BORO as a Foundation to Enterprise Ontology

Sergio de Cesare[1] and Chris Partridge[1,2]

[1]Brunel University London, U.K.

[2]BORO Solutions, London, U.K.

Abstract

Modern business organizations experience increasing challenges in the development and evolution of their enterprise systems. Typical problems include legacy re-engineering, systems integration/interoperability and the architecting of the enterprise. At the heart of all these problems is enterprise modeling. Many enterprise modeling approaches have been proposed in the literature with some based on ontology. Few however adopt a foundational ontology to underpin a range of enterprise models in a consistent and coherent manner. Fewer still take data-driven re-engineering as their natural starting point for modeling. This is the approach taken by Business Object Reference Ontology (BORO). It has two closely intertwined components: a foundational ontology and a re-engineering methodology. These were originally developed for the re-engineering of enterprise systems and subsequently evolved into approaches to enterprise architecture and systems integration. Together these components are used to systematically unearth reusable and generalized business patterns from existing data. Most of these patterns have been developed for the enterprise context and have been successfully applied in several commercial projects within the financial, defense, and oil and gas industries. BORO's foundational ontology is grounded in philosophy and its metaontological choices (including perdurantism, extensionalism, and possible worlds) follow well-established theories. BORO's re-engineering methodology is rooted in the philosophical notion of grounding; it emerged from the practice of deploying its foundational ontology and has been refined over the last 25 years. This paper presents BORO and its application to enterprise modeling.

Keywords: BORO, foundational ontology, generalized business patterns, perdurantism, extensionalism, mereology, grounding, set theory, legacy re-engineering, enterprise architecture, integration, reuse.

1. Introduction

The Business Object Reference Ontology BORO has chosen to adopt a closer integration with philosophy than other ontologies in the information systems domain such as, for example, Bunge-Wand-Weber (BWW) (Wand and Weber, 1993) and the Resource Event Agent Enterprise Ontology (REA-EO) (Geerts and McCarthy, 2002). Also, unlike them, it emerged from and developed in commercial projects rather than in academia.

BORO includes a foundational (or upper) ontology and a closely intertwined methodology for information systems (IS) re-engineering (Partridge, 1996), hence the term BORO refers to both the ontology and the methodology. BORO was originally conceived in the late 1980s to address a particular need for a solid legacy re-engineering process and then evolved to address a wider need for developing enterprise systems in a 'better way'; in other words in a way that was less cumbersome, compared to the heavyweight methodologies of the time, enabling higher levels of reuse and, as a consequence, capable of reducing the effort and cost of (re-)developing, maintaining and interoperating enterprise systems. It was eventually publicly documented in (Partridge, 1996).

The BORO Foundational Ontology is strongly rooted in philosophical ontology. Ontology is defined by Jonathan Lowe as "the set of things whose existence is acknowledged by a particular theory or system of thought" (Honderich, 2006, 670). This definition is particularly relevant in the context of enterprise modeling and systems development since it grounds ontology in reality (i.e. "the things whose existence is acknowledged") rather than one's subjective conception of what constitutes the real world. As such BORO is a realist ontology, one that recognizes the existence of an objective reality. In contrast, an alternative stance is conceptual idealism, which considers reality as mentally constructed. Realism is one of the metaphysical (or metaontological) choices that underpin BORO (Partridge et al., 2012). A key motivation for the choice of realism is that it It can be argued that it (along with other metaontological choices discussed further in this paper) increases the likelihood that different enterprise modelers will represent the same business objects (or in general 'reality') in the same way.

BORO's origin and predominant area of application has been the enterprise. As an enterprise ontology its suitability has been demonstrated in many industrial projects across a range of business domains including finance, oil and gas, and defense. While BORO is one of the few enterprise ontologies to have been adopted in industrial and commercial sectors, BORO's application has not been limited to the enterprise context; examples include multisensor defense systems, relativistic time, and spatial and temporal boundaries. The general nature of BORO increases its level of applicability as an enterprise ontology since innovation provides opportunities for continuously introducing new types of elements (e.g., technologies, business models, processes, product, services, contracts, etc.) that may be not only unforeseen but also significantly different from what pre-existed.

This paper is aimed at presenting BORO (both the foundational ontology and methodology) and demonstrating its suitability for representing the enterprise domain. This involves bringing together and summarizing in an effective manner multiple sources of information and the many previously published sources that adopt BORO. In order to do so a systematic framework must be adopted. For this reason the structure of the paper reflects the Ontology Documentation and Analysis Framework (ODAF) described by Geerts (2016) with the following sections discussing BORO in light of each ODAF component. Conclusions are then presented in the last section of the paper.

2. Content

Central to BORO is a top level that provides the framework for the rest of the ontology. This top level is presented in this section. The top level is framed by a number of explicit ontological choices, described later in this paper. These crystalize into a system of ontological categories, the top level objects (Section 2.1). These top level objects are characterized by top level patterns (Section 2.2) including patterns that describe types of relationships or tuple types (Section 2.3).

2.1 BORO’s Top Level Objects

A foundational ontology can be defined as an ontology that “defines a range of top-level domain-independent ontological categories, which form a general foundation for more elaborated domain-specific ontologies” (Guizzardi and Wagner, 2004). A categorical foundational ontology is one where these ontological categories are disjoint and form a complete partition of the set of all things that exist; this means that given any object, that object must instantiate one and only one top-level category.

BORO is a categorical foundational ontology, and as such it is theoretically not constrained in what it can model. Pragmatically, as a formal ontology, it is more usefully focused on domains that have been, or need to be, formalized, in particular automated computer systems. This is because it is significantly less effort to build a formal ontology for a domain that is already formalized (for example, where computer systems exist), than one that has never been formalized (for example, where all the current processes are manual).

Figure 1 represents BORO's top level. The notation used is that of the Unified Modeling Language's (UML) class diagrams (OMG, 2015) with the semantics of BORO. This variant of UML is called BUML (or BORO-UML) and is used in the UML-based figures in this paper. A basic understanding of UML class diagrams is sufficient for the reader to understand the models and examples in this paper, though it is worth remembering that these use a BORO semantics.

In the BUML model of Figure 1 *Objects* represents the three top level BORO categories: *Elements*, *Types* and *tuples*. Every object belongs to one and only one of the three categories which are framed, as mentioned earlier, by a range of metaphysical choices. These choices mean that, within BORO, each category has its own identity criteria.

- *Elements* are individual objects whose identity is given by the element's spatiotemporal extent (or extension); i.e. the space and time it occupies. BORO simplifies things by assuming that matter and space-time are identical (this is a metaphysical stance that has been called super-substantivalism (Sklar, 1974; Schaffer, 2009). An example of an element would be the person *John*.
- *Types* are collections of any type of object (in other words, objects of any of the three categories). The identity of a type is determined by its extension, the collection of its instances (i.e. members). For example, the extension of the type *Persons* is the set of all people. In BORO, *Types* play a similar role to universals in other foundational ontologies.
- *Tuples* are relationships between objects. The identity of a tuple is defined by the places in the tuple. An example is (*Mary*, *John*) in which the elements *Mary* and *John* occupy places 1 and 2 in the tuple respectively. *Tuples* can be collected into types, called tuple types. An example is *parentOf,* which is the collection of all relationships between parents and their children. Section 2.3 will describe tuple types and their top level patterns in more detail.

There is a system of ontological dependence relations between these categories. One rather abstract way of developing an understanding of these, and so developing a better understanding of the categories is through grounding (Fine, 2010), which provides a kind of ontogenesis narrative for the objects in the ontology. The grounding (ontogenesis) narrative starts with a single element, the pluriverse of all possible worlds (a position Schaffer (2010) calls 'priority monism'). Consider the generative operation of decomposition that divides an element into all its parts. If we apply this to the pluriverse we then have all the elements.

This operation exhausts all elements as the pluriverse and its parts are all the elements. Then consider the generative type-builder operation; we can then apply this to the (previously generated) elements to build the type Elements; this is the ontological category of Elements. Then consider the generative operation power-type-builder (power-types are described in more detail below). Apply the powertype-builder operation to the set Elements – this builds the type that has all the subsets of Elements as its members. Applying the power type-builder operation repeatedly builds a type hierarchy. Finally, consider the generative tuple-builder operation, this takes a number of any type of object, including tuples, and organizes them into a tuple. This grounding approach is reflected in the BORO methodology. An example is provided in Partridge (2002a).

A more concrete way to develop understanding is through specific examples. We illustrate the example given earlier in Figure 2 (individual elements and tuples are represented as grey rectangles; the dashed lines represent type-instance relationships).

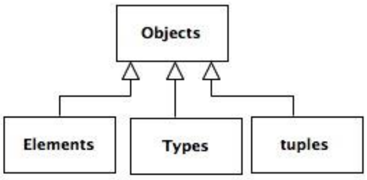

Figure 1. BORO Top Level

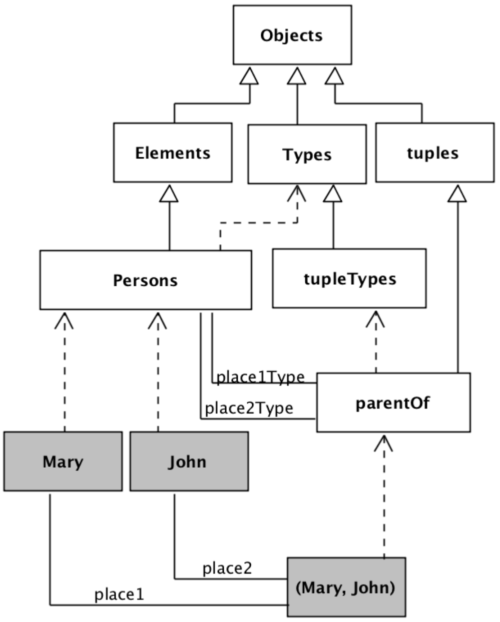

Figure 2. Example

The examples used in this section and the following will focus on the general example of persons. Subsequently we will use enterprise specific examples. The main reason for adopting a more general example here is to later demonstrate how the BORO patterns are easily applicable to the more specialized enterprise domain. Generality enables a high level of reuse, which is widely considered beneficial in information systems development and integration.

2.2 BORO's Top Level Patterns

As noted above, in BORO the identity criterion for an element is its spatiotemporal extent, hence an extensional criterion of identity. Two elements are different if they have different spatiotemporal extents. This is consistent with the philosophical theory of persistence and identity known as perdurantism (or 4D which integrates or combines 3D space and 1D time). In 4D an individual can be extended through time and, if so, is not fully present at any instant in time. One result of this is that objects are modally flat (Lewis 1986); i.e. the other possible ways they could be are counterparts, different objects. An opposing theory is endurantism (or 3D) whereby an individual object is fully present at every instant of its (possible) existence. 3D individuals are typically modally extended; i.e. the other possible ways they could be are exactly the same object. In this case, their identity is typically based upon a criterion they inherit from their defining kind, which is the basis for the identifying characteristics. As objects are modally extended, these need to pick out as the identical object the other ways the object could be. One way of contrasting the two approaches is the difference in the ease with which the criterion of identity can be articulated. The 4D extensionalist has a simple straightforward general criterion of identity for all individuals – their extension. The 3D-ist has multiple criteria of identity, one for each disjoint kind, which has characteristics that define when it is identical. The 3D approach has been well defined in principle (Wiggins, 2001), but people working in this area have found it difficult to articulate the criterion exactly for most natural kinds (person being one such example). So 3D *John* as a child is the same person as *John* as an adult under the relevant criterion of identity for person (as a kind) – though the exact details of the criterion are not yet known. In 4D *John* is just the spatio-temporal extension that stretches from his birth to his death. *John* as a child and *John* as an adult are temporal parts of *John* as a whole. The simple general criterion of identity here is exact.

Figure 3 represents the example above from a 4D perspective with a space-time map. The vertical axis represents space and the horizontal axis represents time (for illustrative purposes only one spatial dimension is represented). Space-time maps are an effective diagramming technique for visualizing 4D extents and their temporal part relationships. In the example each of John's temporal parts are bounded by start and end boundaries.

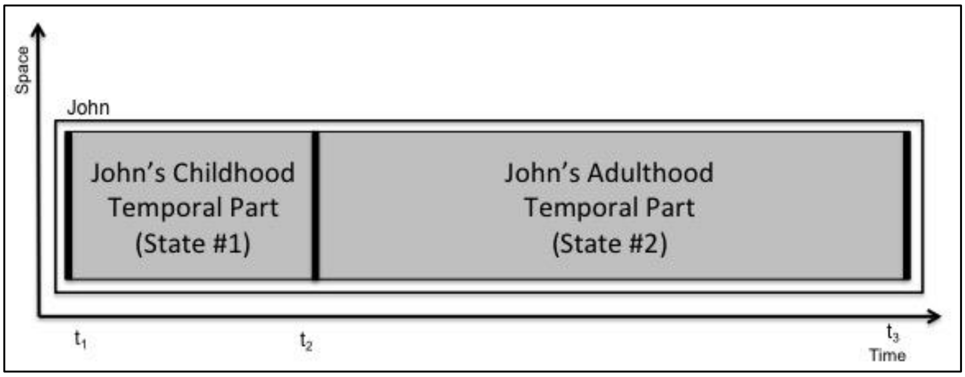

Figure 3. Space-time map exemplifying temporal parts

Figure 4 is a model of BORO's pattern for whole-part relationships and Figures 5 and 6 show how the pattern (or the generalized reusable component) is applied to model the example. Specifically *John's Childhood* and *John's Adulthood* are represented as instances of *PersonStates*; these are elements that represent stages of persons hence the temporal part relationship with *Persons*. As the diagram illustrates, any new element, type or tuple that is defined must ultimately instantiate one of the top-level ontological categories (or types), i.e. *Elements*, *Types* or *tuples*.

