte Konstruktion und Arbeitsplanung, Aachen: Shaker Verlag, 1996.
[18] R. Arp, B. Smith, A.D. Spear, Building Ontologies with Basic Formal Ontology, The MIT Press, 2015. doi:10.7551/mitpress/9780262527811.001.0001.
[19] M. Rabenberg, C. Benson, F. Donato, Y. He, A. Huffman, S. Babcock, J. Beverley, Grounding realizable entities, in: 15th International Conference on Biological and Biomedical Ontology, July 17–19, 2024, Enschede, The Netherlands. ceur-ws.org/Vol-3939/paper3.pdf.
[20] L. Jansen, Functions, malfunctioning, and negative causation, in: A. Christian, D. Hommen, N. Retzlaff, G. Schurz (eds.) Philosophy of Science: Between the Natural Sciences, the Social Sciences, and the Humanities. Cham: Springer International Publishing, pp. 117–135.
[21] A.D. Spear, W. Ceusters, B. Smith, Functions in Basic Formal Ontology, Applied Ontology 2016; 11: 103–128. doi:10.3233/ao-160164.
[22] A. Barton, L. Jansen, A modelling pattern for multi-track dispositions for life-science ontologies, in: ODLS 2016. Ontologies and Data in Life Sciences. Proceedings of the 7th Workshop of the GI Workgroup Ontologies in Biomedicine and Life Sciences (OBML), Halle (Saale), Germany, September 29-30, 2016. ceur-ws.org, Aachen 2016, H.1–2. ceur-ws.org/Vol-1692/abstractH.pdf.
[23] A. Barton, L. Jansen, J-F. Ethier, A taxonomy of disposition-parthood, in: S. Borgo, O. Kutz et al. (eds.), Proceedings of the Joint Ontology Workshops 2017 (JOWO 2017), Bozen–

Bolzano, Italy, September 21–23, 2017, Aachen 2018, ceur-ws.org/Vol-2050/FOUST_paper_10.pdf.
[24] J. Hastings, S. Cox, R. West, C. Notley, Addiction ontology: applying basic formal ontology in the addiction domain, Qeios, 2020. doi:10.32388/HZHJIP.
[25] F. Toyoshima, A. Barton, Two approaches to the identity of processes in BFO, in: FOUST VII: 7th Workshop on Foundational Ontology, 9th Joint Ontology Workshops (JOWO 2023), co-located with FOIS 2023, 19-20 July, 2023, Sherbrooke, Québec, Canada. ceur-ws.org/Vol-3637/paper33.pdf.
[26] ISO/IEC 21838-2:2021, Information technology — Top-level ontologies (TLO) Part 2: Basic Formal Ontology (BFO), 2021.
[27] R. Baratella, M. Fumagalli, Í. Oliveira, G. Guizzardi, Understanding and modeling prevention, in: R. Guizzardi, J. Ralyté, X. Franch (eds), Research Challenges in Information Science: 16th International Conference, Research Challenges in Information Science. RCIS 2022. Lecture Notes in Business Information Processing, Springer, vol 446, pp. 389–405. doi:10.1007/978-3-031-05760-1_23.
[28] A. Gangemi, V. Presutti, Ontology Design Patterns, in: S. Staab, R. Studer (eds.) Handbook on Ontologies. Berlin, Heidelberg: Springer, pp. 221–243.
[29] D. Schober, B. Smith, S.E. Lewis, W. Kusnierczyk, J. Lomax, C. Mungall, C.F. Taylor, P. Rocca-Serra, S.A. Sansone, Survey-based naming conventions for use in OBO Foundry ontology development, BMC Bioinformatics 2009; 10:125. doi:10.1186/1471-2105-10-125.
[30] S. Köhler, L. Carmody, N. Vasilevsky et al., Expansion of the Human Phenotype Ontology (HPO) knowledge base and resources. Nucleic Acids Research 2019, 47: D1018–D1027. doi:10.1093/nar/gky1105.