4D-SETL
A Semantic Data Integration Framework

Sergio de Cesare[1], George Foy[2] and Mark Lycett[2]
[1]*College of Business, Arts and Social Science, Brunel University London, Uxbridge, U.K.*
[2]*College of Engineering, Design and Physical Sciences, Brunel University London, Uxbridge, U.K.*

Keywords: Foundational Ontology, Perdurantist, 4D, Semantic Data Integration, Modelling, Graph Databases, Integration Frameworks.

Abstract: Although successfully employed as the foundation for a number of large-scale government and energy industry projects, foundational ontologies have not been widely adopted within mainstream Enterprise Systems (ES) data integration practice. However, as the closed-worlds of ES are opened to Internet scale data sources, there is an emerging need to better understand the semantics of such data and how they can be integrated. Foundational ontologies can help establish this understanding and therefore, there is a need to investigate how such ontologies can be applied to underpin practical ES integration solutions. This paper describes research undertaken to assess the effectiveness of such an approach through the development and application of the 4D-Semantic Extract Transform Load (4D-SETL) framework. 4D-SETL was employed to integrate a number of large scale datasets and to persist the resultant ontology within a prototype warehouse based on a graph database. The advantages of the approach included the ability to combine foundational, domain and instance level ontological objects within a single coherent system. Furthermore, the approach provided a clear means of establishing and maintaining the identity of domain objects as their constituent spatiotemporal parts unfolded over time, enabling process and static data to be combined within a single model.

1 INTRODUCTION

An enterprise may acquire data from many sources in many different forms (Zikopoulos and Eaton, 2011). Key considerations in integrating such data include dealing with the diversity of representation and the interpretation of the inherent explicit and implicit semantics. The latter of these considerations is particularly important in the context of ES integration as, if left unrecognised, it can lead to the things of importance (e.g., domain objects and their relationships), their nuances and the state of affairs they represent being misinterpreted (Lycett, 2013). These considerations are well recognised within database integration projects (Arsanjani, 2002; Campbell and Shapiro, 1995; Sheth and Larson, 1990).

Ontology has emerged as a promising way of dealing with such diversity, however many popular domain ontologies have no grounding in a consistent foundational view of reality (Cregan, 2007) and therefore can add further diversity. A foundational ontology can be employed to provide this 'grounded' view of reality and thus provide an explicit theory and a common reference through which to interpret, model and thus integrate data. Foundational ontology "defines a range of top-level domain-independent ontological categories, which form a general foundation for more elaborated domain-specific ontologies" (Guizzardi *et al.* 2008). From a philosophical perspective, foundational ontologies provide the criteria for ontological commitments – statements on the things believed to exist within the context of a particular theory (Bricker, 2014). Several foundational ontologies currently exist (Gangemi *et al.,* 2002; Grenon and Smith 2004; Partridge 2005; Guizzardi, *et al.,* 2008; Herre 2010) which differ in the ontological commitments they make but, importantly, there is little existing work that examines their suitability as an ultimate 'mediating layer' within a practical data integration context.

Here, we employ a 4D foundational ontology as a means of dealing with the diversity of representation and semantics within acquired data. We do this in the context of a semantic Extract-Transform-Load framework (called 4D-SETL from

Cesare, S., Foy, G. and Lycett, M.
4D-SETL - A Semantic Data Integration Framework.
In *Proceedings of the 18th International Conference on Enterprise Information Systems (ICEIS 2016) - Volume 1*, pages 127-134
ISBN: 978-989-758-187-8

this point) that uses a 4D foundational ontology to harmonise data, then generates a graph database that accords with the semantic commitments made by the ontology. We examine the effectiveness of the framework by applying it to semantically interpret and integrate a number of large-scale datasets and to instantiate a data warehouse based on a graph database to persist the resultant ontology. In doing this, the paper is structured as follows. Section 1 outlines the problem of variety in terms of the semantic heterogeneity that exists within systems modelling and foundational ontologies and also identifies a number of the weaknesses in current integration approaches. Section 2 describes the core categories and foundational patterns of the BORO foundational ontology. Section 3 introduces the 4D-SETL framework. Section 4 provides details of the experimental dataset integrated. Section 5 details the outcomes and limitations.