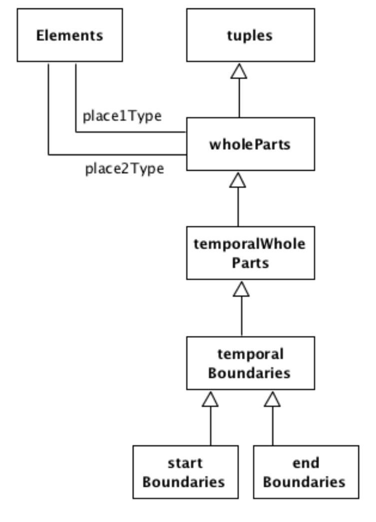

Figure 4. Upper level pattern of whole-part tuples

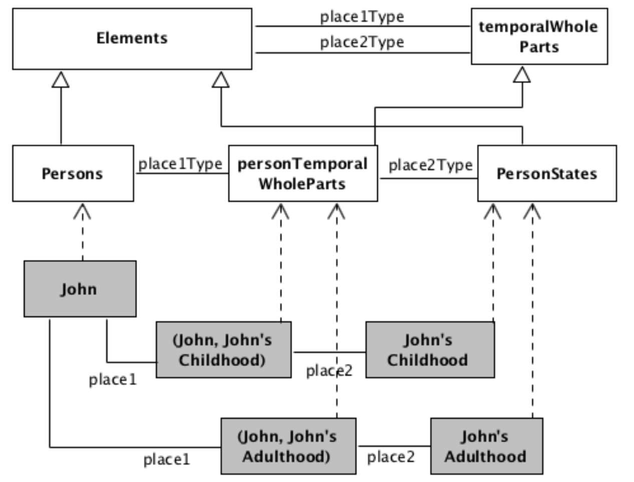

Figure 5. Example of temporal whole part tuples

While the example focuses on temporal whole-part relationships, it is important to note that *temporalWholeParts* is a subtype of the tuple type *wholeParts*; the *wholeParts* relationship applies when an element is part of another for the whole existence of the latter. In other words, this is a general spatio-temporal whole-part relationship and not restricted to spatial wholes-parts. For example, John's head is a part of John during his whole life and not for a portion of it.

In BORO an element can be temporally bounded by other elements. Partridge (1996) originally called these boundary elements 'events'. *temporalBoundaries* is the tuple type representing such relationships and it is a subtype of *temporalWholeParts*. In the example (Figure 6), *John's Birth* and *John's 18th Birthday* temporally bound *John's Childhood* at its start and at its end, therefore the respective tuples instantiate *startBoundaries* and *endBoundaries* respectively. The bounding elements are temporal parts of *John's Childhood*.

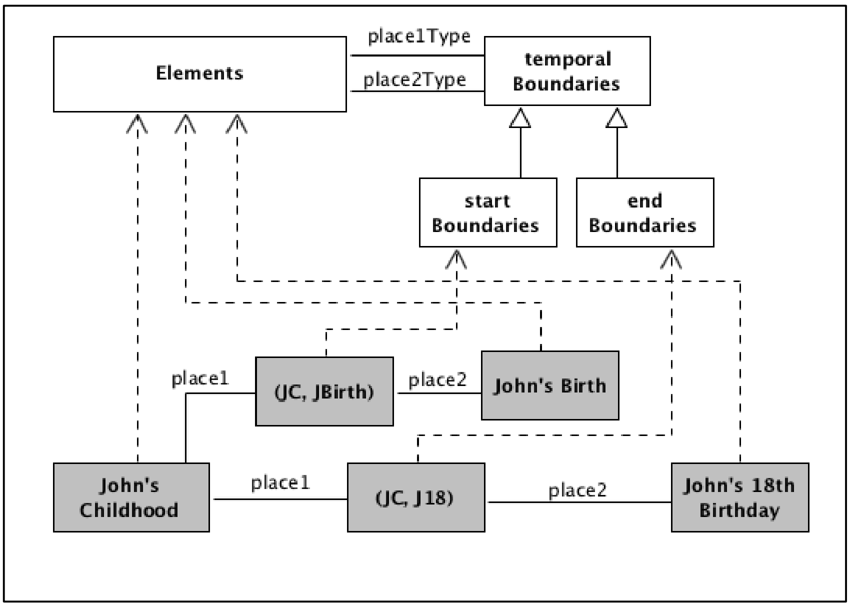

Figure 6. Example of temporal boundaries in temporal whole part tuples

BOROs approach enables a particular concrete approach to times. Consider a universe extended in both space and time. Then consider a portion of that universe sliced along the time dimension; sliced at one end at the beginning of the year 1980 (say) and sliced at the other at the end of the year 1980. This time slice of the universe is, in BORO, the year 1980. Different times are different slices. This allows a particularly concrete way of talking about events in relation to times – which allows for whole-parts patterns to be generalized across times. This is modeled in Figure 7, which shows the upper-level *happensIn* pattern for representing the 'occurrence' of an element in a specific time period (interval) or instant. For convenience rather than refer to a period or instant, the type *PeriodsOrInstants* (Figure 8) may be used. The example in Figure 9 represents *John's Birth* occurring on the day of *30 June 1980*. Given that *happensIn* is a subtype of *wholeParts*, *John's Birth* is a part of the (concrete) day *30 June 1980*.

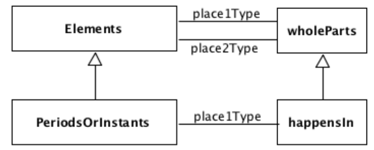

Figure 7. Upper level pattern: happensIn

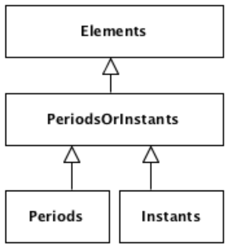

Figure 8. Upper level pattern: Time periods and instants

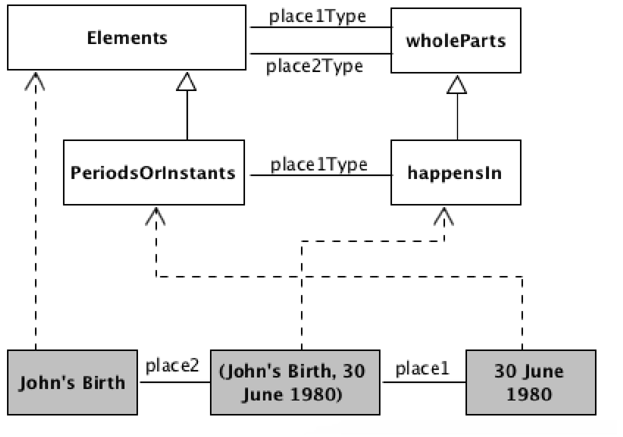

Figure 9. Example of an element occurring in a time period

In order to represent temporal sequencing between elements the *before-after* upper level pattern is used as demonstrated in Figure 10.

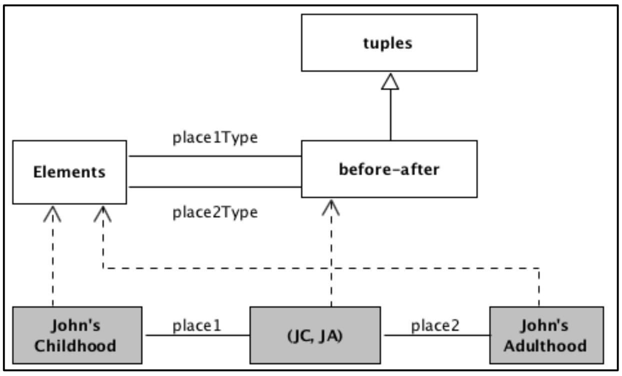

Figure 10. Upper level pattern for temporal sequencing of elements

This section has presented some of BORO's upper level patterns for modeling 4D extents (elements) and their mereological (whole-part) relationships. The following two subsections will discuss types and tuples. From this point onward all examples will be enterprise specific.

Immutability of Types

A key metaphysical choice is how to treat change over time. This choice affects the nature of types. BORO's extensional choice for elements simplifies the extensional choice for types. As noted earlier, BORO types are extensional. The extension of a type is its instances; two types are different if and only if they have different extensions. Sets are similarly extensional, as the extension of a set is its members and two sets are different if and only if they have different members-extensions. One consequence of this is that BORO *Types* (and sets) are immutable. They cannot change their instances-members over time. This feature underwrites the type-builder operation described above – given all the instances of a type (members of a set) you can characterize the identity of the type (set).

An alternative, non-extensional choice, is mutable types. The following example illustrates the difference. Consider the type *UKCompanies*, its instances are all past, present and future companies of the United Kingdom. In BORO the immutability of types implies that if an element is an instance of a type, it cannot subsequently cease to be an instance of that type. The latter statement may appear counterintuitive since one may argue, for example, that an individual company (e.g., *Acme Ltd.*) will eventually be dissolved. One would assume that the set of *UKCompanies* before and after Acme's dissolution would resemble Figure 11 with the type having a changing extension over time. In BORO, these would be two different types: the type of UK Companies at t_1 and the type of UK Companies at t_2.

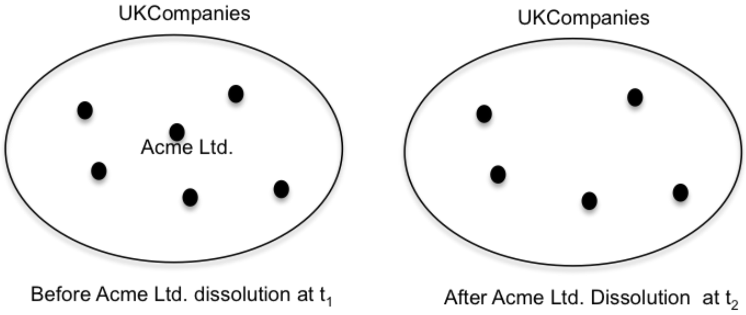

Figure 11. Different set memberships at different times (a non BORO view)

The representation in Figure 11 is more compatible with an endurantist-based ontology (for example, the Unified Foundational Ontology (Guizzardi, 2005)), which does not consider individual objects to be temporally extended nor composed of temporal parts. To better explain the benefits acquired from an IS engineering perspective when adopting 4D individuals, one can consider change in the context of an enterprise.

Figure 12 illustrates an example with a space-time map that adopts a similar pattern to Figure 3, this time applicable to the enterprise. In this example *Acme Ltd.* changes its business activity from paper production to the manufacturing of electronic devices. These changes are quite frequent and are recorded by national company registrars (such as Companies House in the United Kingdom). Normally this occurs by filing with the registrar a change in the company's Standard Industry Classification (SIC) code. A 3D interpretation of this scenario would describe the types involved as depicted in Figure 13. In the figure *Acme*

Ltd. is an instance of *Paper Companies* from t_1 to t_2, at t_2 de-instantiates Paper Companies and instantiates *Electronics Companies* from t_2 to t_3. With this representation, information concerning the previous history of the company is lost. From an IS engineering perspective a design decision (outside of the foundational ontology adopted) would be required to maintain that history.

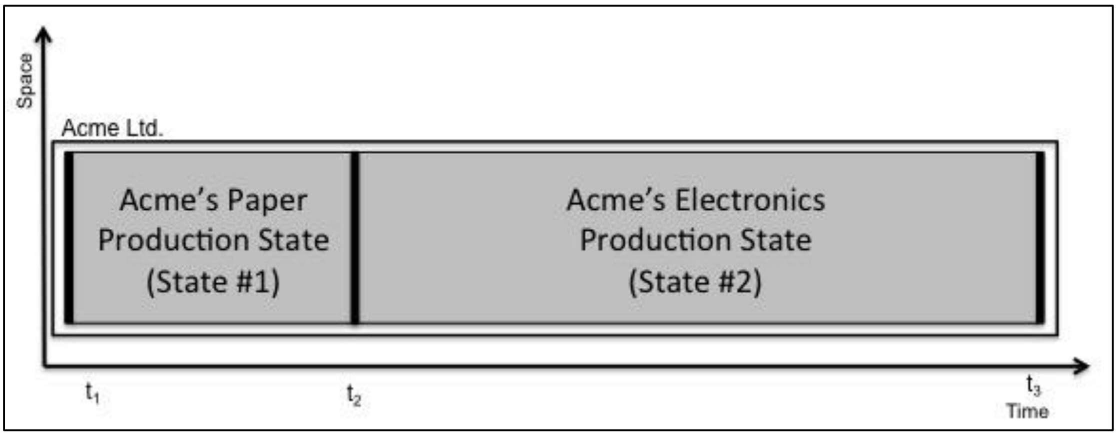

Figure 12. Change of an enterprise's business activity – space-time map

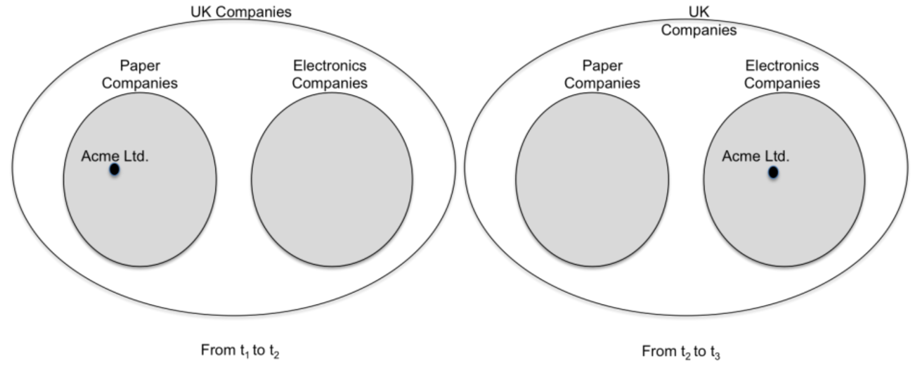

Figure 13. Change of a company's business activity (a non-BORO view)

Figure 14 shows how the same scenario is represented in BORO and which types are instantiated. *Acme Ltd.* is always an instance of *UK Companies*. The two states of *Acme Ltd.* instantiate *Paper Production States* and *Electronics Production States* respectively. Like the relationship between *Acme Ltd.* and *UK Companies*, *State#1* and *State#2* always instantiate their respective types. The four types in Figure 14 are therefore immutable; at all times their instances remain the same. In order to represent the times at which the change(s) occur, the pattern in Figure 7 is applied. From an IS engineering perspective, the ontology provides all the necessary constructs to underpin the IS design.

