Towards GFO 2.0:
Architecture, Modules and Applications

Patryk BUREK[a,1], Frank LOEBE[b], and Heinrich HERRE[c]

[a] *Institute of Computer Science, Faculty of Mathematics, Physics and Computer Science, Marii Curie-Sklodowskiej University, Lublin, Poland*
[b] *Computer Science Institute, University of Leipzig, Germany*
[c] *Institute for Medical Informatics, Statistics and Epidemiology, University of Leipzig, Leipzig, Germany*

Abstract. The General Formal Ontology (GFO) is a top-level ontology that has been developed by the Onto-Med Research Group since the early 2000s. Since that time several new theoretical results have been achieved as well as numerous projects have utilized the ontology, especially in complex domains such as bioinformatics and medical computer science. This leads to the need for an up-to-date overview of GFO and access to its applications.

This paper represents the first step towards introducing the GFO 2.0 framework, which aims at the integration of the work that is already present, but scattered in various publications, and its provision as a ready-to-use and reusable framework. For this purpose we summarize key features of GFO so far, outline a novel modular architecture and survey first modules for GFO 2.0, linking to applications. Finally, a rigorous and systematic development process is indicated.

Keywords. top-level ontology, GFO, module

1. Introduction

The General Formal Ontology (GFO) is a top-level ontology that originated as a component from the broader GOL project (General Ontological Language), commenced in 1999 by the Onto-Med Research Group [1] at the University of Leipzig. Version 1.0 of the GFO specification was released in 2006 [2], followed by a minor revision 1.0.1 (draft) [3] in 2007. Since that time much new theoretical research on GFO has been pursued and presented as well as numerous applications have been realized.

The GFO is a broad conceptual framework partially formalized in first order logic (FOL) and serialized in the Web Ontology Language (OWL) [4]. The projects in which GFO was or is utilized, especially in complex domains such as bioinformatics and medical computer science, cf. e.g. [5-6], have demonstrated a wide spectrum of applications, including ontological and conceptual modeling, the development of domain ontologies and ontology design patterns as well as of Unified Modeling Language (UML) profiles [7].

---

[1] Corresponding Author, University address: pl. Marii Curie-Sklodowskiej 5, 20-031 Lublin, Poland; E-mail: patryk.burek@poczta.umcs.lublin.pl .

On the other hand, we recognize the need for a consistent and systematic overview of all new results. This relates further to the aim of providing GFO as a ready-to-use and reusable framework that is suitable for further applications in both research and industrial projects. Currently especially the readiness for use is hampered by the fact that many results remain hidden in discussion papers and reusable artifacts are missing.

Therefore, we aim at taking next steps. The actual artifacts of GFO, comprising of ontology files and/or UML profiles and accompanying material (guides/manuals, etc.), need to be integrated with new theoretical findings as well as with the results obtained during the application of the framework.

The current paper represents the first step towards introducing the GFO 2.0 framework. With the aim of eventually integrating all of the work, here we survey and develop a kind of a map of the existing bits and pieces. We expect such a collection to be already useful for future applications as well as for the further development of GFO itself, first and foremost as a new starting point towards releasing GFO 2.0 in a production-ready form. This release will cover new results, yet it will likewise comprise additional artifacts necessary for applying GFO in projects. Hence, our objective is not only to establish and elaborate theoretical foundations, but also to make them and GFO as a whole much more easily available for practitioners. Enhancing the development process with more rigor than during the last years, we will rely on an iterative and modular approach, as successfully adopted in software engineering. Accordingly, GFO should less be understood as a single, monolithic ontology. Instead, we will outline a conceptual framework and an ontology, the modules of which will form a coherent whole, on the one hand, but can be used flexibly and selectively to the extent required, on the other hand. Altogether, the current paper is a primer preparing the roadmap for the GFO 2.0 project.

The remainder of the paper is organized as follows. In the next section we give a short overview on key features and ontological choices of GFO and relate that with further state-of-the-art top-level ontologies. In section 3 we discuss the main architectural principles underlying the GFO 2.0 release, including the adopted modular approach to ontology development. Section 4 provides an overview of several main modules of GFO and serves therefore as a foundation for the roadmap of the whole project. Finally, section 5 concludes the paper with a summary and future work, in particular an outline of the roadmap and the development process.

2. State of the Art

An ontology is to be understood as a formalized system of categories and relations that describe knowledge about a domain of reality. Top-level or foundational ontologies contain those categories that can be applied to a very broad spectrum of domains. Nowadays, top-level ontologies (TLOs) play various roles: They serve as a framework for organizing domain-specific knowledge, in many cases of complex fields; they can be applied to the ontological analysis and the foundation of areas of knowledge about a domain; and they can be used to create new theories. Furthermore, TLOs provide a method to transcend the conceptual boundaries of a field of research. Meanwhile efforts are under way to offer TLOs as standards, not at least in order to provide more stability and reliability when referring to them or using them. In particular, the recent standard ISO 21838 [8] is devoted to defining requirements for TLOs.

2.1. Ontological Choices in GFO

In general, GFO is intended to become a scientific theory which covers empirical areas, such as physics, chemistry and biology, and likewise non-empirical domains, such as mathematics and set theory, but also the humanities. Subsequently, we summarize six special features that constitute the hallmarks of GFO.