1.1 Semantic Data Integration

Data integration is problematic on several counts. Firstly, people perceive and conceptualise reality in different ways. Even when a set of models is developed by the same individual, they can make different (and sometimes arbitrary) choices about the same reality at different times and in different contexts (Kent, 1978). Secondly, in the course of modelling reality, a designer may confuse what is being represented with the representation itself (Partridge *et al*., 2013). Thirdly, different structures and restrictions are introduced by heterogeneous modelling methods and languages (e.g., Entity-Relationship, OWL etc.). Fourthly, it is common practice to develop a number of models in systems development – conceptual, logical and physical data models for example (Codd, 1970). This layering can have an adverse effect as the original semantic structures may be distorted or lost completely as the emphasis of the modelling activity moves from representing the real world to representing data structures. Consequently, when integrating data that originates from different sources, the problem of semantic heterogeneity arises – resolution is required regarding differences in meaning, interpretation or the intended use of related data which forms a barrier to coherent semantic data integration (Doan, Noy and Halevy, 2004).

1.2 Heterogeneous Foundational Ontologies

Ontology provides a way of dealing with semantic data integration. From a computational standpoint, an ontology is generally taken as a 'specification of a conceptualization' (Gruber, 1995) – that is, a description of the concepts and relationships that are considered legitimate within a particular system of thought. In terms of the concrete implementation of software systems, foundational ontologies can be used to establish the fundamental 'meta' objects and relations used to construct more specific domain ontologies. If a common foundational theory is extended and specialised to model a number of domain ontologies, then objects common to each of these domains will have the necessary (common) grounding to enable semantic integration. Consequently, foundational ontologies are important as they provide a standpoint that underpins all the domain models to be integrated – providing a semantic grounding.

It is the case, however, that several such standpoints (related to foundational ontologies) exist, Each provides a criterion for the ontological commitments made (implicitly or explicitly), which are principally the things believed to exist within the context of a particular theory such as four-dimensionalism (Quine, 1952; Sider, 2003). An understanding of ontological commitment, however, means that the computational view of ontology needs to defer to a philosophical one, which is more specifically concerned with the nature of being (metaphysics). As metaphysical theories differ on a number of dimensions (realism versus idealism, endurantism versus perdurantism to name but two) differences thus appear in foundational ontologies. Furthermore, and perhaps more importantly, the degree to which foundational ontologies are actually grounded in metaphysics varies. Clearly, a lack of consensus at the metaphysical level introduces obstacles to semantic integration (Campbell and Shapiro, 1995) that result in weaknesses in computational applications:

a) **Lack of Grounding**. Many current models employed within information systems have no form of grounding in a more fundamental theory (Cregan, 2007). Thus the ontological commitments underlying the model are unknown. On examination of many Linked Open Data ontologies, they are often ungrounded.
b) **Integrating Elements from Models which are Founded on Different Theories.** There are many automatic translation techniques for translating RDBMS schema and data to an OWL 'ontology'. However, there is a lack of recognition that the expressivity of Description Logics (that underlie OWL) and RDBMS are

different as are the unique naming and the open/closed world-view assumptions.

c) **Model Strata and Translations.** As noted earlier, the requirement to translate the high-level models of reality created at the initial design to structures that are focused on the execution environment can result in semantic distortion. There is also the problem of translating between run-time representations; the often cited OO-RDBMS impedance mismatch (Ireland *et al.*, 2009).
d) **Over Simplification to Fit a Model of Reality to a Tractable First Order Logic (FOL) Theory.** The simplification of the abstraction of reality to fit neatly into a FOL theory, thus ignoring the fact that reality is not so simple and higher order objects exist (Bailey, 2011).
e) **Dividing Models into Static and Dynamic Types.** The separation of static and dynamic aspects of reality into different structural and process models leads to the development of incompatible abstractions together with ‘exotic’ relations that are employed to bridge these static and dynamic worlds.
f) **Naming and Meaning Confusion.** There is often confusion between an entity’s naming and meaning (Bailey and Partridge, 2009). An object’s place in reality (and within ontology) should define its meaning.
g) **Establishing Identify.** Many modelling and information systems use ephemeral means of establishing an entity’s identity which do not function well over time.
h) **Employing Techniques that do Not Scale.** Software tools such as OWL tableau calculus-based reasoners are constrained by memory and cannot be easily scaled to inference over ontologies containing large instance populations (Bock *et al.*, 2008). The alternative is to use simplified semantics and rule based reasoning - that could in many cases employ standard RDBMS techniques.
i) **Semantic Integration Mismatches**. For a more extensive discussion on the types of semantic integration mismatches see Visser *et al.*, (1997) who provide an extensive list of e semantic mismatches that can occur when integrating disparate datasets.