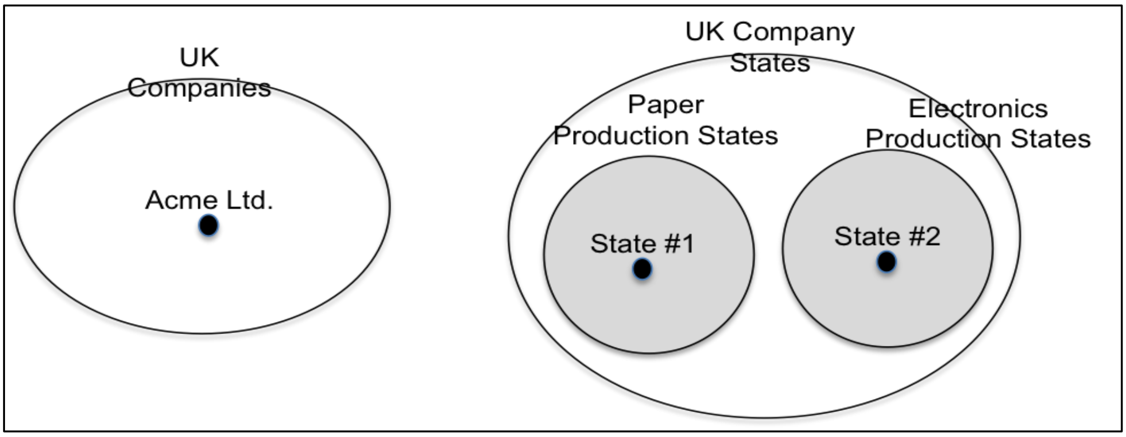

Figure 14. Immutable types and temporal parts

Powertypes

The *Powertype-builder* operation, noted earlier, and the powertypes it builds are an important part of the BORO framework. In BORO a powertype is broadly equivalent to powerset in set theory (as defined by axiom VI (Zermelo, 1908)) where it is defined as the set of all subsets of a given set. As a consequence if A is a type then P(A), its powertype has as its instances all subtypes of A (and all instances of P(A) are subsets of A). From a grounding perspective, one can see powertypes are a generative mechanism for raising types to the next level, in a similar way to Boolos's (1971) iterative conception of a set. They are also useful as a boundary mechanism for classifications of a type, as exemplified in Sections 6 and 10. In BORO the relationship between a type and its powertype is called *powertypeInstances*. At a foundational level examples of powertype instance relations are shown in Figures 15 and 16.

Figure 15 provides two examples of powertypes. These are:

- *ElementTypes*: The powertype of *Elements*, hence it has as instances all subtypes of Elements.
- *tupleTypes*: The powertype of *tuples*, hence it has as instances all subtypes of tuples.

Obviously these are not the only subtypes of *Types*. In fact, the type with these two instances *{John, Acme Employees Type}* is a perfectly valid type in BORO and illustrates that a type can have instances of any type, which, in this case, are an element and a type.

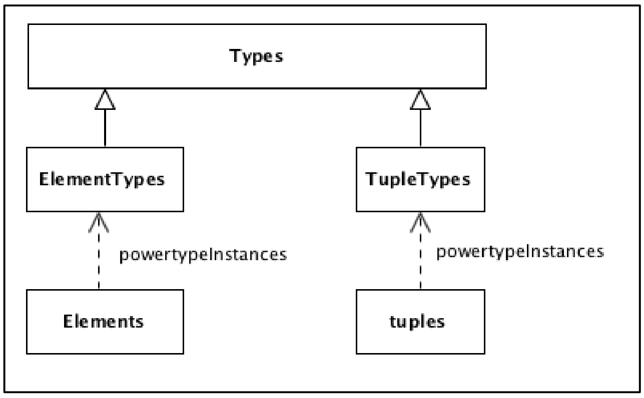

Figure 15: Types

2.3 BORO Top Level Tuple Types

In BORO a tuple is a relationship between two or more objects. Tuple types are sets of tuples. Relationships between two objects are called *couples*. The various common foundational tuple types (see Figure 16) have been defined or used in the previous sections. For convenience these can be summarized here as follows:

- *superSubTypes*: Relationships between two types (which corresponds to the subset relation between two sets). For example, there is a superSubType relation between *Cars* (the supertype) and *SportCars* (the subtype). By definition all instances of *SportCars* are instances of *Cars*.
- *typeInstances*: Relationships between a type and any of its instances corresponding to set membership. For example, the relation between *Cars* (the type) and *John's Car* (the instance).
- *powertypeInstances*: Relationships between a type and its powertype. *powertypeInstances* is a subtype of *typeInstances*.
- *wholeParts:* Relationships between two elements in which the 4D extent of one element is completely contained within that of another element for the entire existence of both.
- *temporalWholeParts*: Relationships between two elements in which the 4D extent of one element is completely contained within that of another element but only for a particular period of time. Note that *temporalWholeParts* is a sub type of *wholeParts* so an instance of the subtype is also an instance of the supertype.

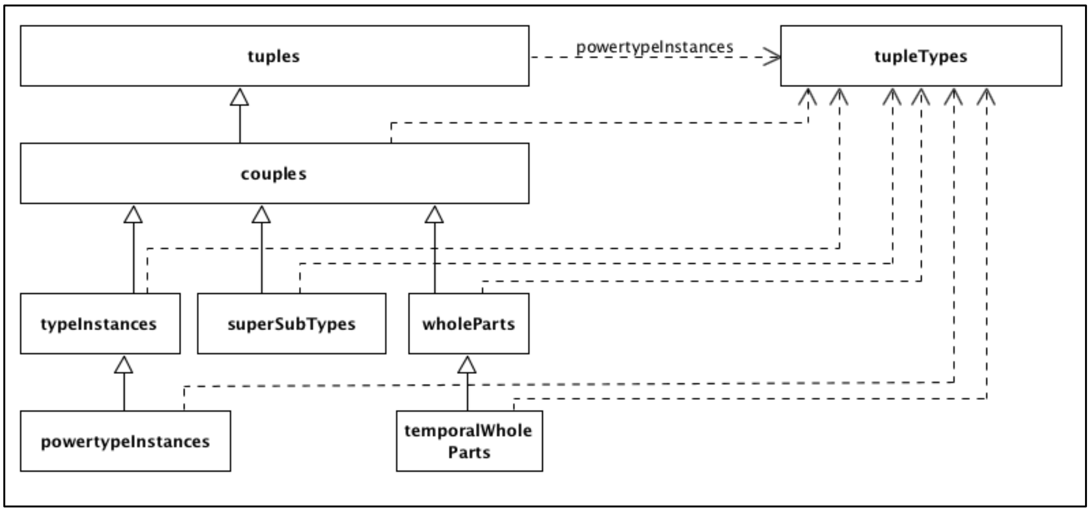

Figure 16: Foundational tuple types

In Figure 16 the foundational tuples are subtypes of *tuples* and instances of *tupleTypes*. This is due to the powertypeinstance relationship (defined above and exemplified in Section 10) between *tuples* and *tupleTypes*.

Figure 17 shows the place types for each of the foundational tuple types defined above. *typeInstances, powertypeInstances* and *superSubTypes* are normally represented with the UML instantiation (dashed line and arrow) or subclass notation (full arrowhead) as in the figures above. In the UML instantiation is a type of dependency as defined in the Object Management Group (OMG) specification of UML 2.5 (OMG, 2015). Figure 17 reifies these tuple types in order to explicitly show the place types.

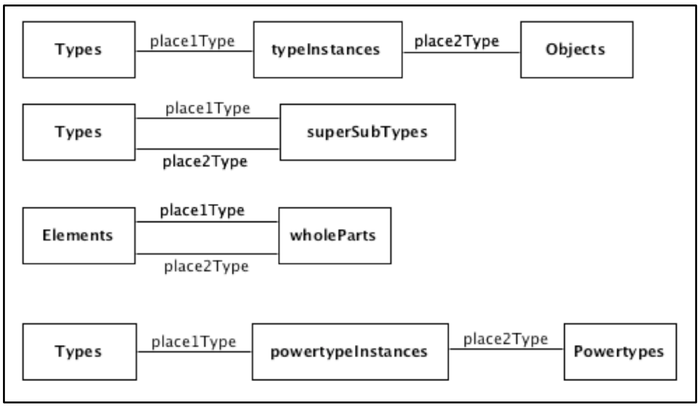

Figure 17: Foundational tuple types and respective place types

3. Purpose

The original purpose of BORO was to enable the re-engineering of legacy systems, since at the time (i.e. the end of the 1980s and early 1990s) and still today there are not many used and tested methodologies for this. BORO evolved into a more effective and efficient general methodology for the early modeling stages of system development compared to the methodologies that existed at the time (Partridge, 1996). Effective re-engineering here refers to the ability to salvage the business knowledge embedded in the existing system for reuse in the new system while, at the same time, being able to interpret and represent such knowledge in a more general and reusable manner. Reusability has always been at the heart of the BORO methodology and the foundational ontology is key to driving the semantic interpretation and discovery of generalized business patterns from the original legacy data.

Generalized business patterns are ontological models that are (1) systematically derived from legacy system data; (2) grounded in the BORO foundational ontology; and (3) applicable across organizations and/or domains. Section 2 presented examples of generalized business patterns at the foundational level (e.g., temporal whole-parts, types-instances, superSubTypes and powertypes). Examples of enterprise domain patterns will be presented in Section 4 and include, for example, contracts/agreements, business processes, accounting, geopolitical regions and naming.

Reusability of existing foundational and domain patterns increases efficiency since patterns previously discovered can be applied to problems that have already been modeled, hence reducing the time required to semantically interpret and model the legacy representation.

Legacy re-engineering with BORO involves three major activities (Partridge, 1996; Daga et al., 2005): (1) semantic interpretation of existing data; (2) semantic improvement of the original data or model with possible discovery of new patterns; and (3) harmonization of the new ontological models with existing ones. While originally conceived for legacy re-engineering, since the activities of the BORO methodology are more generally about semantic model improvement and transformation, BORO has been more recently applied to other related information systems engineering purposes. These mainly include the following interrelated areas:

- Enterprise Architecture and interoperability: Significant work has been carried out to develop reference architectures for enterprise architecture data exchange. Most of the reference architectures and standards developed (and being developed) for this purpose are in the defense sector. These include the International Defence Enterprise Architecture Specification (IDEAS) Group (IDEAS, 2016), the British Ministry of Defence Architecture Framework (MODAF) Ontological Data Exchange Model (MODEM) (Hagenbo et al., 2012) and the Unified Profile for DoDAF/MODAF (UPDM) (OMG, 2015). The latter is the product of an OMG initiative to develop a modeling standard that supports both the USA Department of Defense Architecture Framework (DoDAF) (McDaniel, 2012) and the UK Ministry of Defence Architecture Framework (MODAF).

- Enterprise systems integration and interoperability: BORO has been adopted in industry to underpin the enterprise architectures of large organizations, for example in the oil and gas sector. In this case the BORO foundation and its ontological patterns provided a common semantic grounding for a shared architecture across the business processes and the enterprise systems of the companies involved. A standard of the International Standards Organization (ISO) which was influenced by BORO is ISO 15926-2:2003 titled “Industrial automation systems and integration - Integration of life-cycle data for process plants including oil and gas production facilities - Part 2: Data model” (West, 2010).
- Tool interoperability: As a consequence of the reference architectures developed by the IDEAS Group (IDEAS, 2016), different tool vendors have created profiles for supporting for the interoperability of models developed in IDEAS.

