Enterprise Data Modelling: Developing an Ontology-Based Framework for the Shell Downstream Business

Matthew WEST[a]* Chris PARTRIDGE[b], Mark LYCETT[c]

[a]*Shell International Petroleum Company, Shell Centre, London, SE1 7NA. United Kingdom.*
E- mail: matthew.west@shell.com

[b]*BORO Centre Ltd.,83 Sixth Avenue, London, W10 4HF. United Kingdom.*
E-mail: chris.partridge@BOROCentre.com

[c]*School of Information Systems, Computing and Mathematics*
Brunel University, Uxbridge, Middlesex. UB8 3PH. United Kingdom.
E-mail: mark.lycett@brunel.ac.uk

Abstract: This paper examines the development of a conceptual model that defines Shell's information requirements - the Downstream Data Model (DDM). The model has its roots in a framework based on the notion of ontological commitment and the focus of the analysis seeks to provide useful insights into the metaphysical aspects relevant to the creation and deployment of the DDM – primarily that related to the extensional nature of the model. The impact of this choice and the methodology employed in the production of the model is examined through example patterns covering spatial and temporal dissectiveness and the use of powerclasses. Having been through the experience of conceptual model development, the work concludes that the separation of the implementational and epistemological 'gloss' from a studied understanding of ontological commitment is a necessary evolution of practice in conceptual modelling.

Keywords: ontology, data model, enterprise, ISO 15926, BORO Methodology, four-dimensionalism, extensionalism, metaphysical choices.

1. Introduction

Conceptual modelling is an important activity in terms of both organizational understanding and systems development. Despite this importance, and evidence to suggest that 'errors' in modelling are increased by orders of magnitude later in the systems development and

* Corresponding Author: Matthew West, matthew.west@shell.com.

maintenance process, it is well noted that conceptual modelling remains more of an ‘art’ than science [11, 12, 13]. In addition, given an age of integration, it is increasingly recognized that semantic understanding and interoperability is a key challenge for organizations and their systems [10, 15]. Semantic interoperability is a knowledge-level concept that provides the “ . . . ability to bridge semantic conflicts arising from differences in implicit meanings, perspectives, and assumptions, thus creating a semantically compatible information environment based on the agreed concepts between different business entities.” [15].

Ontology is an emerging mechanism for dealing with semantic interoperability. The broadly accepted definition of ontology in the computing field is that of a ‘specification of a conceptualization’ [4]. Typically, the use of the concept in practice has been that of specifying the conceptualization in terms of an ‘agreement’ on meaning between the parties involved in ontology development - what ‘exists’ is that which can be represented and the ontology becomes a reification of an agreement on knowledge. A subtle but important difference exists between agreement of semantics (what will be represented) and the ‘nature’ of what will be represented however. In this sense, nature examines what a concept actually ‘commits’ to in the business world – leading to the construction of a more general understanding of the ‘real world’, which constrains and guides other more specific models. The issue is that, in mainstream academic and commercial work, practitioners typically regard their data models as representations of the ‘real world [7], rather than ‘a reification of an agreement on knowledge’.

With such issues in mind, and in-line with the workshop aims, the paper discusses (a) a concrete problem, which is that of defining Shell’s information requirements and (b) the solution route taken in attempting to solve that problem. The solution itself takes the form of the Downstream Data Model (DDM), which has its roots in an ontological framework based on the notion of ontological commitment. The focus of the analysis seeks to provide useful insights into the strategic aspects relevant to in the creation and deployment of the DDM – primarily those related to the metaphysical choices made. The paper is structured as follows. Section 2 presents the research environment – a global oil company – to clarify the business drivers behind the innovation and the explicit business purpose of the DDM. Section 3 describes the framework behind the DDM, concentrating on the key metaphysical choices made particularly that of four-dimensional paradigm. Section 4 then highlights the implications of those choices in relation to key elements of the model – highlighting patterns emerging from analysis related to (a) spatial dissectiveness, (b) powerclasses, (c) participation and (d) managed relationships. The paper concludes by reflecting on the implications for future practice.

2. Background

Shell is a global group of energy and petrochemicals companies. Shell Downstream encompasses all the activities necessary to transform crude oil into Shell petroleum products and petrochemicals, and deliver them around the world. Shell’s Downstream Business refines, supplies, trades and ships crude oil worldwide, and manufactures, transports and markets fuels, lubricants, bitumen, LPG and bulk petrochemicals for domestic, transportation and industrial uses. Altogether, the organization employs some 80,000 people.