2 BORO FOUNDATIONAL ONTOLOGY

Having examined several foundational ontologies from a philosophical perspective, the research described here adopts the Business Object Reference Ontology (BORO) (Partridge, 2005) to semantically interpret the original datasets and models. We adopt BORO on the grounds that the ontology can overcome the dichotomy that exists between dynamic and static modelling paradigms and its metaphysical thoroughness. Hence, the same model can represent processes and things that are not traditionally considered as processes (e.g., people, products, machines, etc.). BORO represents all individual elements (e.g. the activity, the person assuming a role and the resource consumed) in exactly the same way (i.e. as spatiotemporal extents). BORO is based on a philosophical (rather than computational) definition of ontology because it requires more clarity on “the set of things whose existence is acknowledged by a particular theory or system of thought” (Lowe, 1998, p.634.). Key to overcoming the dichotomy noted is the fact that BORO is perdurantist (and thus extensionalist) in its nature. In perdurantism (or 4D) an individual object is never wholly present at one point is time, but only partly present (a temporal part). For example, John is not fully present in any given phase of his life (e.g., childhood), he is fully present from his birth to his death only – therefore, John’s childhood is a temporal part of John. Identity is thus defined by an individual object’s spatiotemporal extension (or extent). Figure 1 represents the key part of the foundational ontology relevant for the purposes of this paper.

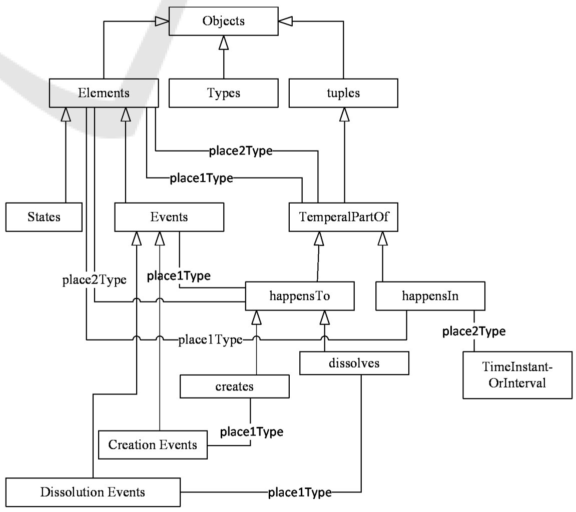

Figure 1: BORO Foundational Ontology (top level).

More in-depth discussions are provided in Partridge, 2002; 2005; Bailey and Partridge 2009; Bailey, 2011; Partridge et al., 2013). At the top level the BORO foundational ontology represents:

• **Elements**, which are individual objects or objects with a spatiotemporal extent. For example, the person *John*.

• **Types**, which are sets or objects that can have instances. The identity of a type is also extensional but, in this case, it is defined as the set of its instances (i.e. members). For example, the extension of the type *Persons* is the set of all people.

• **Tuples**, which are relationships between objects. The identity of a tuple is defined by the places in the tuple. An example is (*Persons*, *John*) in which the type Persons and the element John occupy places 1 and 2 in the tuple respectively. This specific tuple is an instance of the tuple type *typeInstances* in BORO.

In turn, Elements is subtyped by:

• **Events:** An event is an element that does not persist through time (i.e. an event has zero 'thickness' along the time dimension). Events represent temporal boundaries that either create (CreationEvents) or dissolve (DissolutionEvents) elements (e.g., a person or a person's childhood state).

• **States:** A state is an element that persists through time. States (and elements in general) are bounded by events. A state (like all elements) can have further temporal parts (i.e. states and events). Specific TupleTypes (or types whose instance are tuples) relevant here are:

• **temporalPartOf:** This tuple type relates an individual with its temporal parts (states and/or events).

• **happensTo:** This tuple type relates an event with one or more elements affected by the event. happensTo has two subtypes:
- creates: Relates a creation event with the element(s) whose creation is triggered by the event.
- dissolves: Relates a dissolution event with the element(s) whose dissolution is triggered by the event.

• **happensIn:** This tuple type relates an event with a time instant or interval (TimeInstantsOrIntervals) and it indicates the time in which an event takes place.