(1) *Ontological regions and levels of reality.* In GFO, the world is subdivided into four ontological regions: the material region, associated to the natural sciences; the psychological region, which is related to the phenomena of the mind, such as intentionality; the ontological region of socio-systemic entities, including communication between subjects, social phenomena like social roles, economy, the role of labor for creating artifacts, and the inter-relation between man and nature. Finally, GFO covers the ontological region of ideas and abstract entities.

The conception of ontological regions is partially borrowed from the philosophy of N. Hartmann [9], cf. also [10], though there are differences, in particular in treating set theory as a particular ontological sub-region of the abstract region. Pursuing an integrative approach, we argue that important ideas of influential figures of history cannot be neglected,[2] such that a broad classification of modes of being is necessary to achieve a comprehensive picture of the world.

(2) *Ontology of the material region.* One main contribution of GFO in this field consists of a new theory of space and time [12], based on ideas of Franz Brentano [13]. Another contribution is the postulation of a basic law, called integration axiom. It says that for any material object there exists a process that corresponds to the object in a particular way. The axiom utilizes a new type of individuals, called presentials. Moreover, it is the fundament for a new classification of properties of spatio-temporal material entities [14] and it allows for a new interpretation of the particle-wave duality in quantum mechanics.

(3) *Ontology of categories and the multi-categorial approach.* GFO admits various kinds of categories, which are classified into universals (Aristotelian, Platonic), concepts, and symbolic structures. The individual-category dichotomy is a basic feature of formal ontology, however, it must be refined by taking the various kinds of categories into consideration. None of the mentioned types of categories can be neglected. Otherwise important phenomena related to the meaning of terms and to the communication between humans cannot be captured. Sets are included among the categories, yet they play a particular role, because they are based on the membership-relation, which differs from instantiation. An important feature of GFO is the introduction of types of higher order for concepts and for sets. This yields a powerful mechanism to represent concepts the instances of which are themselves concepts.

(4) *The principle of integrative realism.* GFO postulates the existence of a reality that is independent of the mind. Integrative realism is determined by two features. First, the subject has access to real and independent entities of the world only through concepts, being a part of the mind. Secondly, there is a law-like correspondence between the subjective phenomena of the mind and the independent reality. This approach leads to a new understanding of the relation between ontology and epistemology, because both cannot be separated. Ontology is directed at the independent objects of the world, though, these appear to the subject through the knowledge (involving perception and concepts)

---

[2] For example, Kurt Gödel defends a rigorous mathematical Platonism, where mathematical entities such as numbers and sets have an objective existence that is independent of the mind and of the material world [11].

that the subject has about them. This leads, we believe, to a new interpretation of works by G. Frege [15], B. Russell [16], and K. Twardowski [17]. Further, GFO pursues a logical approach to the denotation problem. One basic idea is that an entity of reality is reflected as a (logical) individual within a model-structure, which models a part of reality. The description of this individual can then be given against background knowledge in the form of a formal logical theory.

(5) *The onto-axiomatic method*. This newly established method combines formal ontology with the axiomatic method and the model theory of mathematical logic, with A. Tarski's work as an important pillar [18]. We postulate that the onto-axiomatic method provides an overarching principle for the rational reconstruction of existing theories as well as for the development of new scientific theories. The semantics of formalized theories must be further developed into an ontological semantics, cf. [19]. The application of the method combines and unifies top-down with bottom-up principles.

(6) *Top-level ontologies in an open evolutionary system.* The approach of a coordinated evolution [20] is rejected, if such an approach assumes some overarching coordinating top-level ontology within which ontologies evolve. P. Feyerabend is a strong critic of a similar principle applied to the development of science in general [21]. For GFO, the future development of ontologies is deliberately assumed to be open. This notwithstanding, there are some basic principles about the organisation and structuring of a network of top-level ontologies. Briefly indicated, these ontologies are presented in a formal language and there are various relations connecting them, among which we consider the interpretability relation to be of utmost importance. If one ontology can be interpreted in another ontology, then the latter is at least as expressive as the former. It can be expected that there will never be a single system covering all other systems, since the future is – so to say – open. The interpretability relation creates a partial ordering between ontologies, the investigation of which constitutes a research field of its own.

2.2. Comparison with other Top-Level Ontologies

Within limited space, we relate GFO selectively to some further representatives of top-level ontologies. The process ontologies of J. Seibt [22] and of M. West [23] are 4D-ontologies, where objects are special processes. These ontologies lack the deep duality between objects and processes, and a particular feature of objects disappears, the phenomenon of being wholly present at a time point. Hence, the notion of presential cannot be explicated, because a process cannot be wholly present at a time point. Furthermore, both ontologies do not provide any account of categories.

Turning to pure 3D-ontologies, BFO (Basic Formal Ontology) [24] analyzes the distinction between objects and processes differently. In BFO processes are – so to say – properties of objects, they depend on objects. Furthermore, BFO does not allow for an ontology of categories, in particular concepts are excluded from BFO. Universals in BFO resemble Aristotelian universals, while there is no place for ideal entities, such as platonic universals or sets. The treatment of boundaries in BFO is a relevant contribution. The topic has been adopted and advanced by GFO, up to a complete axiomatization in connection with the analysis of time [12] and solving a problem of touching entities.

DOLCE [25, 26] exhibits some commonalities with GFO, yet also relevant differences. DOLCE contains a classification of properties, which can be understood as categories. But a basic distinction between concepts, universals, and symbolic structures appears to be missing, likewise a full type-system (with concepts of higher order).