BORO is also currently being adopted in research projects of the Engineering and Physical Sciences Research Council (EPSRC) in the areas of Semantic Business Process Management (de Cesare and Lycett, 2013) and credit risk assessment (Lycett et al., 2014). In the former the emphasis is on the discovery of ontological business process patterns, while in the latter BORO is adopted as a means to semantically engineer heterogeneous company datasets as well as for the purpose of modeling an ontology of credit risk.

Section 11 will provide further detail concerning the standards and reference architectures mentioned above within the context of BORO’s evolution and history.

4. Scope

In the context of Enterprise Modeling BORO has been applied to many enterprise related projects from which a variety of generalized business patterns (GBPs) have been modeled. Patterns have been developed for a variety of objects ranging from concrete artifacts such as pumps to more intentionally (or socially) constructed objects (Searle, 1995); these include things like contracts, processes, services and capabilities. As it may be clear from previous sections, in BORO intentionally constructed objects are elements with 4D extent. In modeling them, mereology (the study of whole-parts) plays a key role with the BORO foundational whole-parts pattern being fundamental.

As an example let us consider business processes. In the literature a business process is typically defined as ""a collection of activities whose final aim is the production of a specific output that is of value to the customer" (Hammer and Champy, 1993, p. 85). While definitions of this kind provide the reader with a broad understanding of what processes may be, they are not however sufficient in explaining the physical reality of a process. Since all elements in BORO are 4D extents, processes also are physical and not abstract (as many would consider them to be). In BORO the extent of processes overlap with the extents of the people, machines, documents, tools, etc. that are involved in the processes. Let us consider a car repair process instance in which *Joe the mechanic* fixes *car XYZ*. To keep the example simple we will ignore any tools and car parts used. In this simplified example there are two elements involved Joe and the car, precisely a temporal part of Joe and a temporal part of car XYZ. In BORO the process of this example is the mereological sum (or fusion) of these two temporal parts (Lewis, 1986; Sider, 2005). The example is illustrated in Figure 18. The repair process of Car XYZ is therefore the mereological sum of states 1 and 2. It is important to note that these states are bounded by boundary elements (typically known as events), which initiate and end the two states. These boundary elements are represented with thicker black lines in the space-time map and, by virtue of the axiom defined in Figure 4, are temporal parts of their respective states.

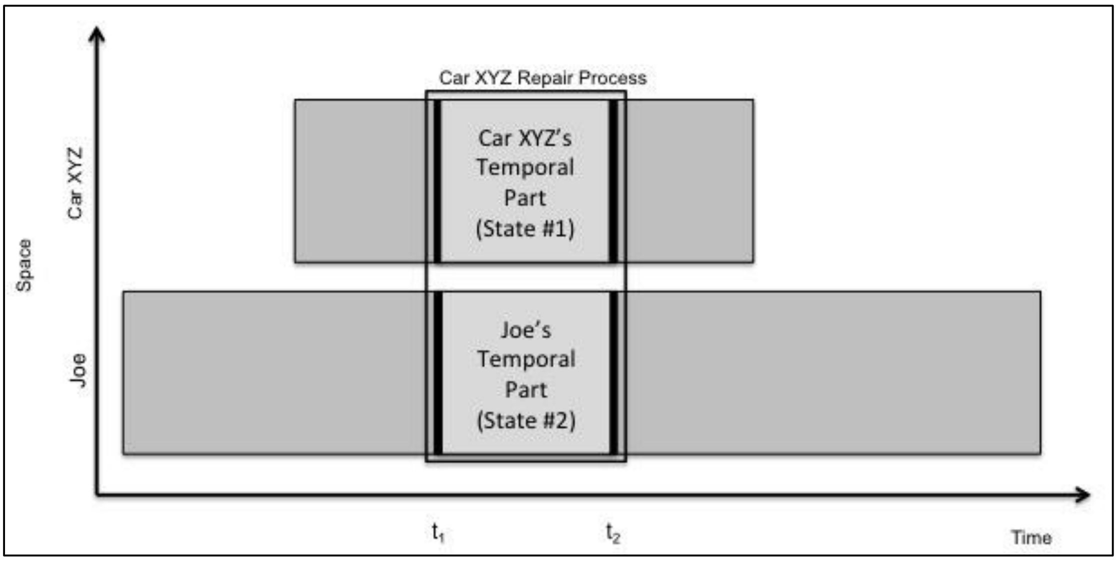

Figure 18: Car repair example space-time map

A similar pattern can be observed in the case of contract executions. Here two parties assume contractual commitments and take part in events (such as payments and deliveries) to fulfill their commitments. As explained in de Cesare and Geerts (2012) the contract execution is also a mereological sum of the contractual parties involved. Processes and contracts are two typical examples in which perdurantism and mereology combine to provide a semantically powerful explanation of many intentionally constructed objects.

The scope of BORO as an enterprise ontology is determined by the set of general business patterns that have been re-engineered from existing systems and data over the course of many projects. Besides processes and contracts numerous other patterns have been modeled and discussed in various publications. For example, the double entry bookkeeping pattern has been re-engineered and given a new shape in Partridge (2002b). Table 1 provides a list with their sources.

Pattern	Source
Names	Partridge (1996, 2005)
Geopolitical Regions	Partridge (1996), Daga et al. (2005)
Contracts/Agreements/Deals	Partridge (1996), de Cesare and Geerts (2012)
Processes	Partridge (2002a), de Cesare et al. (2016)
Roles	Partridge (2002a), West (2010), de Cesare et al (2015)
Services	Partridge and Bailey (2010)
Capabilities	UPDM (2015)
Accounting	Partridge (2002b)
Classification/ Classification Systems	Partridge et al. (2015)

Table 1: General Business Patterns and sources

5. Source

BORO's foundational ontology and its generalized ontological business patterns derive from the following sources respectively: (1) metaphysics or the metaontological choices one can make about reality and (2) semantic re-engineering of existing knowledge sources (e.g., legacy systems, existing enterprise data sources and models, existing standards, etc.). In this section we will focus on the metaontological choices that the BORO foundation makes, while the following section on ‘elicitation’ will be dedicated to how the BORO methodology works in order to semantically transform legacy knowledge sources and subsequently empirically discover new ontological patterns.

As discussed in the introduction, BORO is based on philosophical ontology. In Philosophy ontology, as a discipline, is the study of ‘what there is’ (Hofweber, 2014), derived from Quine’s (1948) original question (“What is there?”). The scope is quite broad and unlike other disciplines, such as biology and sociology, which study something, ontology studies everything (Berto and Plebani, 2015).

This broad definition helps to introduce the types of questions that one needs to ask in order to know about ‘what there is’ and to start understanding the theoretical underpinnings of a foundational ontology. These questions include, for example, what kinds of things exist, how something extends across space-time, when two things are the same or different (identity) and so on. More generally questions of this kind tend to focus on the objects that exist and the relations between them. In this sense, in Philosophy an objectification of ontology has developed whereby definitions, such as Lowe’s (Honderich, 2006) presented in the introduction, consider ontology as “the set of things …”. This sense of ontology is particularly useful in a field like Enterprise Modeling (and more generally in business and engineering) whose main purpose is to represent the things in the enterprise domain (or more generally, the world). Different choices lead to different characterizations of reality and, as a consequence, lead to foundational ontologies that produce different representations of the enterprise. Different types of enterprise design can have a significant resource impact on issues such as systems development, re-engineering, integration and interoperability.

The types of choices and commitments that a foundational ontology makes draw upon metaphysical theories of reality. In recent times the term metaontology was introduced by van Inwagen (1998) to refer to these metaphysical choices. BORO’s metaontological choices can be summarized as follows (Partridge, 2002c; Partridge et al., 2012):

- Realism: BORO is a realist ontology whereby there exists an objective reality (for example, see (Smith, 2004)). This is opposed to a conceptual idealist stance whereby reality is individually constructed by one’s own concepts (or ideas); in other words reality is the result of subjective interpretation. Since BORO adopts a realist stance, the objects that BORO models are objects in the real world. An idealist would instead claim to model one’s mental conception of what exists.
- Perdurantism (or 4D): BORO is a perdurantist ontology whereby individual objects (elements) extend spatially and temporally; therefore, a BORO element is never wholly present at a specific instant in time. Identity is defined by an element’s spatiotemporal extension. The opposing theory is endurantism whereby individual objects are fully present at any given time and do not extend temporally. The identity of an individual object is defined by its essential properties (or attributes) (Sider, 2005).
- Physical objects: All individual objects are physical. In BORO there are no abstract objects. BORO types and tuples are not physical, yet they are grounded in the physical world. First-order types are physically grounded in their physical instances, higher-order types

are ultimately grounded, via the type-instance hierarchy, to first-order types and, therefore, to 4D extents. Tuples are physically grounded via their places. As Section 4 illustrated even intentionally (or socially constructed) objects are physical and not abstract.

- Possible worlds: BORO adopts Lewis' (1986) theory of possible worlds and counterparts. In other words, it is possible to model in BORO any number of possible worlds including our actual world. Possible worlds are used, for example, to model possible future scenarios, as is the case of the future execution of a contract, not known at the time when the parties agree to their contractual obligations. Lewis' counterparts relate things in different possible worlds; for example, the John Smith who pays on time in possible world 1 as opposed to the John Smith who pays late in possible world 2. (This is an example of modal flatness mentioned earlier).

As Sections 6 and 10 emphasize, the BORO ontology has been empirically evaluated over the course of many years in multiple industrial/commercial projects. The metaontological choices discussed above have demonstrated to be appropriate in terms of cost effectiveness and productivity.

6. Elicitation

While the BORO foundational ontology is grounded in the set of metaontological choices described in Section 5, the content of BORO domain ontologies (or generalized business patterns) is derived empirically via the BORO methodology, semantically driven by the foundational ontology and existing generalized business patterns (GBPs).

BORO has adopted an outside-in approach (top-down and bottom-up) to address two different modeling requirements. As philosophers have recognized for a while, there are a variety of very general metaphysical questions that cannot be resolved purely by empirical research. A good modeling environment needs to address these in a coordinated and systematic way. One clear advantage of doing this is that it provides a well-designed framework for modeling. The top-down approach looks to provide a foundational ontology that fixes the metaphysical choices across the model. A minimalistic approach has been adopted, where the foundational level only aims to deal with metaphysical choices and not to overstep the boundary into the empirical domain (what Quine (1948, p. 4.) would call a desert landscape).

Secondly, there is a requirement that the model reflects the world well. This is addressed with a scaled, bottom-up, grounded approach. Bottom up in that it starts with particulars, which are usually simplest to identify. Grounded in that the building of types and tuples is always grounded in particulars. Scaled in that it aims to include as much data as possible in the model, typically all the data from a number of candidate computer systems. This helps to ensure that the model is empirically grounded. It also leads to two unusual aspects; firstly BORO models also contain particulars/data, where traditional models usually only contain types/data schemas – even then, often only the significant types/data schemas. Secondly BORO models contain as much data as is available.

For the purposes of this paper, and in order to demonstrate the way in which ontological models are derived from the enterprise data of legacy systems, an example related to countries and, more generally, geopolitical regions is presented here (Daga et al., 2005).

Geopolitical Regions is one of the generalized business patterns described in Partridge (1996) and used in Daga et al. (2005) to re-engineer a specific dataset of an existing corporate system implemented in different organizations. The data in this case was represented in a proprietary database language based on the traditional entity-attribute paradigm. Details of the overall approach can be found in Partridge (1996, Chapter 11). For reasons of space limitation, the following example only highlights the main phases and most relevant aspects of the methodology.

The methodology (in all its evolved versions) has two important phases, which were called Content Interpretation and Content Sophistication in Daga et al. (2005). Interpretation refers to the transformation of the original data (at both an individual and type level) into a model that conforms to the BORO foundational ontology. Sophistication refers to identifying semantic improvements in the derived BORO models. These improvements can occur either by applying existing (and previously discovered) ontological patterns and/or by identifying cases (expressed for example as competency questions) which the original model is not capable of satisfying.

The following subsections will (1) describe the original data to be semantically analyzed and transformed; (2) show how the original data model is transformed into a BORO model and (3) identify semantic improvements.

Preparation of the Legacy Data

In this example the legacy data that we begin with is on the surface quite simple and consists of an entity type called *CountryTable* with two attribute types *countryID* and *countryName*_(Figure 19). The way this entity type is implemented in a specific organization is by instantiating *CountryTable* with a chosen set of countries; this set normally corresponds to a standardized country list such as ISO 3166 or that of the Office of National Statistics (ONS) in the UK. Figure 20 shows two such example implementations in two different organizations. Both examples obviously just represent a subset of all countries recognized by both standards.

Figure 19: *CountryTable*

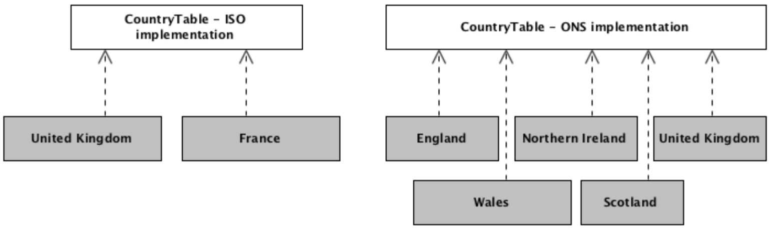

Figure 20: Two example implementations of *CountryTable*

For the purposes of this example we will not show the re-engineering of the attribute types and attributes, but only of the entity type and its instantiated entities. The interested reader can refer to Partridge (1996, Chapter 14) on the re-engineering of names and codes (a subtype of names) and the application of the BORO naming pattern.