As a note of importance for the example shown later, names are types in BORO. The instances of the name of an individual (e.g. John Smith's Name) are all utterances (written, spoken, etc.) that name that individual (e.g., John Smith). Therefore while a name, is a type its instances are spatiotemporal extents. To provide clarity within the ontology, 'names' as much elements of the ontology as the things they name. A name object will belong to a Name Space which holds all names related to a particular naming authority or domain. As the ontology adopts a theory of utterances – each utterance of a name is an individual element and so has an extent (Strawson, 1964). Therefore, a name is a Type that has as instances all utterances of the same name individuals.

3 4D SEMANTIC EXTRACT TRANSFORM AND LOAD FRAMEWORK (4D-SETL)

Given an outline understanding of the foundational ontology, we now describe a Semantic Extract-Transform-Load framework. Given a variety of data input, 4D-SETL is designed to output a graph database in accordance with the BORO foundational ontology. The framework was designed around a number of industry standard tools and technologies (e.g., a UML design tool and a Graph Database), supplemented where necessary with custom software implemented in Java. The key technology choices made for the initial implementation were threefold. First spreadsheets were employed to document each dataset. Second, a UML design tool (Enterprise Architect) was selected as the graphical design tool for the ontological models and a BORO custom UML profile created: The advantage is that BORO UML enables easy manipulation and design of each of the required domain ontologies. Last, the Neo4J Graph database was chosen for persistence, for several reasons: (a) Primarily due to its flexibility in enabling BORO to be used as the foundational 'schema' (both can be seen as graphs); (b) scalability in order to handle model and instance data volume appropriately; (c) Neo4J's web-based interface also provides access to the graph database for development testing; and (d) Neo4J Cypher provides an appropriate means of querying and updating the graph database resident data.

The Semantic Extract Transform Load (ETL) process is shown in Figure 2, the key stages of the process are as follows:

a) **Semantic Extraction and Transformation.** The input data to a semantic integration process may be structured in many forms –e.g., as fixed record or delimited tabular files, RDF, RDFS, OWL etc. – and may consist of both model (schema) level and/or instance level data. Thus the first stage in the semantic integration process begins with documenting the dataset which can be considered a semantic extraction and transformation process. The BORO foundation provides a view of reality and the patterns that can be employed perform this interpretation and transformation. The foundational ontology provides the equivalent of a canonical data model (Saltor *et al*., 1991) that can be employed to develop domain models providing the semantics that are common to all datasets that will be integrated. Thus the translation process results in a new schema (domain ontology) that extends the ontic categories and patterns of the foundation. Through this process, schemas are developed to represent the entities and relationships that are represented by the data. Finally this schema is documented using a profile of UML that conforms to BORO semantics.

b) **Ontology Model ETL.** Once a domain model has been created in an ontologically consistent form the semantic load and integration process can be undertaken. Firstly the domain ontological model, which includes such patterns as type and classification taxonomies, is translated from the BORO UML model and loaded to the graph database. The 4D-SETL framework provides a Java application to translate the BORO UML and load it to the graph database.

c) **Ontology Data ETL.** Next, the instance level dataset is loaded and integrated. It is through this process that the integration of individual elements takes place. Integration can be considered to take place within vertical and horizontal planes. Initially the 'vertical' relationships between an individual element and the domain ontology (and hence the foundation ontology) is asserted, which consists of establishing the individual relationships (such as type instance, etc.). Then the 'horizontal' relationships that are deemed to hold between individual domain level objects are established (such as a company being located at a particular geographic location). Foundational ontological patterns can then be applied to simplify this process. This can be a complex process that requires both one-to-many and many-to-one transformations. The 4D-SETL framework provides a Java application to perform this process.

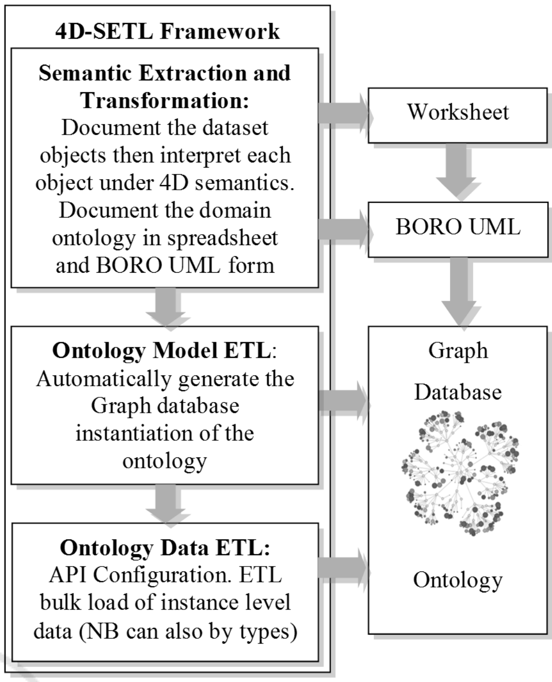

Figure 2: 4D-SETL Framework.