In an increasingly competitive downstream market, Shell assessed that the cost and complexity of business systems and processes provided an opportunity to improve performance. As a consequence, it has aggressively sought to achieve operational excellence through an ongoing program of global standardization. In practice the organization seeks to achieve such excellence primarily through a combination of (a) business portfolio improvements, (b) the introduction of global processes and standards underpinned by a simplified global organization and (c) the adoption of consistent behaviours to reinforce the perceived benefits of going global. Process streamlining forms a key component in the strategy to simplify and standardize the way that the organization does business, with the objectives of:

- Promoting more accurate and responsive customer interactions.
- Removing errors and rework.
- Reducing costs by eliminating ‘noise’ in business processes.
- Providing proven and simpler ways of doing things.

Unsurprisingly, the standardization of the critical IT systems is seen as key to the success of the streamlining initiative. Thus, a partner initiative aims to replace fragmented Enterprise Resource Planning and other legacy information systems with a harmonised global platform. Broadly speaking, the aim is to reduce the number of operational information systems to less than a tenth of those that existed at the start of the globalisation process (a reduction that is significant).

In order to assist standardization on the process and systems fronts, Shell have also sought to instigate a step change in the way that key Master and Reference Data (MRD) is managed in relation to their customers, products, suppliers, materials, technical assets and accounts across the Downstream businesses and functions. One key requirement here is that of deploying quality standards and measures to ensure that key reference data is fit for purpose. This means, for instance, that the right product be delivered to the right customer at the right address, with costs and profits correctly classified and reported. Consequently, Data Quality Standards (DQSs) have been defined along such lines (e.g., no obsolete customers, no duplicate records etc.) and a significant program has been instigated to ensure that streamlined IT systems are cleansed and validated. Cleansing is the process of removing or correcting data that is incomplete, inaccurate or improperly formatted. Validation is the process of ensuring that DQSs have been properly implemented. Again, this is a significant program, with an effort estimated at 300 man-years. Data cleansing is seen as important as poor data quality not only results in inefficient business processes, it also potentially limits organizational ability to analyze, understand and manage the business in the most effective ways.

The effort here mirrors observations in the literature that data quality issues have become increasingly prevalent in practice - costing organizations significantly, alienating customers and suppliers and hindering decision making and the implementation of strategy for example [1, 2, 19, 20, 23, 26]. In addition, data quality in the context of compliance has become more critical since the Sarbanes-Oxley Act of 2002. The intrinsic treatment of data quality (devoid of context) is problematic however [23]. To this extent, a smaller element of the body of literature starts to form the basis of a ‘business case’ for a slightly different but less explored perspective on the issues of data quality. Redman [19] is an early work that notes the difference between comparing data and the real world and ‘database bashing’, which is more in line with typical industrial treatment. Importantly, he also notes that the fact that solution approaches of the former type are often attempted downstream, resulting in improvements are not typically sustained (*ibid*). Orr [14] makes a similar argument, proposing that data quality is “the measure of the agreement between the data views presented by an information system and that same data in the real world” (p.67). Other works take a similarly representational view on data quality, noting the importance of the semantic/ontological foundations of data quality and including incomplete and/or ambiguous representation as key design deficiencies [26, 18]. For reference, the later literature in the area favours the term ‘information quality’ over ‘data quality’. While much of the literature explicitly uses the terms interchangeably, the distinction is that ‘data’ typically refers to the stored content, whereas ‘information’ refers to the situation where such content has been delivered/presented and interpreted [18].

The representational view has relevance here as the MRD team realized that, in the context of streamlining and standardization, data cleansing and validation is not action enough in relation to MRD. The Downstream Data Model (DDM) was thus developed in response to the recognition that the large number of relatively independent projects that were bringing about the transformation in business and IT processes required a standardized basis for integration and consistency across the business. In essence, common processes and common systems indicated a strong requirement for a common data model. The stated business purposes of the model are to:

- Identify the key objects of interest to the business and the relationships between them
- Provide a specification of the information requirements for the Downstream business
- Identify the underlying transactions and relationships
- Provide a basis for checking that the process model includes the processes for managing both objects and data about objects
- Provide a basis for checking that the physical data model, user and system interfaces in applications support the information requirements

In the context of the literature, the approach is a sensible one. The objective of streamlining IT systems around a core set of Enterprise Resource Planning systems is a common means of attempting to provide a seamless integration across a full range of organisational processes – uniting functional and global areas within the business and making their data visible in a real-time manner. Some analyses of ERP systems implementation indicate that organizations must be willing to develop common definitions and understanding for both data and process across the business [24, 25], though typically the concentration is on the link with process.