DOLCE is another pure 3D-ontology, where we are not aware of an integration axiom like that of GFO. For a more detailed comparative study on DOLCE and GFO, see [14].

The Unified Foundational Ontology (UFO) [27, 28] incorporates early developments from DOLCE, the Ontology of Universals underlying OntoClean [29] and aspects of GFO into a single coherent foundational ontology. UFO is essentially a 3D-Ontology. The notion of type as used in UFO, corresponds to the general notion of category in GFO, though we see no distinction between concepts, universals and symbolic structures. Similarly, sets do not seem to play any relevant role in UFO.

In summary, the top-level ontologies mentioned above do not satisfy all characteristics (1)-(6) of GFO in sect. 2.1. While we believe in a high level of expressiveness of GFO, it remains an interesting project to study the mutual interpretability among the ontologies.

3. Architecture for GFO 2.0

Concerning the architecture of GFO, we note first that very early formalizations already comprised a few hundred formulas in first-order logic. Developing, managing and maintaining such a theory as a monolithic artifact is hardly feasible and can prevent or at least hinder its usage. Since then and to some extent inspired by software engineering, the modularization of ontologies has been studied from several angles, cf. [30, 31], even though there is no universal way to modularize an ontology.

Over the years, GFO has been continuously elaborated, revised and applied. This includes the formation of modules to a certain extent, but in a rather ad-hoc way. In working on GFO 2.0 we pursue an explicitly modular approach with some novel aspects.

3.1. Modules

First of all and at its core, by a *module* we understand basically a theory of some (typically limited) modeling problem or area. Moreover, a module constitutes a coherent, ideally self-contained part of a larger system (in our case, of GFO). At least within that system each module is expected to be extensible in the sense that it is well geared to other modules and can be used in consistent combination with them.

'Module' and 'theory' are here to be understood in a conceptual sense. Insofar a module can also be described as a conceptual framework of the problem or area that it addresses. Accordingly, a module is not language-specific, neither regarding any natural, nor any formal language. Of course, in order to capture and provide a module we need to rely on language(s).

This yields a distinction between, on the one hand, 'module' in the conceptual reading described so far – from now on referred to as the *content of a module* – and, on the other hand, artifacts intended to represent that content – *representational artifacts of a module*. Examples of such artifacts are specifications in formal or semi-formal languages, such as axiomatizations in first-order logic, OWL patterns and UML profiles. Note the semantic shift, in that now the term 'module' incorporates both aspects, a content side and a representational side. For simplicity,[3] we use 'module' to refer to either side or jointly to both sides.

Not each artifact of a module must cover all of its content. It is further rarely the case that a single representation in a given modeling area is the best option in all

---

[3] and as is common for dot types, cf. [32]

application or implementation scenarios. Therefore, a single module may contain many patterns. Similarly, there may be more than one axiomatization, e.g. due to using different formal languages or formalizing the module content by means of the same language in different ways.

In understanding an ontology as a collection of modules in the sense of conceptual frameworks we summarize the following benefits:

(1) The module (its content) is agnostic with respect to a particular formalism or implementation language, i.e., there is no implementation bias. Instead, a module may be provided in the form of multiple implementations for different application contexts.

(2) Modules are intended to fit and function together, but should be designed to be maximally independent from one another. Consequently, they can be used separately, but with built-in extensibility if other modules may need to be included.

(3) While modules as conceptual frameworks appear similar to ontology design patterns, the latter – at least in all cases we are aware of – appear attuned to a certain level of granularity, and some of them have an application-bias. The notion of module as described above responds to such bias through the offer of its content in (potentially) several languages. Moreover, it is intended to cover a whole range of module sizes, from very small modules to possibly large and complex frameworks.

3.2. Further Organization Principles

From an architectural point of view, three aspects account for further dimensions in the organization of GFO. The first draws on the notion of *ontological regions* as discussed in sect. 2.1. Secondly, there are several *top-level distinctions*, i.e., very generic and widely applicable distinctions that re-occur across different ontological regions. Prime examples are "classical" distinctions, such as objects vs processes, or attributes vs their bearers vs facts and situations. Thirdly, *meta-level abstraction* distinguishes ordinary categories from those categories involved in analyzing ontologies themselves, as well as from categories of formal entities (such as sets) that are employed in connection with the formalization of ontologies.

Let us consider those three aspects and their manifestations in the context of GFO and its modular structure more closely, starting with the four major ontological regions. The material region is the one among those four that has been developed furthest within GFO. This applies also with regard to the treatment of top-level distinctions, which has frequently been tested and applied in connection with material entities. Ontological regions may also be viewed as modules, notwithstanding their huge extent as theories. This applies in the sense of the contents of modules, whereas, at the current stage, GFO comprises no representational artifacts that correspond to regions as a whole.

Instead, the modules presented in the next section – for example, (i) categories and individuals, and (ii) attributes – largely result from the study of top-level distinctions and associated kinds of entities. These modules are much more limited in size than ontological regions, for which reason they are more manageable. Furthermore, those distinctions have been analyzed for decades in information sciences and for centuries in philosophy. This has yield many inspirations when establishing these theories.