The model in Figure 20 highlights how the designers of this legacy system had intended that organizations would or should adopt only one preferred country classification system for their business. This raises the question of how the system (and its underlying model) would cope if an organization operates in multiple countries and for legal or operational requirements necessitates the adoption of two or more country classifications. Moreover when adopting multiple classifications it is quite typical that one object (in this case the element *United Kingdom*) can be classified in different ways.

There is also another important question that arises by analyzing the individuals of the two implemented entity types and comparing them. One notices how the United Kingdom (so called home) countries are recognized in the ONS implementation. England, Northern Ireland, Scotland and Wales are officially recognized as countries in classifications such as that of the ONS. This highlights a semantic relationship between countries that is uncommon yet exists; it is the whole-part relationship. In fact, England, Northern Ireland, Scotland and Wales are all parts of the United Kingdom.

To assess the suitability of an ontological model to represent the enterprise reality, Gruninger and Fox (1995) suggest the use of competency questions. In their words, “competency questions are the benchmarks in the sense that the enterprise model is necessary and sufficient to represent the tasks specified by the competency questions and their solution”.

From the above analysis the following competency questions emerge:

- (CQ1) Can a country instantiate multiple classifications of countries?
- (CQ2) Can multiple country classification systems be represented?
- (CQ3) Can both nesting (e.g., United Kingdom) and nested countries (e.g., England, Northern Ireland, Scotland and Wales) be represented?

Content Interpretation

Interpretation is the re-engineering phase in which the original data models are semantically transformed into models that conform to the BORO foundational ontology. The approach typically proceeds bottom-up; that is starting from the original instance level entities. As Figure 20 shows these are things like the United Kingdom, France, Scotland and so on. These entities refer to individual countries in the real world. The entity type *CountryTable* (in Figure 19) instead refers to a set of recognized countries or countries that the implemented system commits to. By interpreting this data content with BORO semantics we obtain the model in Figure 21. The new model is obtained by understanding what things in the real world that *CountryTable* and its instances map to. These are the type Countries and its individual instances, which are elements.

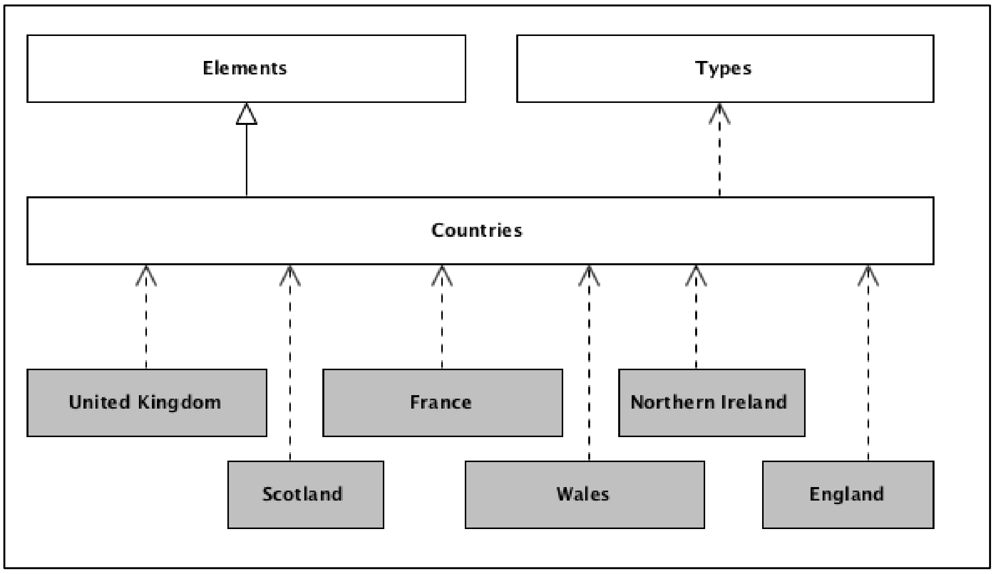

Figure 21: Grounding countries in BORO semantics

Next we interpret the two entity types of Figure 20: *Country Table – ISO implementation* and *Country Table – ONS implementation*. In BORO semantics these two entity types map to BORO types and both classify countries. On the surface it would appear that the semantics of the two implemented tables also commit to countries, however their semantics require further unbundling. In BORO the identity of types is defined by their membership. In order to understand whether the two implemented tables are ontologically one and the same, it is necessary to compare their members. It is apparent that they are not the same, for example Scotland is not an instance of *Country Table – ISO implementation*. Moreover applying the same identity test one notices that *Countries* in Figure 21 is a superset of both. The derived BORO model is shown in Figure 22.

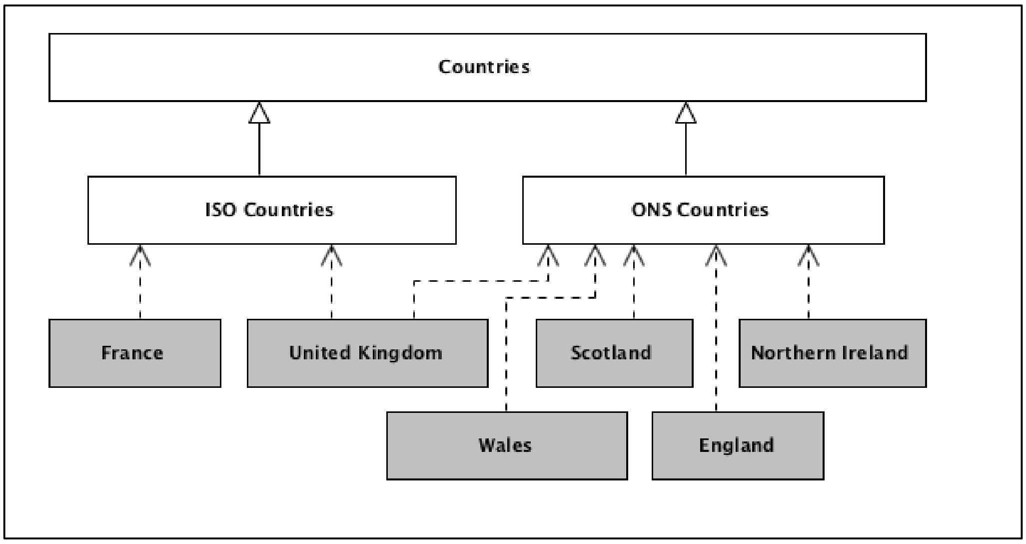

Figure 22: Subtypes of *Countries*

The ontological model in Figure 22 allows countries to be classified in both ways. One can notice how the United Kingdom is represented once while it is multiply classified by two subtypes of countries.

With the models in Figure 21 and 22 all legacy entity types and entities of Figures 19 and 20 are now mapped to and transformed into a BORO model. Interpretation is merely about transforming the legacy data semantics into BORO semantics. The next phase attempts to improve the semantics by identifying the deficiencies of the model and applying BORO patterns.

Content Sophistication

Let us apply the three competency questions to the BORO model in Figure 22.

(CQ1) Can a country instantiate multiple classifications of countries?

The new model appears to be capable of now representing multiple country subtypes. In fact, one can represent further subtypes of *Countries*, for example *BoE Countries* (or countries recognized by the Bank of England) (see Figure 23). As discussed above since multiple classification is possible in BORO, a country can instantiate more than one of these subtypes.

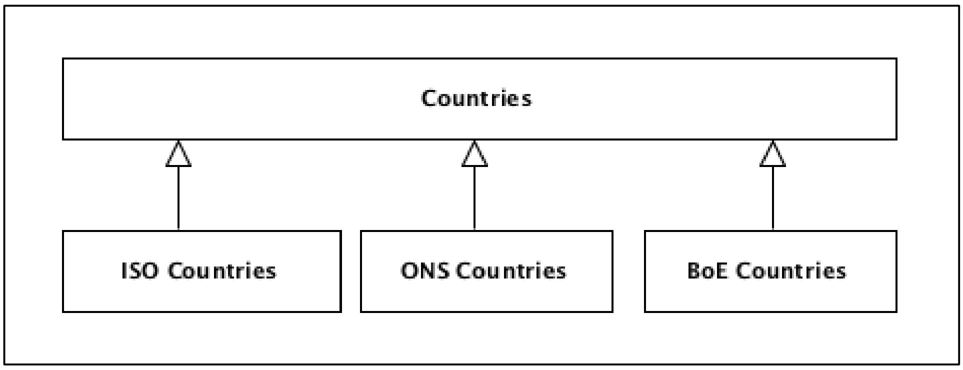

Figure 23: Adding a new country subtype

While Figure 23 is semantically correct and it implicitly allows for multiple country subtypes to be represented, the model does not explicitly unbundle all of the underlying semantics. While at this stage one knows that the subtypes of Figure 23 all instantiate BORO *Types*, the model as it stands is not capable of saying more about the ways in which countries can be classified.

(CQ2) Can multiple country classification systems be represented?

In Section 2 BORO powertypes were introduced and in Section 10 will be applied to the classification of business activities based on the UK Standard Industry Classification system. The example in Section 10 is an example of how the BORO pattern for classification systems is applied and it makes use of powertypes. In order to improve the interpreted model in Figure 23 with regards to (CQ2), the BORO classification system pattern is applied. This is represented in Figure 24.

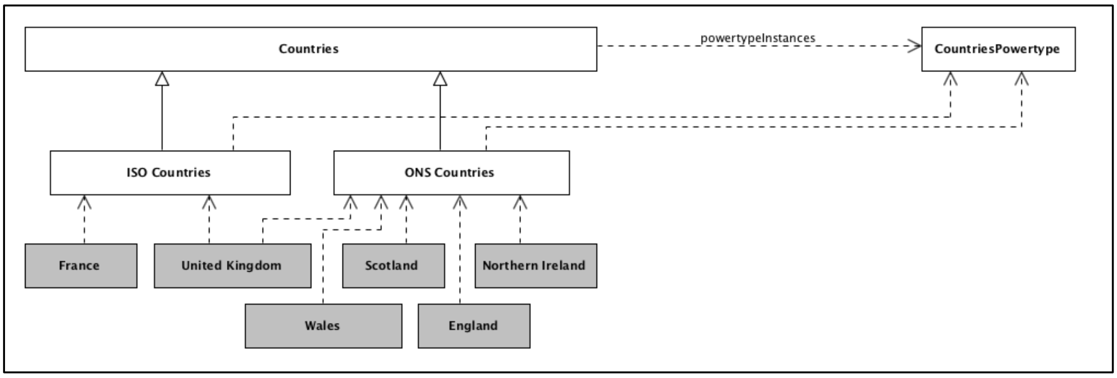

Figure 24: Countries Powertype

The model in Figure 24 provides the semantics to describe the classifications systems themselves. For example, ONS and BoE classifications are UK classifications. Moreover while the examples used here are country classifications of nationally or internationally recognized institutions, a business organization may have its own internal country classification system(s). Figure 25 shows how the pattern adopted in Figure 24 enables us to represent these cases. In Figure 25 all subtypes of *Countries* are instances of *CountriesPowertype*. This occurs by default since BORO's powertype is set theoretic. This makes it possible to not only explicitly represent the many classifications as subtypes of *Countries* and instances of the powertype, but also to classify the classifications themselves. In Figure 25, *ISO Countries* is classified as an international classification, *ONS Countries* and *BoE Countries* are classified as UK classifications, and ACME's internal classification scheme is classified as an organizational classification.

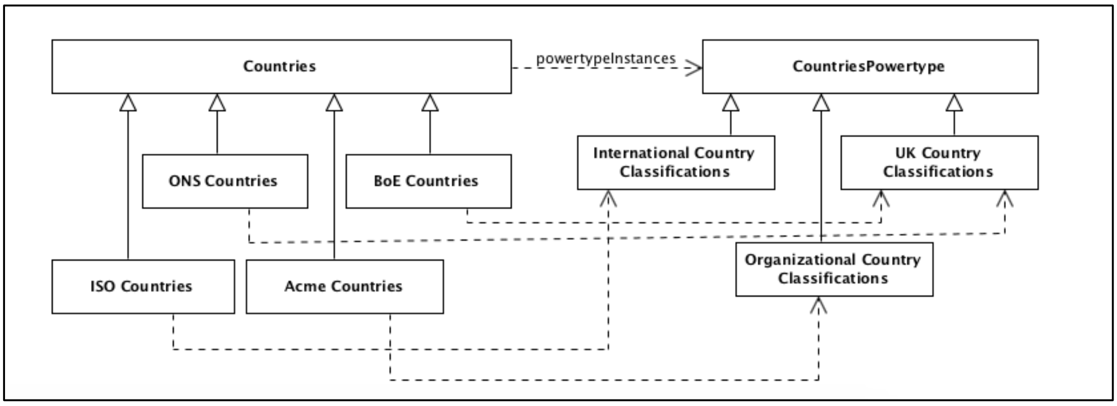

Figure 25: Explicit representation of country classification systems and their groupings

(CQ3) Can both nesting (e.g., United Kingdom) and nested countries (e.g., England, Northern Ireland, Scotland and Wales) be represented?