3. Foundations of the Model

The DDM thus represents a model of the Shell domain that is independent of any system in which representations of the domain may be implemented. This characterization requires a focus on the information requirements of the organization (and thus of any system) allowing the structure or processing of the system to remain undetermined. For reference, the DDM is a Computationally Independent Model (CIM) from a Model Driven Architecture (MDA) perspective. CIMs are relatively under-explored in relation to the work available in other areas of the MDA and initial work in relation to the DDM indicated that a finer separation of concerns was required in relation to the CIM classification. Essentially, one can distinguish an ontological representation (a model of 'what is', in essence a view from nowhere) from epistemological representation (a model of 'what is known' about the domain by some agent, and how it may be represented in a system). In these terms, the DDM is developed as a hybrid model - it is in large part an ontology, but with an epistemological 'gloss', which represents what Shell as an organization (the 'agent') knows, rather than what any particular system knows. Given that business users in Shell are unfamiliar with terms such as ontology and epistemology, the DDM is referred to within Shell as a (conceptual) 'data model'. This can be further distinguished from an implementational representation – a representation in terms of the technology used to implement a particular application or set of applications of the implementation. In traditional data processing this would be known as a physical model.

3.1. Methodology

Shell's downstream business is extremely wide in its scope. In traditional enterprise modelling, this would lead to a large-scale model containing a significant number of entities. One of the greatest risks in these kinds of projects is the difficulty in managing the complexity of a large complicated model. Hence, a conscious decision was made to adopt the BORO methodology that focused on identifying the general underlying patterns that occurred across and within domains, rather than their specific specialisations in the domains – for reference, a similar policy had been successfully adopted in the development of the current version of ISO 15926-2:2003 [9]. This methodology are described in detail in[16].

The starting point for the methodology is that of providing a strong ontological foundation, which involves two key things. First, it is important to clarify the metaphysical choices that underlie the ontology [17]. Second, it is important to clarify the ontological categories that are being committed to alongside their associated criteria of identity. It is also important that such choices are made on the basis of their pragmatic implications for identifying the general underlying patterns. Understandably, it is vital that these commitments are followed through – for consistency; the DDM must stick to its metaphysical choices and ontological categories. A key role of the BORO Methodology was helping to ensure this. With choices and categories clear, the methodology in application seeks to facilitate the identification of general patterns that underlie (a) different domains and (b) different aspects of the same domain. In essence, if general ways of explaining the world can be reengineered out of many specific ones, a more compact model can be developed.

3.2. A Key Metaphysical Choice – 3D Versus 4D

In practice, there are a number of ways in which the world can be modelled (where the choice may be explicit or implicit in the conceptual technology used). The key metaphysical choice discussed here relates to the use of a three-dimensional (3D) paradigm or four-dimensional (4D) paradigm (though they are also known as endurantism, and perdurantism respectively – reflecting that the choice is about the treatment of change rather than dimensions). A 3D ontology treats things as 3D objects (sometimes called continuants) that pass through time. The principles of the 3D paradigm are:

1. Objects are three-dimensional objects that pass through time and are wholly present at each point in time.
2. Objects are viewed from the present. The default is that statements are true now.
3. Objects do not have temporal parts.
4. Different objects may coincide at a point in time, i.e. occupy the same 3D extension (non-extensionalism).

Thus, to talk about an object at different times it is necessary to time index statements in some way (e.g., *X* at *t*). In contrast, a 4D ontology treats all individuals - things that exist in space-time - as spatio-temporal extents (i.e., as 4D objects). The 4D paradigm is usually associated with an extensional view of objects and its principles are:

1. Individuals exist in a manifold of four dimensions, the three of space plus time. So things in the past and future exist as well as things in the present.
2. The four dimensional extent is viewed from outside time rather than in the present.
3. Individuals extend in time as well as space and have temporal parts as well as spatial parts.

4. When two individuals have the same spatio-temporal extent they are the same thing (the extensionalist criteria of identity).

Thus a 4D object is not (usually) wholly present at a point in time, but its whole is extended in space as well as time. The object at a point in time is a temporal part of the whole. Change is naturally expressed through a four-dimensional classical mereology [22]. For reference, good proponent descriptions of the 4D paradigm can be found in Sider [21] and Heller [6]. It is noted that it is possible to have an approach that combines a 3D ontology for some types of individuals, say physical objects, but uses a 4D approach for others, such as activities.