4 EXPERIMENTAL DATA

As the foundational ontology is an integral part of the framework, prior to processing any of the domain ontological elements (model and datasets) the foundational ontology is transformed to graph format and loaded to the database. This is achieved via the 4D-SETL framework which extracts the BORO UML as XML (XMI), then transforms it to a set of nodes and edges that are loaded to the database. The graph database ontology also includes the UML model identifiers as indexed node and edge parameter key-values, these are employed to enable the reproduction of the design time UML models within the graph database runtime environment and to establish the relationships between the foundation and other subsequent domain model elements that are loaded. The 4D-SETL framework was applied to Extract, Transform and Load (ETL) five datasets of varying scale and complexity related to corporate data:

- Calendar: temporal locations (1856 to present).
- Location: spatial locations (~2.5M locations).
- Standard Industrial Classification (taxonomy).
- UK Companies (~3.5M)
- UK corporate officers (~12M)

5 OUTCOMES AND LIMITATIONS

Having applied the framework, our experience is that BORO provides a coherent lens through which to view and model the world together with the foundational ontological elements and patterns through which the domain ontologies can be developed to represent the datasets to be semantically integrated (Partridge, 2002). In terms of domain ontology development, this work concurs with the work of Keet (2011), who stated that employing foundational ontologies provides advantages in terms of the quality and interoperability of domain ontologies. Developing such domain ontologies provided the means of semantically integrating data conforming to different models and theories – a necessary evil in dealing with variety in big data.

Employing a graph database provided the means of importing and restructuring data in a manner that directly reflects the ontological model patterns without the normal translation to tabular RDBMS or Object Oriented form and not introducing the 'impedance mismatch' problem (Ireland *et al.*, 2009). Dispensing with RDBMS storage in favour of a property graph data model removed the partitioning of the storage structures between data and schema and allows both 'schema' ontological model objects and instance level objects to be updated at run time. This supports the work related to graph databases by Webber (2012). Related to this finding, it was also demonstrated in this study that patterns could be established within the warehouse that directly reflected the physical or socially constructed patterns of reality such as taxonomies and taxonomic ranks, the latter of which employed the powertype pattern (equivalent to the set theoretic powerset) to more accurately reflect the nature of such classification systems. These aspects of 4D ontologies (along with others) provide a greater level of flexibility and reusability when evolving the warehouse system and therefore concur and take forward the initial findings of Partridge (2002).

In practical terms, we propose that the data structures resulting from the 4D-SETL process are more suitable for discovering relationships within data rather than for example processing aggregate data (Vicknair *et al*., 2010). It is relatively easy, for example, to discover all relationships that exist between two elements using a standard algorithm from the Neo4J library (designed to find all available paths or the shortest path between two nodes). Further, the Cypher graph database query facilities provide the means of discovering more complex patterns of relationships between the people, company officers, company activities, events and physical location. Finally, it was found through the evaluation and empirical experiment on the prototype warehouse (graph database) that data load and information retrieval response times that the prototype could be developed into a practical information system. This was confirmed by performing test data query (graph traversal) experiments that for example, performed graph traversals to retrieve all companies within a postcode location (61 milliseconds) and all officers for a specific business organisation (37 milliseconds) thus the prototype produced indicative response times within bounds that would support interactive applications (Bhatti *et al.,* 2000). Testing also confirmed the graph database performance evaluation undertaken by Vicknair *et al.* (2010). Thus using a graph database and the parameter graph model to store the ontology, alongside query information via graph traversal, circumvents the issues that limit the ability of systems built using triple stores and tableau calculus-based reasoner technology to deal with ontologies that are both expressive and have with very large instance level elements (arguably exactly what one would want from big data). Neo4J is highly scalable and provides capacities for Nodes/Edges of ~34 billion and properties at least ~ 68 billion respectively.