Finally, certain modules (or fragments of them) dealing with top-level distinctions lend themselves to the analysis of ontologies and their constituents (and the formalization of ontologies). First and foremost this is the case for the theory of categories and individuals. In the present paper we do not elaborate on the self-analytic nature of these modules and their value for meta-level considerations, but we refer to [33] for that.
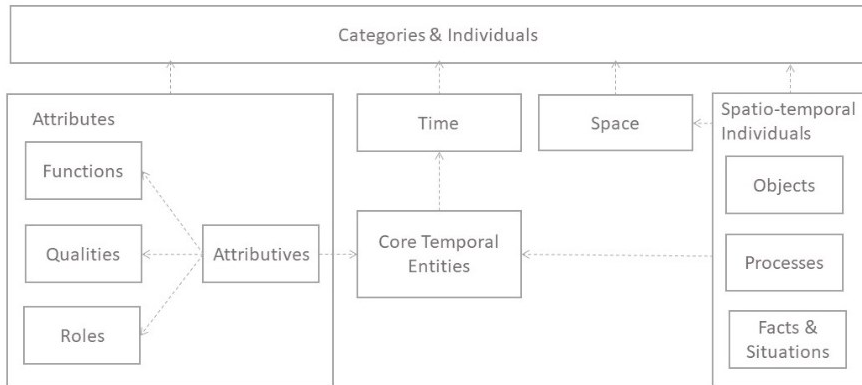
Figure 1. UML-like diagram depicting the modules of the GFO and their dependencies. Each rectangle represents a module with optionally embedded submodules. Dashed directed arrows indicate dependencies.

4. Overview on Selected GFO Modules

The current section provides an overview of the main modules of GFO 2.0. The findings are organized into modules together with applications. A UML-like components diagram in Figure 1 depicts the organization of the modules.

4.1. Categories and Individuals

A basic distinction in formal ontology is that between individuals and categories. Individuals are uniquely determined entities which cannot be instantiated. In contrast, categories are instantiable and can be predicated of other entities. We postulate that this basic distinction holds in any of the ontological regions. Categories themselves are classified into universals (subdivided into platonic and immanent universals), concepts and symbolic structures. [4] Sets are entities that behave in some respect similar to categories, but on the other hand, they differ from them because they cannot be predicated of other entities. Platonic universals are independent of the subject[5] and of the material world, they have an ideal, mind-independent existence. They belong to the ontological region of ideal entities. Typical entities of this kind are mathematical entities, for example as occurring in pure geometry and number theory. Immanent universals (which correspond to Aristotelian universals) are in the real material things; they are independent of the subject, though they depend on the real things. Concepts are in the mind, they are creations of the subject. Since the subject is the bearer of any intellectual activity, concepts play the central role in any process of contemplation, perception, and theory formation. We acquire the world through our concepts and therefore we consider them as the most important kind of categories. Concepts provide the basic means to access any other entity. Insofar they realize an interface between the subject and the

---

[4] This distinction draws partly on the theory of J. Gracia in [34].

[5] The term *subject* has the meaning of *recognizing self* and is synonymously used for the term *mind*. The relation between subject and object is a central topic of the European philosophy since 1500, culminating in the classical German philosophy of the 19th century.

entities outside of and independent of the mind.[6] For any of these kinds of categories (concepts, immanent universals, platonic universals) we introduce a basic relation: an entity e is an instance of the concept C, formally: instanceOf(e, C). We further distinguish the extension and the intension of a category. The extension of a category is defined as the set of its instances, whereas its intension refers to its meaning or content. We assume that the intension of a concept has parts, which we call *categorial parts*.

There remain two further kinds of categories, namely symbolic structures and sets. Symbolic structures have tokens as instances, which are material entities and therefore individuals. Furthermore, for symbolic structures (and their instances) we introduce another basic relation, the relation of denotation with the meaning den(s, S, e) := 'the token s (being an instance of the symbolic structure S) denotes the entity e'. The relation den(X, Y, Z) can be iterated, which may lead to complex denotation systems. Further, we consider sets as a particular kind of category. Sets exhibit a full simple type system, which can similarly be established for concepts. However, sets clearly differ from concepts, a.o. because their basic relation is the membership relation, introduced as 'the entity e is an element of the set S'.

A few examples of axioms follow, regarding the signature of which we explain Int($\cdot$) for intension, int(x, y) for x being the intension of y, ext(y, x) for y being the extension of x, and cpart(x, y) for x being a categorial part of y.

1. $\text{Int}(C) = \{c \mid \text{cpart}(c, C)\}$
2. $\neg \exists x \, (\text{Cat}(x) \wedge \text{Ind}(x))$
3. $\forall x \, (\text{Cat}(x) \rightarrow \exists y \, (\text{Set}(y) \wedge \text{ext}(y, x)))$
4. $\forall x \, y \, (\text{Cat}(x) \wedge \text{Cat}(y) \wedge \text{int}(x, y) \rightarrow \text{ext}(y, x))$

Note that axiom (1) represents a simplified approach, for which we assume in addition that cpart satisfies the conditions of a partial ordering with least upper bound for any two of its elements, as well as that any categorial part contains (or is) an atomic part.

Aspects of this module were applied in the analysis of the notion of core ontology and the development of GFO-Bio [35], a core ontology for biology, as well as in an ontological theory of the notion of data element [36] resulting from the MDR project on a metadata repository for clinical and epidemiological research. The method of search ontologies as introduced and employed in the projects Ontovigilance and OntoPMS [5] relies particularly on the relation between symbolic structures and denoted concepts.

4.2. Time and Space

Space and time are basic categories which account for fundamental assumptions behind those individuals that are said to be in space and time. We introduce two modules *Space* and *Time* in GFO 2.0 for covering notions related to spatial and temporal phenomena.