The third competency question recognizes that countries can be parts of other countries as is the case of the United Kingdom and its home countries. The improvement that BORO semantics provides to the model in Figure 22 occurs by applying the whole-part pattern of Figure 4 to countries. More generally this pattern is applicable to Geopolitical Regions (and discussed in Partridge (1996, Chapter 16)). The semantically improved model is shown in Figure 26.

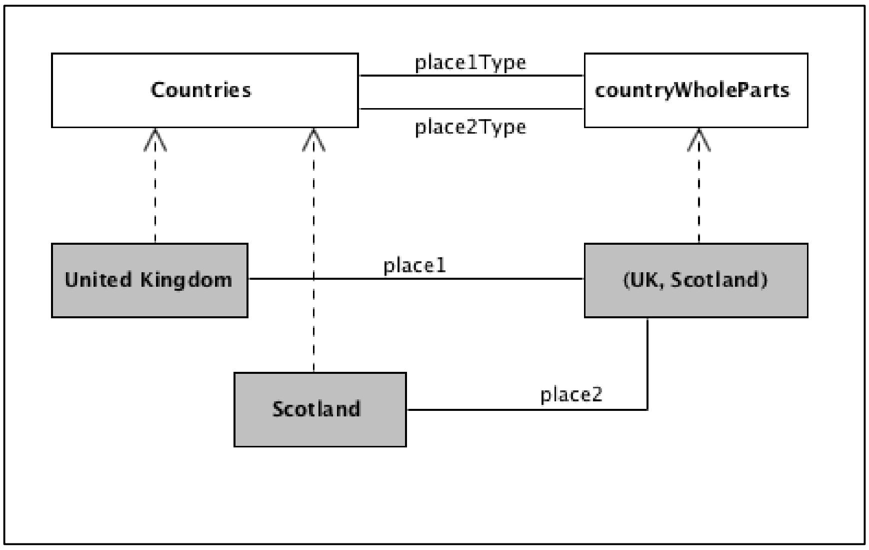

Figure 26: Whole-part relationships between countries

The transitivity of the whole-part tuple type simplifies the representation while at the same time being operationally more fruitful. For example, if a product or service is sold in Scotland, by means of this transitivity, the product will automatically be considered by the enterprise system as being sold in the United Kingdom.

Figure 27 illustrates the final sophisticated model and its grounding in the BORO foundational ontology. The evaluation of this re-engineering example will be discussed in Section 9.

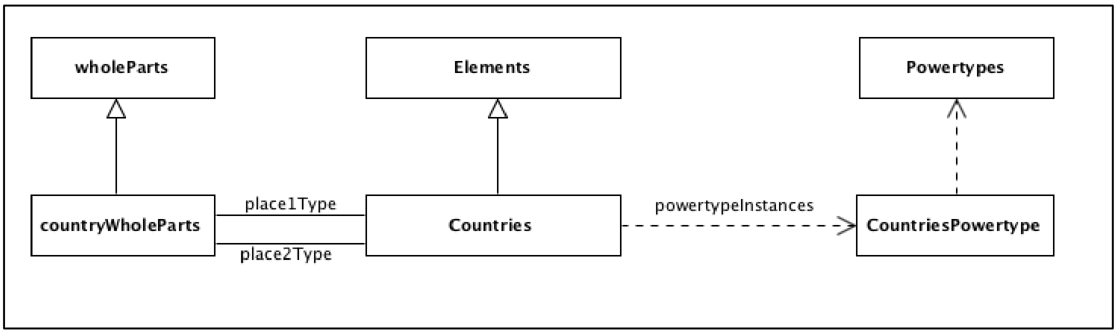

Figure 27: Final semantically improved BORO model for countries.

The example presented above shows how the BORO approach is capable of discovering new general business patterns via the semantic interpretation and sophistication of existing legacy enterprise data. The improved models can identify new patterns that feed into future legacy re-engineering projects aimed at semantic improvement. It should be noted that whenever a new general business pattern is discovered, the BORO process is akin to what occurs in Action Design Research.

7. Specification and Representation Formalism

The notation adopted to represent BORO has evolved over time. The original notation is presented in Partridge (1996). An example extracted from the book is illustrated in Figure 28 in relation to the naming of countries.

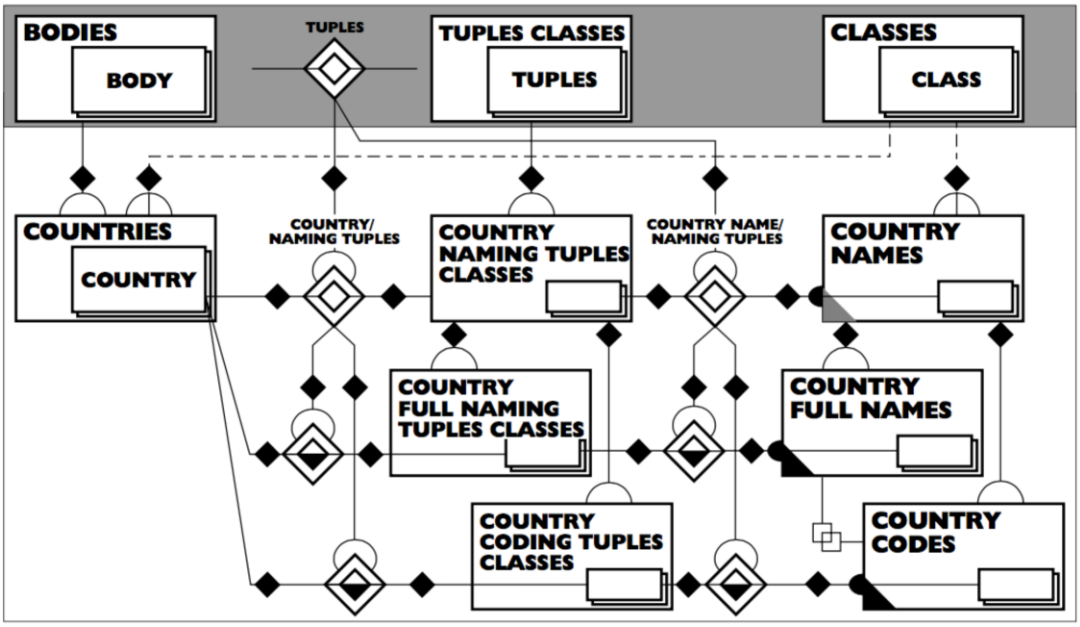

Figure 28: Example of the original BORO notation

The notation was specifically designed to represent the BORO models. Shortly after the publication of *Business Objects* (Partridge, 1996), the UML (OMG, 2015) emerged and became widespread in both industry and academia. Around the beginning of the 2000s a new BORO notation emerged which was based on the UML. This new notation was adopted within the Semantic Integration Environment (SITE) research project (Paul and Macredie, 2001) at Brunel University London and was evolved over the following years in projects like the IDEAS group (IDEAS, 2016). The IDEAS notation is explained and found at http://www.ideasgroup.org/foundation/.

Since BORO-UML (or BUML) integrates UML notational semantics with BORO's real world semantics, a person approaching BORO for the first time, and knowledgeable in UML, would be able to learn the notation much more quickly than the one exemplified in Figure 28. Currently BUML is the notation normally adopted in BORO projects.

The notation used in this paper is the same as the one adopted by the IDEAS group apart from some minor differences: (1) In IDEAS tuples are represented as diamonds, while we have chosen to represent them as rectangles; both conform to UML notational semantics; (2) IDEAS makes broad use of stereotypes indicating the foundational object that the object represented instantiates; (3) IDEAS adopts color to represent different stereotypes. We have not adopted the latter two (stereotypes and color) simply to avoid cluttering the diagrams in the limited space available and to provide readers with more legible diagrams when printed. It must be noted that in BORO the same modeling language/notation is used for enterprise modeling and for other purposes.

BORO models are normally presented as visual diagrams with the notation explained above. This type of representation has proven to be the most effective for representing the model and the discovered ontological patterns. This is primarily due to the way visual models can more easily communicate the representations among groups of people working together as opposed to more mathematical and logical representations which are more common in fields such as artificial intelligence or formal verification and validation. For data manipulation, the model is typically stored in a database (Partridge 1996). Tools have been developed to migrate the model from a UML Repository to the database and back.

Normally visual models, like UML models, have traditionally been considered to be semi-formal, in other words it is assumed that they do not ensure the same level of formal consistency that, for example, representations based on logic (e.g., first-order logic) and formal semantics do. According to some this is also true of the UML whose semantics and formal structures exhibit areas of incompleteness and inconsistency. For the representation of ontologies both real-world (or material) semantics and formal consistency are necessary. While the UML may per se not provide the necessary support, the UML notation enriched with BORO semantics does. Therefore, BUML is the predominant formalism used for coding the foundational ontology and its generalized business patterns.

As explained in Sections 2 and 3, from a formal/theoretical perspective, BORO is extensional with extensional criteria of identity. Elements (i.e. particulars) are 4D extensional drawing theoretically upon Lewis (1986) and Sider (2005) as well as from classical mereology (Simons, 1987; Varzi 2007). Type extensionality theoretically draws upon Set Theory and more specifically its axiom of extensionality (Lewis, 1986).

8. Evolution

As noted earlier, BORO emerged in the late 1980s from a legacy re-engineering project as a closely intertwined foundational ontology and re-engineering methodology. This is reflected in the original name for the approach (that combined ontology and methodology) REV-ENG: an acronym for Reverse Engineering. In the early days, the original modeling team who were defining an early version of the approach thought that they were just reverse engineering enterprise systems. They later realized however that the activity involved much more; the extraction and modeling of business content from the legacy systems (reverse engineering) plus the forward engineering of improved semantic models. However the name REV-ENG was established and so it remained; it is used by Partridge (1996).

REV-ENG was later adopted in the Semantic Integration Environment research project (Paul and Macredie, 2001) at Brunel University London from 2000-2003 and it evolved for the purposes of that project into an approach named Content Sophistication (Daga et al., 2005). The approach is currently named BORO and it has evolved further within many industrial and commercial projects since 2000.

The development of the BORO approach has been ongoing for a number of decades. Historically the core foundation was originally developed by a team of KPMG consultants working in the late 1980s and early 1990s primarily in relation to the development and legacy re-engineering of enterprise systems (these projects are described by Partridge (1996, pp. xiii-xiv)). In the early 1990s, the team working on EPISTLE (European Process Industries STEP Technical Liaison Executive) became aware of this work and amended their data model to accommodate 4D extensional elements. This was standardized as 'ISO 15926: Part 2' and it has been built upon in various ways, including the work of West (2010). Sections 10 and 11 will provide further details related to the application of BORO and other standardization efforts.

The original foundational ontology has stood the test of time and BORO has not experienced any substantial formal changes since it was originally and systematically described by Partridge (1996). However, as a result of the initiatives noted earlier in the defense sector, numerous industrial projects (primarily in the financial and oil and gas sectors) and publicly funded research projects, BORO has evolved in the following directions:

- In the original version of BORO there was a greater emphasis on *Events* as fundamental foundational objects. Partridge (1996) distinguished between *Physical Bodies* (including *States*) and *Physical Events*. The former are temporally extended 4D extents. The latter are objects with no temporal extent (temporal slices with zero thickness of the 4D universe), so typically of 3D extent. Events were regarded as important as they marked the boundaries of physical bodies and, with states, contributed to explaining change (Partridge, 1996, Chapter 8). Events, in the sense explained here, are now regarded as less fundamental in BORO. One reason is that the notion of temporal boundaries is not so clearly equivalent to zero temporal extent in practice, examples including the signing of contracts and the foundation of a company, which are more accurately regarded as not happening at a specific time instant but over a time period (however brief it may be). Hence the evolution toward the ontology represented in Figure 4. It must be noted that 3D events have not been deprecated and they can be validly used where needed – for example, where one needed to model exactly when the contract was signed.
- Currently the original terms Things and Classes are more widely known as Elements (or Individuals as in IDEAS (2016)) and Types respectively. In IDEAS (2016) and Partridge (1996) Objects are called Things. While the current usage tends to not use the words Things and Classes, these words can validly name their respective objects.
- The original notation used to represent the BORO ontology was designed by Partridge (1996) and described in his original book (chapters 9 and 10). With the introduction of the Unified Modeling Language (UML) among the software engineering and conceptual modeling communities at the end of the 1990s and its widespread industrial adoption in the 2000s, as part of the Semantic Integration Environment research project (mentioned earlier) a new BORO notation more aligned with UML class diagrams was developed. Currently, as explained in Section 7, the notation adopted to represent BORO models is a variation of UML class diagrams based upon BORO semantics.
- The most significant evolution has occurred in relation to the discovery and representation of further and numerous domain patterns. Examples of these ontological patterns include, for example, services (Partridge and Bailey, 2010) and capabilities (Hagenbo et al., 2012).

On the whole the BORO foundational ontology has remained quite stable. The primary reason for this stability is its strong grounding in philosophical ontology with the metaontological choices described in Section 5; choices rooted in theories of reality (such as realism, perdurantism, extensionalism of individuals/types, mereology and possible worlds), which have been extensively studied and debated over decades (or in some cases centuries or millennia) resulting in solid ontological theories. As explained in Section 9, since the generalized business patterns are empirically derived and tested against new datasets over time and over the course of several industrial projects, the models publicly available in the published literature and reference architectures are relatively stable and mature.