With the issue of disparate data sources in mind, the work here has: (a) Examined the potential contribution of foundational ontology; and (b) described an implementation of a Semantic Extract-Transform-Load framework (4D-SETL) based on BORO, a 4D foundational ontology. Foundational ontologies provide a 'grounding' for our view of reality and thus provide a common reference through which to model and integrate heterogeneous data. The 4D-SETL framework uses the BORO foundational ontology to harmonise data and then generates a graph database that accords with the semantic commitments made by that ontology. The effectiveness of the framework was examined applying it to large-scale open datasets related to company information to semantically interpret and integrate the datasets and to instantiate a prototype graph database warehouse to persist the resultant ontology. Our implementation is a prototype at this stage and the use of foundational ontologies is not without challenge (e.g., automation in the context of real-time data streams). Accepting such limitations, however, the potential utility of the 4D-SETL framework can be seen in its ability to model and instantiate a number of complex ontological structures, such as higher order taxonomic ranks. The patterns specialised from the core foundational BORO ontology patterns offer a high degree of flexibility and reusability when evolving the graph-based warehouse system. We have thus demonstrated how a 4D (perdurantist) foundational ontology can be employed to semantically interpret and structure data, showing that a single coherent ontology can be developed and loaded to a graph database without the problems associated with current approaches – e.g., model distortion, over simplification or scalability problems.

Understandably, the work here is not without its limitations, which may be summarised as follows. First, and at the outset, the interpretation process is manual. BORO encourages the development of patterns (for ontological reuse), which allow for partial automisation but skills in ontological modelling are necessary throughout. In the context of dealing with variety in big data automatic translation of data is of particular importance. As a consequence, pattern development and the extraction of the rules associated with that are also of importance for ongoing research. Second, as previously noted, BORO is one of several foundational ontologies and further work is required to understand their relative comparative advantages and disadvantages.

The work here was supported by funding from the Engineering and Physical Sciences Research Council (Project EP/L021250/1). The experimental research data and metadata (Ontology) for this project was sourced from the following organisations: Companies House (2016), Company Information; UK Office of National Statistics (2016) Geographic Location (ONSPD Product); UK Office of National Statistics (2016), Standard Industrial Classification; Company Officers: (A commercial credit reference agency); BORO Engineering Limited (2016), Foundational Ontology.

The Companies House and ONS Datasets are UK Open Government Data and can be freely downloaded. The Company Officers and BORO Ontology are commercial in-confidence.

REFERENCES

Arsanjani, A. (2002) 'Developing and Integrating enterprise Components and Services', *Communications of the ACM,* 45(10), pp. 30-34.

Bailey, I. (2011) 'Enterprise Ontologies–Better Models of Business', in *Intelligence-based systems engineering.* Springer, pp. 327-342.

Bailey, I. and Partridge, C. (2009) 'Working with extensional ontology for defence applications', *Ontology in Intelligence Conference.*

Bhatti, N., Bouch, A. and Kuchinsky, A. (2000) 'Integrating user-perceived quality into web server design', *Computer Networks,* 33(1), pp. 1-16.

Bock, J., Haase, P., Ji, Q. and Volz, R. (2008) 'Benchmarking OWL reasoners', *Proc. of the ARea2008 Workshop, Tenerife, Spain (June 2008).*

BORO Engineering Limited (2016) 'BORO Ontology'. Available from: < http://www.borosolutions.co.uk/solutions/resources/boro-presentations-and papers >. [16 February 2016].

Bricker, P. (2014) 'Ontological Commitment', in Edward N. Zalta (ed.) *The Stanford Encyclopedia of Philosophy.* Winter 2014 edn.

Campbell, A. and Shapiro, S. (1995) 'Ontological Mediation: An Overview', *IJCAI Workshop on Basic Ontological Issues in Knowledge Sharing.* 1995. AAAI Press, Menlo Park, CA,.

Codd, E. (1970) 'A relational model of data for large shared data banks', *Communications of the ACM,* 13(6), pp. 377-387.

Companies House (2016), Free Company Data Product. Available from: < http://download.companieshouse.gov.uk/en_output.html >. [16 February 2016].

Cregan, A. (2007) 'Symbol grounding for the semantic web', in *The Semantic Web: Research and Applications.* Springer, pp. 429-442.

Doan, A., Noy, N.F. and Halevy, A.Y. (2004) 'Introduction to the special issue on semantic integration', *ACM Sigmod Record,* 33(4), pp. 11-13.