The module of time has a high impact on information systems and the understanding of time is fundamental for numerous domains, since the representation of a domain typically involves entities related to time, such as events, changes and processes. The significance of the conception and representation of time entities and reasoning about

---

[6] We emphasize that GFO defends a realist philosophy, introduced as *integrative realism*, expounded in some detail in [14]. We reject Kant's artificial separation between *thing as such* and its *appearance*. The main arguments against such a separation were already explicated by Hegel in his *Phenomenology of the Spirit*.

temporal data and knowledge is supported by the presence of the time concept in most top-level ontologies as well as by dedicated time ontologies, e.g. OWL-Time [37].

The Time module consists of two submodules, *Time Core* and *Time Region*, and is already backed by a rich axiomatization in first-order logic. The latter has been analyzed metalogically, proving consistency, completeness and decidability as presented in [12]. The mature formalization of the module goes hand in hand with low ontological commitment, inspired by ideas of Franz Brentano [13]. The Time Core submodule introduces two basic temporal categories, namely intervals (also called chronoids) and time points (aka time boundaries). This basic distinction is a foundation for further modules of GFO, especially for the core temporal module. There is the assumption that the temporal continuum can be introspectively accessed without any metrics, and that it cannot be understood and grasped only in terms of sets of time points. Equally, neither can intervals be simply reduced to the sequence of time points, nor should they be identified with intervals (sets) of real numbers. Central basic relations of the module are temporal part-of, the temporal coincidence of time boundaries, and relations linking time boundaries with chronoids. The Time Region submodule extends Time Core by addressing mereological sums of chronoids, referred to as time regions.

The Space module in turn accounts for basic notions for representing spatial entities, also axiomatized in FOL. The theory is likewise inspired by the ideas of Franz Brentano on space, time and the continuum and it starts from four primitives: the category of space regions, the relations of being a spatial part and being a spatial boundary, as well as the relation of spatial coincidence.

Both modules of Space and Time are fundamental elements of GFO 2.0 and they supply basic notions utilized in further modules of the framework. Especially, Core Time is used in the Core Temporal Entities module and Space in Spatio-Temporal Individuals. Moreover, they are directly relevant in connection with projects on the navigation in surgical interventions, cf. BioPass and COMPASS [38].

4.3. Core Temporal Entities

The module *Core Temporal Entities* is responsible for representing all entities located in time. As such it depends on the Time module and more specifically on the distinction between time intervals and time points introduced there. On that basis we introduce two notions, *presentials* and *time extended entities* (TEEs). The former are immutable entities, which exist and are fully present at exactly one time point, therefore having no temporal extension. Time extended entities exist over a time period and thus can undergo changes. Presentials and TEEs are glued together by the snapshot relation, denoted snapshot(x, y), 'presential x is a snapshot of time extended entity y'. For each presential there exists a time extended entity that that presential is a snapshot of. From a modeling point of view presentials can be interpreted as reified temporal snapshots of TEE.

In its current state the module comes together with the OWL Temporal Entities Pattern [39] which supports the modeling of an interplay between presentials and TEEs as it is needed in common cases where data on presentic observations is aggregated into time extended entities. The pattern is applied in connection with the development of the Cell Tracking Ontology (CTO) [39], there for gluing raw observation data to more complex data structures such as cells and cellular genealogies.

4.4. Spatio-Temporal Entities

Material entities are classified into objects, processes and entities that depend on them. *Objects* occupy space and endure through their lifetime, which means that they are the same through their life-time and "are" wholly present (exhibit a presential) at any time point of their life-time. Processes evolve through time and have a temporal extension.

Subsequently, we collect some exemplary axioms on objects.[7]

5. $\forall x \, (\text{Obj}(x) \rightarrow \exists y \, (\text{Chron}(y) \wedge \text{lifetime}(x, y))$
6. $\forall x \, (\text{Obj}(x) \rightarrow \exists y \, (\text{SReg}(y) \wedge \text{occ}(x, y))$
7. $\forall x \, y \, (\text{Obj}(x) \wedge \text{mpart}(y, x) \rightarrow \exists u \, (\text{occ}(y, u) \wedge \text{SReg}(u)))$
8. $\forall x \, y \, t \, (\text{Obj}(x) \wedge \text{lifetime}(y, x) \wedge \text{tp}(t, y) \rightarrow \exists z \, (\text{exhib}(x, z, t)))$
9. $\forall x \, y \, t \, (\text{exhib}(x, y, t) \rightarrow \text{Pres}(y))$

Processes evolve through time, they cannot be wholly present at time points. Since we restrict considerations in this section to material processes, we stipulate that any process boundary contains a material presentic object as a part. This presentic object possesses object qualities, though further properties can be associated with this object that have their origin in a process, for example, the velocity of a moving object at a time point.

Objects and processes are not isolated from each other, they are integrated in a particular way. This integration is postulated by the integration law, a principal axiom of GFO.

10. $\forall x \, (\text{Obj}(x) \rightarrow \exists y \, (\text{Proc}(y) \wedge \forall z \, t \, (\text{exhib}(x, z, t) \leftrightarrow \text{procbd}(y, t, z)))$
     Integration Law: For any object there is a process such that its process boundaries (procbd) coincide with the presentials exhibited by the object.