9. Evaluation

In BORO, similarly to the discovery/elicitation of generalized business patterns, evaluation of these ontological models is also systematic, grounded and empirical and it is embedded within the BORO re-engineering method presented in section 6. As the example in Section 6 demonstrated, during the phases of interpretation and sophistication, the new models are based on and embed the BORO foundational patterns and existing generalized business patterns. Both sets of patterns are tested over numerous projects. This means that evaluation in BORO is continuous. Ontological models are continuously reused and, as a consequence, are applied to new datasets and new competency questions. Any semantic flaws in the existing ontological patterns would be discovered by this continuous 'stress testing'.

This approach to semantic evaluation is long-term and continuous; quite different from the approaches typically found in the literature. The literature on ontology evaluation normally refers to the adoption of certain criteria and metrics. These can include, for example, consistency, completeness, conciseness, expandability and sensitiveness (Gómez-Pérez, 2004). While these criteria are appropriate and good indicators of a well-developed ontology, most of the literature (mainly on Semantic Web ontology evaluation) tends to implicitly assume that the application of such criteria is sufficient to assess an ontological model and do not emphasize the benefits of evaluating the models against new enterprise data over the long period.

In Daga et al. (2005) a set of sophistication dimensions was proposed in order to evaluate the ontological models derived from the re-engineering of legacy enterprise data. These dimensions are borrowed from Kuhn's (1962) book titled *The Structure of Scientific Revolutions*. In the context of ontological models Kuhn's definitions are adapted as follows (Daga et al., 2005):

- Generality: The degree by which the scope of the types in the improved model can be increased without the loss of information.

- Simplicity: The degree by which the model can be made less complex.
- Explanatory power: The ability of the improved model can give increased meaning to the objects and the relationships expressed.
- Fruitfulness: The degree to which the improved model can meet currently unspecified requirements or is easily extendable to do so.
- Objectivity: The ability of the model to provide a more objective (shared) understanding of the world.
- Precision: The ability of the improved model to give a more precise picture of the business object: in particular, to index a thing to its mode of existence as opposed to its mode of representation and/or application.

Table 2 exemplifies how these dimensions are applied to the re-engineered models of Section 6.

Dimension	Description
Generality	The new model is more general than the legacy model since it allows for multiple country classifications and mereological relationships between countries.
Simplicity	At the type level the new model contains a small set of objects: two types (Countries and its powertype) and two tuple types (country whole-parts and the powertype instance relation). At the individual element level the new model represents each country once (via multiple instantiation) rather than multiple times for each implemented classification as in the legacy system.
Explanatory power	The new model provides a set theoretic definition of country classifications and it is capable of explaining what country classification systems are as well as provide the support to classify the classifications themselves.
Fruitfulness	The new model provides support for extended functionality in terms of providing the necessary semantics to ‘know’ that if an event (e.g., a sale) occurs in a nested country then it also occurs in its nesting country.
Objectivity	The new model is not dependent on a specific system implementation or an organization’s perspective since multiple classifications of countries are now possible within the same enterprise system.
Precision	The new model is a more accurate representation of countries since it is able to represent country whole-part relationships.

Table 2: Semantic evaluation of the re-engineered model

10. Applications

The main areas of application for BORO have been in legacy re-engineering of enterprise systems, Enterprise Architecture and the development of standards for data interoperability. BORO has been adopted in both the industrial/commercial and research contexts. From its initial application in the financial sector at KPMG where it was adopted for legacy re-engineering, BORO has then been successful in resolving data integration problems in the oil and gas sector and integration/interoperability of enterprise systems in the defense industry. As discussed in Section 3 various BORO-based standards have been developed or are being developed from the applications mentioned here.

In a research context BORO has been adopted in three projects funded by the UK Engineering and Physical Sciences Research Council. These projects are: Semantic Integration Environments (Paul and Macredie, 2001), Empirical Modeling of Business Process Patterns with Ontologies (de Cesare et al., 2013) and Semantic Credit Risk Assessment in Business Ecosystems (Lycett et al., 2014). The latter two projects are ongoing. All three research projects have investigated or are investigating problems related to Enterprise Modeling.

An application of BORO is exemplified by de Cesare et al. (2013). In this work the authors present research aimed at re-engineering open company datasets into ontological models for the purpose of data integration. In this research one of the re-engineered datasets was about classification of business activities. This example is presented in Figures 29 to 31 and it demonstrates a typical application of the powertype pattern. The complete pattern, with its formalization, is presented in Partridge et al. (2016).

Powertypes are essential in the representation of classification systems, which are ubiquitous in the enterprise domain (as they are in all domains). Examples include product classifications (e.g., car makes and models), process classifications (e.g., by product, by location or by raw material), employee classifications (e.g., by skillset, by hierarchical level or by organizational unit), and business activity classifications (e.g., national Standard Industry Classifications (SIC)). The instances of a powertype (P(A)) are in effect all possible ways in which the instances of the original type (A) can be classified.

Figures 29-31 demonstrate how powertypes are applied in BORO to represent the UK Standard Industry Classification (SIC) 2007 (ONS, 2007). SIC 2007 is a classification system based on taxonomic ranks similarly to the Linnaean classification system in biology (Partridge et al., 2015). This further supports the assertion that ontological patterns can be used across domains that on the surface may appear completely different. The ranks in SIC 2007 are sections, divisions, groups, classes and subclasses (not all classes have subclasses). Table 3 presents the SIC 2007 ranks with a specific example.

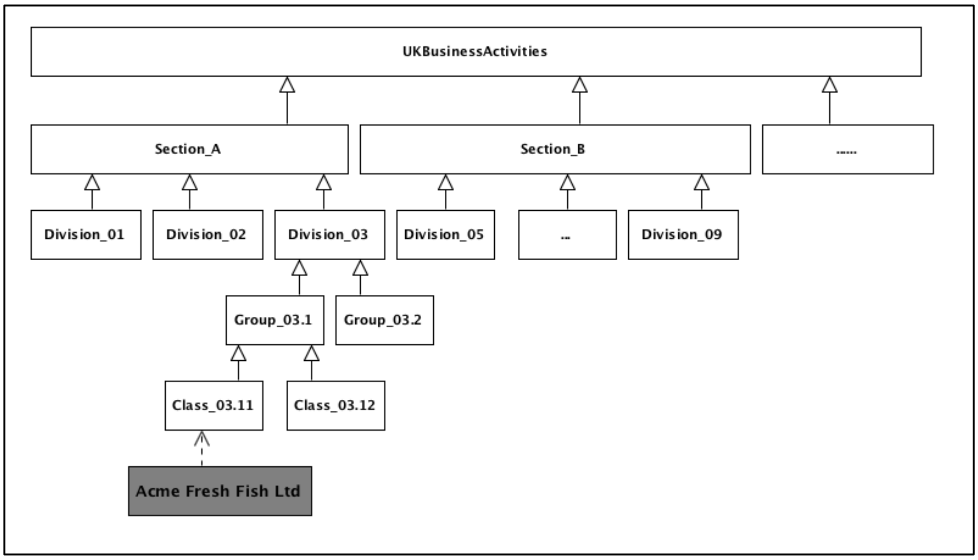

Figure 29. Example of SIC 2007 taxonomy

SIC 2007		
Rank	***Specific Instance***	***Description***
Section	Section C	Manufacturing
Division	Division 13	Manufacture of textiles
Group	Group 13.9	Manufacture of other textiles
Class	Class 13.93	Manufacture of carpets and rugs
Subclass	Subclass 13.93/1	Manufacture of woven or tufted carpets and rugs

Table 3. An example of SIC 2007 ranks

These ranks define successive taxonomic levels in the hierarchy of business activities according to this classification system. *Sections* are subtypes of *UKBusinessActivities*; *Divisions* are subtypes of S*ections* and so on. Figure 29 illustrates the SIC taxonomy where each subtype corresponds to an instance of a SIC 2007 rank. This instantiation relationship between types is more clearly shown in Figure 30. Each rank (or taxonomic level) is a subtype of the powertype of *UKBusinessActivities*. This means that a SIC 2007 rank like *SIC2007_Sections* is a second order type (or type of types).

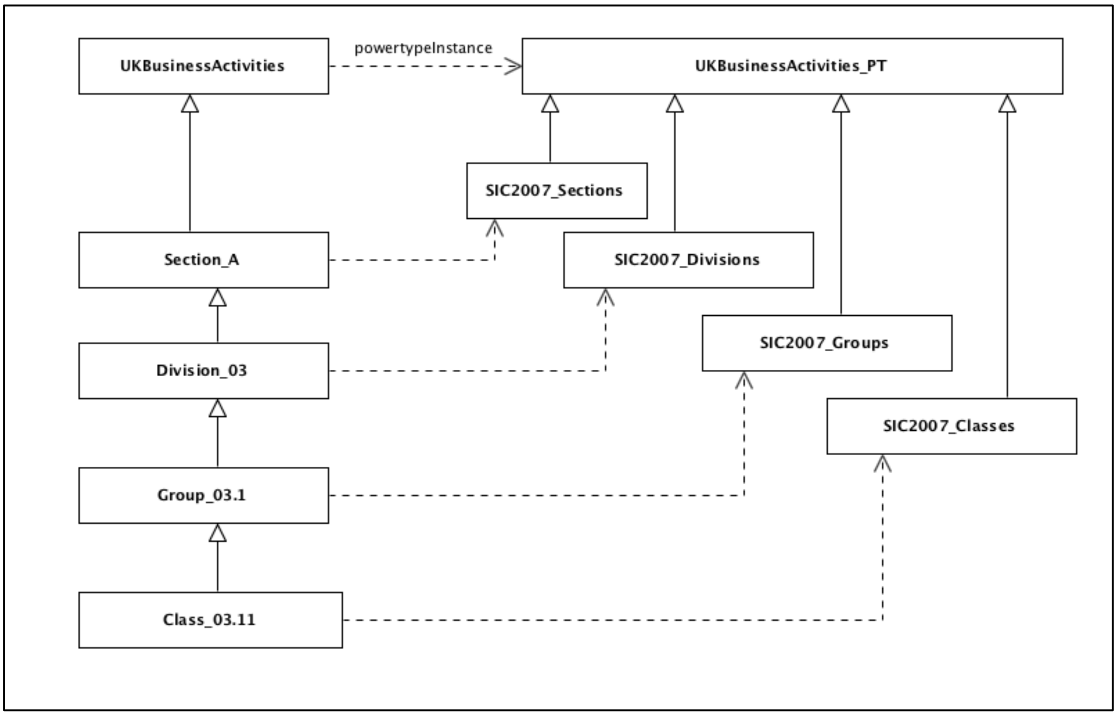

Figure 30. Representation of SIC 2007 taxonomic ranks

The taxonomic ranks (Figure 31) in turn are instances of *SIC2007_Ranks* which is a subtype of *UKBusinessActivities_PT_PT* (or the powertype of the powertype of UK business activities). *SIC2007_Ranks* is a third-order type (or types of types of types).

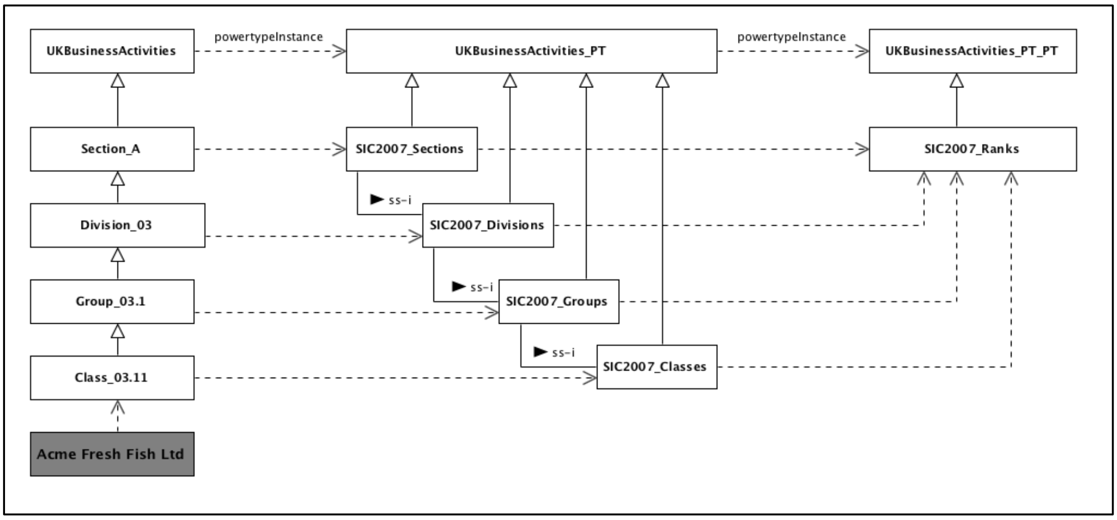

Figure 31. Representation of SIC 2007 Ranks as a type

As the example above demonstrates, BORO types (and powertypes) work in a similar way to classical set theory in which some key aspects are: (1) a set can be any arbitrary collection of things; (2) sets are immutable; (3) powertypes as powersets generate higher-order types.