Gangemi, A., Guarino, N., Masolo, C., Oltramari, A. and Schneider, L. (2002) 'Sweetening ontologies with DOLCE', in *Knowledge engineering and knowledge management: Ontologies and the semantic Web.* Springer, pp. 166-181.

Grenon, P. and Smith, B. (2004) 'SNAP and SPAN: Towards dynamic spatial ontology', *Spatial cognition and computation,* 4(1), pp. 69-104.

Gruber, T.R. (1995) 'Toward principles for the design of ontologies used for knowledge sharing?', *International journal of human-computer studies,* 43(5), pp. 907-928.

Guizzardi, G., de Almeida Falbo, R. and Guizzardi, R. (2008) 'Grounding Software Domain Ontologies in the Unified Foundational Ontology (UFO): The case of the ODE Software Process Ontology.', *CIbSE.* , 127-140.

Herre, H. (2010) 'General Formal Ontology (GFO): A foundational ontology for conceptual modelling', in *Theory and applications of ontology: computer applications.* Springer, pp. 297-345.

Ireland, C., Bowers, D., Newton, M. and Waugh, K. (2009) 'A classification of object-relational impedance mismatch', *Advances in Databases, Knowledge, and*

Data Applications, 2009. DBKDA'09. First International Conference on. IEEE, 36-43.
Keet, M. (2011) 'The use of foundational ontologies in ontology development: an empirical assessment', in *The Semantic Web: Research and Applications.* Springer, pp. 321-335.
Kent, W. (1978) *Data and reality : basic assumptions in data processing reconsidered.* Amsterdam ; Oxford: North-Holland Publishing Co.
Lowe, E.J. (1998) 'Ontology.', in Hondreich, T. (ed.) *The Oxford Companion to Philosophy.* New York: Oxford University Press, pp. 634.
Lycett, M. (2013) ''Datafication': Making sense of (big) data in a complex world', .
Partridge, C. (2002) 'The role of ontology in integrating semantically heterogeneous databases', *Rapport technique,* 5(02).
Partridge, C., Mitchell, A. and de Cesare, S. (2013) 'Guidelines for developing Ontological Architecures in Modelling and Simulation', in Tolk, A. (ed.) *Ontology, Epistemology, and Teleology for Modeling and Simulation.* Berlin Heidelberg: Springer, pp. 27-57.
Office of National Statistics (2016), Postcode Data Product. Available from: < http://www.ons.gov.uk/ons/guide-method/geography/products/postcode-directories/-nspp-/index.html>. [16 February 2016].
Office of National Statistics (2016), Standard Industrial Classification System 2007. Available from: < http://www.ons.gov.uk/ons/guide-method/classifications/current-standard-classifications/standard-industrial-classification/index.html>. [16 February 2016].
Partridge, C. (2005) *Business objects.* 2nd edn. Oxford: Butterworth Heinemann.
Quine, W.V. (1952) *Methods of logic.* Routledge and Kegan Paul.
Saltor, F., Castellanos, M. and Garcia-Solaco, M. (1991) 'Suitability of Data models As Canonical Models for Federated Databases', *SIGMOD Rec.,* 20(4), pp. 44-48.
Sheth, A.P. and Larson, J.A. (1990) 'Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases', *ACM Comput.Surv.,* 22(3), pp. 183-236.
Sider, T. (2003) *Four-dimensionalism: An ontology of persistence and time.* Oxford.
Strawson, P. F. "Identifying reference and truth□values", Theoria, 30(2), 1964 pp. 96-118.
Vicknair, C., Macias, M., Zhao, Z., Nan, X., Chen, Y. and Wilkins, D. (2010) 'A comparison of a graph database and a relational database: a data provenance perspective', *Proceedings of the 48th annual Southeast regional conference.* ACM, 42.
Visser, P.R., Jones, D.M., Bench-Capon, T. and Shave, M. (1997) 'An analysis of ontology mismatches; heterogeneity versus interoperability', *AAAI 1997 Spring Symposium on Ontological Engineering, Stanford CA., USA.* , 164-172.
Webber, J. (2012) 'A programmatic introduction to Neo4j', *Proceedings of the 3rd annual conference on Systems, programming, and applications: software for humanity.* ACM, 217-218.
Zikopoulos, P. and Eaton, C. (2011) *Understanding big data: Analytics for enterprise class hadoop and streaming data.* McGraw-Hill Osborne Media.