Situations are responsible for representing aggregates of objects comprehensible as wholes. In GFO, situations are built upon objects and individualized relations connecting them, as for instance in the case of a cup standing on a table, in which the cup and the table are the objects, stands_on is a relation connecting them and standing_object and supporting_object are the roles of the objects in the context of that relation. All those entities taken together and perceived as a whole is considered as a situation. For every situation *S* there exists an object participating in *S*. If an object as such can be understood as a whole composed of individual qualities connected with their bearer by the inherence relation, then even a single object may be considered as a situation. That interpretation resembles the ontology of Tegtmeier [40], where situations (aka states of affairs) are the basic entities and all other entities (object, processes etc.) are their special cases.

The module is currently equipped with the OWL Situation Pattern which has been applied to the CTO [39]. It enables reconstructing from the raw data of cell tracking experiments the more complex structures called cellular genealogies, which consist of multiple cells linked by relations such as e.g. cell divisions. Other applications of the module are devoted to medical computer science, among them BISON, an ontology of minimally-invasive surgical procedures based on situational information extracted from endoscopic, procedural and sensory data [41].

---

[7] using this signature: Obj(·) for object, Pres(·) – presential, Chron(·) – chronoid, SReg(·) – space region, exhib(x, y, t) – 'object x exhibits the presential y at time point t', lifetime(x, y) – 'x is the lifetime of y', mpart(x, y) – 'x is a material part of the object y', occ(x, y) – 'object x occupies space region y', tp(t, y) – 't is a time point within y'.

4.5. Attributes

The two major generic notions of the *Attributes* module are termed attributives and properties. In contrast to the individuals discussed so far, *attributives* are individuals that characterize other individuals, e.g. the function of a molecule or the color of an apple. Attributives are instances of particular categories that we call *properties*. The module supports modeling both, properties and attributives, and it consists of four submodules. Three of them deal with representing different kinds of properties: qualities, roles and functions. The fourth concerns the assignment of attributives to time related entities.

4.5.1. Qualities

The Qualities submodule covers *qualities* and *quality values*. The former are properties "which are typically expressed in natural and artificial languages by means of syntactic elements such as adjectives/adverbs or attributes/properties, respectively" [2], such as weight, color, speed or temperature. Values in turn are quantities used in measurement, observation or calculation of qualities, e.g. 10kg, green. Often they are scalars or vectors. Both qualities and values are considered as abstracts having no relation to time and being independent from the entities which are characterized by them.

An assignment of a quality and a value to an individual is called an *individual quality*. In case of time-related quality bearers the assignment itself is also of some temporal extent and may involve the change of quality values over time. Therefore, in order to model qualities which change values in time we introduced the OWL Temporal Qualities Pattern which is motivated by the observation that a straightforward approach to modeling qualities in OWL cannot represent the change of quality values adequately [42]. The pattern has been applied to the CTO and it supports consistent modeling of (1) the qualities observed at a single time point, (2) static, non-changing qualities of enduring entities and (3) dynamic, changing qualities of enduring entities [39]. In connection with the Leipzig Health Atlas (LHA) the module forms the basis for specifying complex phenotypes [43].

4.5.2. Roles and Relations

The notion of *roles* is pervasive in many domains, because many categories can be revealed as roles. Roles depend on the contexts that "define" them. This leads to the difficulty that the nature of distinct kinds of roles varies greatly. Loebe [44] distinguishes three major kinds: relational, processual and social roles.

There is a two-fold approach regarding roles in the module structure for GFO 2.0. On the one hand, a very generic module covers the notion of role and neighboring categories and relations with a weak theory. On the other hand, more specific notions of roles are dealt with in connection with other modules, typically in modules where the contexts of a role type are covered.

Relations in GFO are understood as categories of particular entities that "glue their arguments together", where the latter are called *relators*. This differs clearly from formal approaches of viewing relations as mere sets of *n*-tuples of the corresponding arguments. Relations are one of the common kinds of contexts for roles. Indeed, relators are composed of (relational) roles as well as relations are associated with a number of role categories, cf. [45] for more details on this theory.

4.5.3. Functions

The Functions module is responsible for representing functions as commonly used for characterizing entities across many domains, including engineering (e.g. the function of an engine), the natural sciences (e.g. the function of blood cells) and the social sciences (e.g. the function of a manager). Despite many works on functions over the last decades, consensus on a single reading of the term has not been reached. For GFO we rely on a popular account of function formulated in terms of the notion of role, where a *function* is seen as a category that captures a role in the context of some goal achievement.

The Functions module provides patterns for representing function specifications as well as for modeling interdependencies between functions, e.g. function decomposition. Attributive assignments from the Attributes module support the modeling of the assignment of a function to an entity, in the sense of an entity having a function.

The specification of a single function in GFO (aka the structure of function) grasps two aspects: (i) the input-output perspective (traditional for function modeling), which includes modeling goals and side effects, and (ii) the participant perspective, covering entities involved in the realization of a function, such as 'doer', 'contributor' and 'instrument'. Moreover, functions are rarely modeled in isolation, but usually interrelated in complex networks. The Functions module introduces several distinct relations typically hidden behind functional decomposition [46, 47], among them *operand-part* and *function-subsumption*, the latter with subdivisions such as *specialization of mode of realization* and *specialization of function operand*.

The notions of the Functions module have been used for the development of FueL, a UML profile suitable for functional modeling with UML-like graphical notation [48]. The case study of applying FueL to the Gene Ontology revealed several refactoring options [49]. Additional uses in the context of the biomedical sciences include representing knowledge about biological functions [50], dispositions and functional abnormalities [51].