With regards to the latter it is useful to note that in the conceptual modeling literature there are a variety of approaches to second-order types. For example, in Henderson-Sellers and Gonzalez-Perez (2005) the authors explain higher order types using two types of things rather than just one, whereby a type like *CarModels* (or *SIC2007_Sections* above) would be considered as a container which they call a clabject and two related 'facets', a class facet and an object facet. In BORO *CarModels* is just one object.

11. Who, Where, When

The original BORO foundational ontology is described in Partridge (1996) and republished in 2005 (Partridge, 2005). Figure 32 illustrates the timeline of the development of the BORO ontology and initiatives that have adopted BORO as the underpinning ontology (including ISO 15926). The results of the earlier work were documented in Partridge (1996) and in a series of further papers by Daga et al. (2004, 2005), Lycett and Partidge (2009), Partridge (1994), and Partridge and Stefanova (2001).

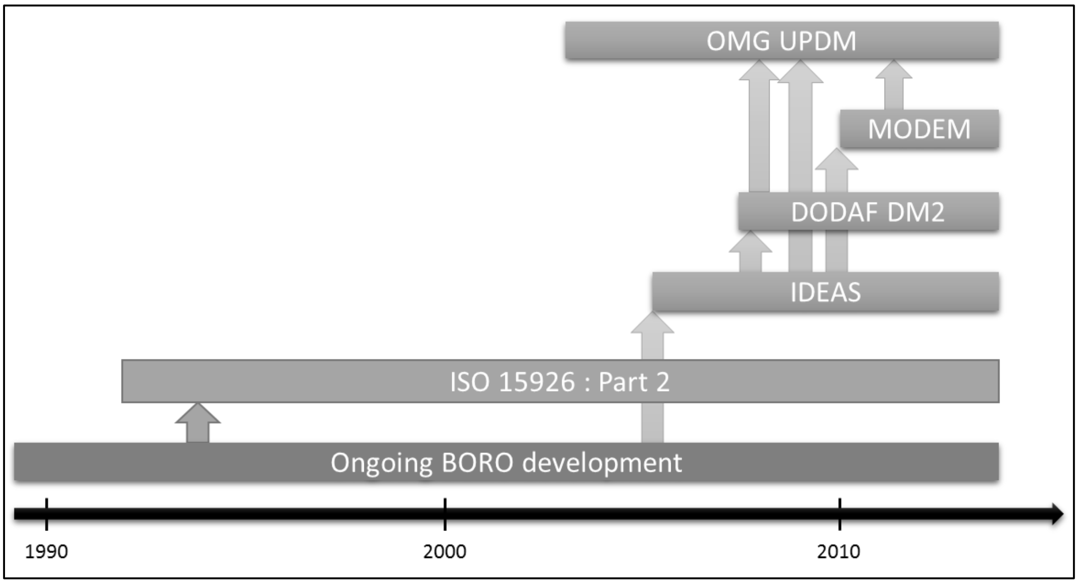

Figure 32: Chronology of BORO development and related initiatives

More recently, BORO has been adopted by the IDEAS Group (www.IDEASGroup.org) for the purpose of developing a data exchange format for military Enterprise Architectures. Subsequently, BORO was also adopted to develop enterprise architecture frameworks in the United States (DODAF 2.0) and the United Kingdom and Sweden (MODEM). In parallel with

IDEAS, DODAF 2.0 and MODEM, there has been also an effort to standardize these initiatives with the OMG as the Unified Profile for DoDAF/MODAF (UPDM).

12. Conclusion

BORO was initially conceived in the late 1980s as a means to re-engineer the content of legacy enterprise systems. Over the past 30 years BORO has been applied to other enterprise related areas in different industrial sectors and is at the basis of standards for data interoperability and exchange, systems integration and enterprise architecture. The BORO foundational ontology has remained relatively unchanged due to its roots in solid and consolidated metaphysical theories of reality. The BORO methodology, driven by the foundational ontology, has been adopted to discover numerous generalized business patterns over the years. These patterns have been evaluated over the long term with empirical enterprise data of numerous industrial projects. This empirical and continuous type of evaluation has made the patterns quite stable and reusable across multiple domains and organizations.

Current and future research on BORO will focus on its adoption in other industrial sectors, such as energy and power, the re-factoring of industry-driven information model standards and, as mentioned previously, further investigation into BORO's adoption in the modeling of credit risk, business ecosystems and business process management.

Acknowledgements

The work was supported by the UK Engineering and Physical Sciences Research Council (grant EP/K009923/1). We would also like to thank the editor of this special issue Professor Guido L. Geerts and the anonymous reviewers for their valuable comments.

References

Berto, F., and M. Plebani. 2015. Ontology and Metaontology. Bloomsbury.

Boolos, G. 1971. The iterative conception of set. Journal of Philosophy 68 (8): 215-231.

Daga, A., S. de Cesare, M. Lycett, and C. Partridge. 2005. An Ontological Approach for Recovering Legacy Business Content. 38th Hawaii International Conference on System Sciences. IEEE Computer Society Press.

Daga, A., S. de Cesare, M. Lycett, and C. Partridge. 2004. Software Stability: Recovering General Patterns of Business. Tenth Americas Conference on Information Systems 4278-4285.

de Cesare, S., G. Foy, and C. Partridge. 2013. Re-engineering Data with 4D Ontologies and Graph Databases. CAiSE 2013 Workshops, Lecture Notes in Business Information Processing 148:304-316. Springer-Verlag.

de Cesare, S. and Geerts, G.L. (2012). Toward a Perdurantist Ontology of Contracts. M. Bajec and J. Eder (Eds.): CAiSE 2012 Workshops, LNBIP 112, 85–96, Springer-Verlag Berlin Heidelberg.

de Cesare, S., B. Henderson-Sellers, C. Partridge, and M. Lycett. 2015. Improving Model Quality through Foundational Ontologies: Two Contrasting Approaches to the Representation of Roles. Advances in Conceptual Modelling - ER 2015 Workshops. Editors: Manfred Jeusfeld and Kamalakar Karlapalem. Springer.

de Cesare, S., D. Juric, and M. Lycett. 2016. Automated taxonomy extraction from semantic business process models. 49th Hawaii International Conference on System Sciences.

de Cesare, S. and M. Lycett. 2013. Empirical Modelling of Business Process Patterns with Ontologies, Engineering and Physical Sciences Research Council. Available at: http://gow.epsrc.ac.uk/ NGBOViewGrant.aspx?GrantRef= EP/K009923/1.

Fine, K. 2010. Towards a Theory of Part. Journal of Philosophy 107 (11):559-589

Geerts, G.L. 2016. ODAF: A framework for documenting and analyzing enterprise ontologies. Available at: http://www.aisvillage.com/ODAF/odaf.pdf

Geerts, G.L., and W. McCarthy. 2002. An ontological analysis of the economic primitives of the extended-REA enterprise information architecture. International Journal of Accounting Information Systems 3(1): 1-16.

Gómez-Pérez, A. 2004. Ontology Evaluation. In Handbook on Ontologies, eds. Staab, S. and Studer R., pp. 251-273. Springer.

Grüninger, M., Fox, M.S. 1995. The Role of Competency Questions in Enterprise Engineering. Benchmarkin - Theory and Practice. 22-31. Springer.

Guizzardi, G. 2005. Ontological Foundations for Structural Conceptual Models, Telematica Instituut Fundamental Research Series No. 15 (PhD thesis).

Guizzardi, G. and G. Wagner. 2004. A Unified Foundational Ontology and some Applications of it in Business Modeling, CAiSE Workshops (3).

Hagenbo, M., L.O. Kihlström, C. Partridge, and P. Gorman. 2012 MODEM - Re-engineering the MODAF meta-model based on the IDEAS foundation model. Available at: http://bit.ly/1FDr7sI

Henderson-Sellers, B., and C. Gonzalez-Perez. 2005. The rationale of powertype-based metamodelling to underpin software development methodologies. Proceedings of the 2nd Asia-Pacific Conference on Conceptual Modelling. 43:7-16. Australian Computer Society, Inc.

Hofweber, T. 2014. Logic and Ontology. The Stanford Encyclopedia of Philosophy. Edward N. Zalta (ed.), Available at: http://plato.stanford.edu/archives/fall2014/ entries/logic-ontology/

Honderich, T. 2005. The Oxford Companion to Philosophy. Cambridge University Press, Cambridge.

IDEAS. 2016 International Defence Enterprise Architecture Specification for exchange. Available at: http://www.ideasgroup.org/foundation/

Kuhn, T. 1962. The Structure of Scientific Revolutions. The University of Chicago Press.

Lewis, D. 1986. On the Plurality of Worlds. B. Blackwell, New York, NY.

Lycett, M., S. de Cesare, F. Moscone, and J. Marriott. 2014. Semantic Credit Risk Assessment of Business Ecosystems, Engineering and Physical Sciences Research Council. Available at: http://gow.epsrc. ac.uk/NGBOViewGrant.aspx?GrantRef=EP/L021250/1.

Lycett, M., and Partridge, C. 2009. The challenge of epistemic divergence in IS development. Communications of the ACM 52(6): 127-131.

McDaniel, D. 2012. NDIA 2012 - NEW 15271 - DOD Architectures and Systems Engineering Integration. Available at: http://www.dtic.mil/ndia/2012system/track215271.pdf

Office for National Statistics, UK Standard Industrial Classification 2007 (UK SIC 2007). Available at: http://www.ons.gov.uk/ons/guide- method/classifications/current-standard-classifications/standard- industrial-classification/index.html

OMG. 2015. OMG Unified Modeling Language TM (OMG UML). Version 2.5. Available at: http://www.omg.org/spec/UML/2.5/PDF/

OMG. 2015. Unified Profile for DoDAF and MODAF (UPDM) - Version 2. Available at: http://www.omg. org/ spec/UPDM/2.1/PDF/

Partidge, C., and Bailey, I. 2010. An Analysis of Services. Model Futures. Available at: http://bit.ly/1KOxfLz

Partridge, C. 1994. Modelling the real world: Are classes abstractions or objects? Journal of Object Oriented Programming 7(7).

Partridge, C. 1996. Business Objects: Re-Engineering for Re-Use, Butterworth-Heinemann.

Partridge, C. 2002a What is Pump Facility PF101? A Study in Ontology? Technical Report 04/02, LADSEB-CNR, Padova, Italy.

Partridge, C. 2002b A new foundation for accounting: Steps towards the development of a reference ontology for accounting. Technical Report 23/02, LADSEB-CNR Padova, Italy.

Partridge, C. 2002c. A couple of meta-ontological choices for ontological architectures. Technical Report 06/02, LADSEB-CNR Padova, Italy.

Partridge, C. 2005. Business Objects: Re-Engineering for Re-Use, The BORO Centre.

Partridge, C., A. Mitchell, and S. de Cesare. 2012. Guidelines for Developing Ontological Architectures in Modelling and Simulation. A. Tolk (Ed.): Ontology, Epistemology, and Teleology of Modeling and Simulation: Philosophical Principles for Intelligent M&S Applications. 27-57, Springer-Verlag, Germany.

Partridge, C., and M. Stefanova. 2001 A synthesis of state of the art enterprise ontologies: Lessons learned. In: Open enterprise solutions: Systems, experiences, and organizations (OES-SEO 2001). 130-133. Luiss Edizioni, Centro di Ricerca sui Sistemi Informativi.

Partridge, C., S. de Cesare, A. Mitchell, J. Odell. 2016. Formalization of the Classification Pattern: Survey of Classification Modeling in Information Systems Engineering. Software and Systems Modelling (forthcoming).

Paul, R. and R. Macredie. 2001. Semantic Integration Environment (SITE). Engineering and Physical Sciences Research Council. Available at: http://gow.epsrc.ac.uk/NGBOViewGrant.aspx?GrantRef=GR/N01897/01

Quine, W.V.O. 1948. On What There Is. Review of Metaphysics.

Schaffer, J. 2009. Spacetime the one substance. Philosophical Studies 145(1): 131–148.

Schaffer, J. 2010. Monism: The Priority of the Whole. Philosophical Review 119 (1): 31-76

Searle, J.R. 1995. The construction of social reality. Free Press, New York.

Sider, T. 2005. Four-Dimensionalism: An Ontology of Persistence and Time. Oxford University Press.

Simons, P. 1987. Parts: A Study in Ontology. Oxford University Press.

Sklar, L. 1974. Space, Time, and Spacetime. University of California Press.

Smith, B. 2004. Beyond Concepts: Ontology as Reality Representation. Conference on Formal Ontology and Information Systems, Amsterdam, 73–84. IOS Press.

Van Inwagen, P. 1998. Meta-Ontology. Erkenntnis 48:233-50.

Varzi, A.C. 2007. Spatial Reasoning and Ontology: Parts, Wholes, and Locations. In Aiello, M. et al. (eds.) Handbook of Spatial Logics. Springer-Verlag: 945-1038.

Wand, Y., and R. Weber. 1993. On the ontological expressiveness of information systems analysis and design grammars. Information Systems Journal 3:217-237.

West, M. 2010. Developing High Quality Data Models. Morgan Kaufmann.

Wiggins, D. 2001. Sameness and Substance. Cambridge University Press.

Zermelo, E. 1908. Untersuchungen über die grundlagen der mengenlehre. I. Mathematische Annalen. 65(2): 261-281.