5. Conclusions and Future Work

We present the top-level ontology General Formal Ontology (GFO), originally released in version 1.0 in 2006, focusing on recent results and its future development. Those results include basic research on formal ontology as well as numerous practical applications, notably in the field of biomedical computer science, but also in the areas of semantic methods and knowledge representation.

The overview of existing work serves as the starting point for the establishment of an integrated framework and architecture for GFO 2.0. Based on this framework, GFO is less to be seen as a single, monolithic ontology, but rather as a system of coherent modules, which can be flexibly used together, depending on a chosen domain and context. GFO 2.0 will thus provide integration principles which should support the composition of various modules to larger unities, depending on the domain to be studied and the level of abstraction of the considered application.

The present paper addresses some of these basic topics. Architectural considerations are outlined, including an explication of the employed understanding of 'module' and further organization principles. In the second part of the paper several selected modules are surveyed and partially linked to existing formalizations and applications.

Our goal for the future is to approach the GFO 2.0 development as a software product, following state of the art industry standards. This will include an explicit roadmap and release planning, following an iterative development approach based on architectural principles for software engineering and working with a backlog. The main architectural principles adopted for GFO 2.0 are (i) a modular approach and, on that basis, (ii) a release plan adhering to an iterative methodology. Accordingly, the modules will each be released separately and in connection with rigorous versioning (major and minor revisions, snapshots, releases) and maturity level tracking for the overall project as well as for the individual modules. For each release version corresponding artifacts shall be published, e.g., on GitHub and as a paper. The present paper is a primer for the roadmap of the GFO 2.0 project.

This overall approach differs from the course of development that led to version 1.0 of GFO (and from that of initial versions of other top-level ontologies) and significantly strengthens earlier, but less systematic steps, e.g., towards a more modular structure. A TLO developed along these lines should no longer be conceived as a large, monolithic theory. In the modular approach, it is rather to be seen as a coherent framework composed of easily replaceable parts that use standardized interfaces, such that those parts constitute building blocks readily applicable to specific engineering problems.

But what will those parts, what will a module look like, given that we aim at very diverse application contexts such as conceptual modeling and the Semantic Web? Several kinds of artifacts are already foreseen or do already exist, among them (i) textual descriptions (research papers, reports, use cases, tutorials, etc.), (ii) axiomatizations in FOL, Common Logic and/or OWL, (iii) ontology patterns, e.g., in OWL and (iv) UML specifications and profiles for applications in conceptual modeling.

Returning to ontological content once more, this project will not only comprise of the modules presented herein. For example, two longstanding open issues are a dedicated ontology of generic social entities, and an extension and refinement of the theory of symbol structures. Overall, clearly and certainly we envisage new modules in the future, not at least inspired by application needs.

References

[1]   Onto-Med Research Group. Home [Internet]. IMISE, University of Leipzig, Germany; [cited 2020 Aug 15]. Available from: https://www.onto-med.de/.

[2]   Herre H, Heller B, Burek P, Hoehndorf R, Loebe F, Michalek H. General Formal Ontology (GFO) [Version 1.0]. Onto-Med Report 8. IMISE, University of Leipzig, Germany; 2006.

[3]   Herre H, Heller B, Burek P, Hoehndorf R, Loebe F, Michalek H. General Formal Ontology (GFO) [Version 1.0.1]. Draft. IMISE, University of Leipzig, Germany; 2007.

[4]   Onto-Med Research Group. General Formal Ontology (GFO): Axiomatizations and Implementations [Internet]. IMISE, University of Leipzig, Germany; [cited 2020 Aug 15]. Available from: https://www.onto-med.de/ontologies/gfo#impl.

[5]   Uciteli A, Kropf S, Weiland T, Meese S, Graef K, Rohrer S, et al. Ontology-based specification and generation of search queries for post-market surveillance. Journal of Biomedical Semantics. 2019;10(1):9.

[6]   Uciteli A, Neumann J, Tahar K, Saleh K, Stucke S, Faulbrück-Röhr S, et al. Ontology-based specification, identification and analysis of perioperative risks. Journal of Biomedical Semantics. 2017;8(1):36.

[7]   Rumbaugh J, Jacobson I, Booch G. The UML Reference Manual. 2nd ed. Addison Wesley; 2005.

[8]   ISO/IEC 21838-1. Information Technology — Top level ontologies TLO — Part 1: Requirements. 2020.

[9]   Hartmann, N. Ontologie. De Gruyter; 1935–1950 (4 Vol.).

[10] Poli R. The Basic Problem of the Theory of Levels of Reality. Axiomathes. 2001;12(3-4):261–283.

[11] Gödel K. Collected Works III. Oxford University Press; 1995.

[12] Baumann R, Loebe F, Herre H. Axiomatic theories of the ontology of time in GFO. Applied Ontology. 2014;9(3-4):171–215.

[13] Brentano F. Philosophische Untersuchungen zu Raum, Zeit und Kontinuum. Felix-Meiner Verlag; 1976.

[14] Herre H: Towards a New Foundational Ontology of Properties, Attributives and Data. In: Ontology Makes Sense. IOS Press; 2019. p. 194-210.

[15] Frege G. Über Sinn und Bedeutung. Zeitschrift für Philosophie und philosoph. Kritik. 1892;100:25–50.

[16] Russell B. On Denoting. Mind. 1905;14:479–493.

[17] Twardowski K. On the Content and Object of Presentations. Martinus Nijhoff; 1977.

[18] Tarski A. Logic, Semantics, Metamathematics: Papers from 1923 to 1938. Clarendon Press; 1956.

[19] Loebe F. Ontological Semantics: An Attempt at Foundations of Ontology Representation [PhD Thesis]. University of Leipzig, Germany; 2015. http://nbn-resolving.de/urn:nbn:de:bsz:15-qucosa-166326.

[20] Smith B, Ashburner M, Rosse C, Bard J, Bug W, Ceusters W, et al. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. Nature Biotechnology. 2007;25(11): 1251–1255.

[21] Feyerabend P. Against Method. New Left Books; 1975.

[22] Seibt J. General Processes [Habilitation Thesis]. University of Konstanz, Germany; 2004.

[23] Stell JG, West M. A four-dimensionalist mereotopology. In: FOIS 2004. IOS Press; 2004. p. 261–272.

[24] Arp R, Smith B, Spear AD. Building Ontologies with Basic Formal Ontology. MIT Press; 2015.

[25] Borgo S, Masolo C. Ontological Foundations of DOLCE. In: Theory and Applications of Ontology: Computer Applications. Springer; 2010. p. 279–295.

[26] Masolo C, Borgo S, Gangemi A, Guarino N, Oltramari A. WonderWeb Deliverable D18. LOA; 2003.

[27] Guizzardi G, Wagner G. Using the Unified Foundational Ontology (UFO) as a Foundation for General Conceptual Modeling Languages. In: Theory and Applications of Ontology: Computer Applications. Springer; 2010. p. 175–196.

[28] Guizzardi G, Herre H, Wagner G. On the General Ontological Foundations of Conceptual Modeling. In: ER 2002. LNCS 2503. Springer; 2002. p. 65–78.

[29] Guarino N, Welty C. An Overview of OntoClean. In: HB on Ontologies. Springer; 2004. p. 151–159.

[30] Stuckenschmidt H, Parent C, Spaccapietra S, editors. Modular Ontologies. LNCS 5445. Springer; 2009.

[31] Loebe F. Organization and Management of Large Categorical Systems. In: Theory and Applications of Ontology: Computer Applications. Springer; 2010. p. 67–100.

[32] Arapinis A, Vieu L. A plea for complex categories in ontologies. Applied Ontol. 2015;10(3-4):285–296.

[33] Herre H, Loebe F. A Meta-ontological Architecture for Foundational Ontologies. In: OTM/ODBASE 2005 (Part II). LNCS 3761. Springer; 2005. p. 1398–1415.

[34] Gracia JJE. Metaphysics and Its Tasks. State University of New York Press; 1999.

[35] Hoehndorf R, Loebe F, Poli R, Herre H, Kelso J. GFO-Bio. Applied Ontology. 2008;3(4):219–227.

[36] Uciteli A, Groß S, Kireyev S, Herre H. An ontologically founded architecture for information systems in clinical and epidemiological research. Journal of Biomedical Semantics. 2011;2(Suppl 4):S1.1–22.

[37] Hobbs J, Pan F. Time Ontology in OWL. Recommendation. W3C; 2006.

[38] Siemoleit S, Uciteli A, Bieck R, Herre H. Processual Reasoning over Sequences of Situations in Endoscopic Surgery. In: GMDS 2017. IOS Press; 2017. p. 222–226.

[39] Burek P, Scherf N, Herre H. A pattern-based approach to a cell tracking ontology. Procedia Computer Science. 2019;159:784–793.

[40] Tegtmeier E. Grundzüge einer kategorialen Ontologie. Alber Verlag; 1992.

[41] Siemoleit S, Uciteli A, Bieck R, Herre H. Ontological Modelling of Situational Awareness in Surgical Interventions. In: JOWO 2017. CEUR 2050. CEUR-WS.org; 2017.

[42] Burek P, Scherf N, Herre H. OWL Patterns for Modeling the Change over Time exemplified by the Cell Tracking Ontology. In: ODLS 2014. University of Leipzig, Germany; 2014.

[43] Uciteli A, Beger C, Kirsten T, Meineke FA, Herre H. Ontological Modelling and Reasoning of Phenotypes. In: JOWO 2019. CEUR 2518. CEUR-WS.org; 2019.

[44] Loebe F. Abstract vs. social roles. Applied Ontology. 2007;2(2):127–158.

[45] Loebe F. Towards an Ontology of Categories and Relations. In: JOWO 2018. CEUR 2205. CEUR-WS.org; 2018.

[46] Suh NP. Axiomatic design: advances and applications. Oxford University Press, USA; 2001.

[47] Burek P, Herre H, Loebe F. Ontological Analysis of Functional Decomposition. In: SOMET 2009. IOS Press; 2009. p. 428–439.

[48] Burek P, Loebe F, Herre H. FueL: Representing function structure and function dependencies with a UML profile for function modeling. Applied Ontology. 2016;11(2):155–203.

[49] Burek P, Loebe F, Herre H. Towards refactoring the Molecular Function Ontology with a UML profile for function modeling. Journal of Biomedical Semantics. 2017;8(1):48.

[50] Burek P, Hoehndorf R, Loebe F, Visagie J, Herre H, Kelso J. A top-level ontology of functions and its application in the Open Biomedical Ontologies. Bioinformatics. 2006;22(14):e66–e73.

[51] Hoehndorf R, Ngonga Ngomo AC, Kelso J. Applying the functional abnormality ontology pattern to anatomical functions. Journal of Biomedical Semantics. 2010;1(1):4.