As an example, one of the entities that crops up repeatedly in the various domains is ‘Shell Organisation’. Shell purchases and sells organisations – and while they are owned by Shell, they are Shell organisations. In a 3D view of the world, Shell Organisation would be a time-indexed property of organisations (the Shell Organisation property owning entity). In the 4D view of the world employed in the DDM, a Shell organisation is the temporal stage of the organisation that was owned by Shell. Notably, both views are reflected in Shell’s systems and the input data to the DDM. The difference here between a time-indexed property and a temporal stage reflects different metaphysical choices, not an empirical feature of the world. The DDM (BORO) approach was to strip out differences due to metaphysical choices, revealing the hidden similarities.

3.3. Pragmatic Considerations

In reality, the ideals of conceptual modelling need to be balanced with the pragmatics arising from organizational history, culture, budget and time considerations and the like. Key considerations here were the (a) modelling start point, (b) modelling language and (c) scope of the work to be undertaken. Addressing these issues in turn, a modelling start-point was given in the form of the Life-Cycle Integration Schema from ISO 15926-2:2003 [9] which is itself a four-dimensional ontology (and was thus consistent with that metaphysical choice). Often the choice of language is a tactical rather than strategic matter, though it is noted that the metaphysical choices/paradigm underlying the tool will influence the outcome or require workarounds if paradigms are incommensurate in some way (later evidenced). The choice made for the DDM was EXPRESS, for the following reasons:

- It is an ISO standard, ISO 10303-11 [8],
- There is a range of tools available to support data model development and publication (Visual EXPRESS from EPM was chosen for the project).
- ISO 15926-2:2003 [9], was developed in EXPRESS.
- EXPRESS is a current Shell standard.

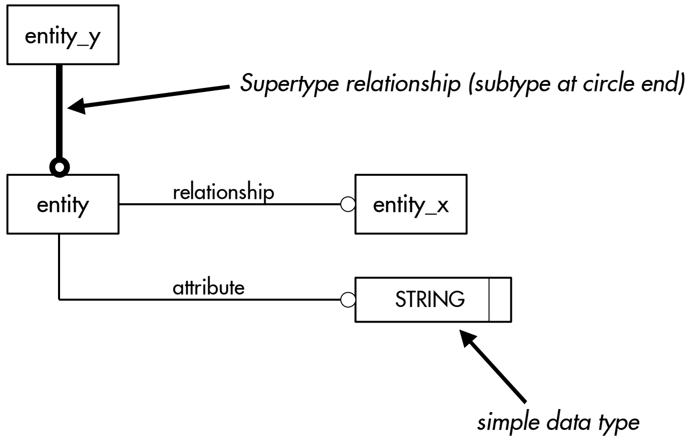

Figure 1. EXPRESS Notation

For reference, Figure 1 illustrates the basic elements of the EXPRESS graphical notation. Entity types are represented by boxes and relationships by lines. A thick line indicates a subtype/supertype relationship. Other relationships are indicated by a line with a lollipop at one end. The relationship name is read from the entity type at the sharp end, to the entity type at the lollipop end. If the relationship line is solid then it is a mandatory relationship for instances at the sharp end of the relationship.

Further to syntax, developing any conceptual model requires some limit on the scope (both in breadth and depth) to ensure the work is manageable within time and cost constraints. Requisite source material is also required as a basis for the work. Given that streamlining within Shell is process-centric, the breadth of the scope was defined as covering the following business processes in Shell's Downstream business (some processes such as Human Resources were scoped out for this version of the model):

- Sell to Business Customer
- Sell to Retail Customer
- Manufacturing
- Manage Lubricants Supply Chain
- Manage Bulk Hydrocarbons Supply Chain
- Procure Goods and Services

The depth of scope of the model was to range from the metaphysical choices at the framework level to a level of abstraction that reflected business language (i.e., leaf subtypes should represent things directly recognised by the business, rather than high-level abstractions of those things). In providing some 'flesh' to the scope, the development of the DDM drew on a range of existing written material as a start point (which meant that interviewing business staff for requirements was not been necessary except for clarification in some cases). The evidence that has been drawn upon in developing the DDM included (a) ISO 15926 [9], (b) the Downstream Process Model, (c) a Glossary of Terms for the Downstream business, (d) the previous version of the Downstream Data Model, (e) Project Logical Data Models (where they have been developed), (f) Physical Data Models from implemented systems, and (g) data from existing systems.

3.3 Changes in Approach

The work was divided up among several data modellers in the form of schemas. EXPRESS and Visual EXPRESS support the development of a number of schemas that make reference to each other to provide integration. Visual EXPRESS, however, is a single user tool, so that each schema can only be worked on by one person at a time (the tool was thus a determining factor in process terms). Initially, each data modeller was allocated one or more process areas to model as one or more process schemas.

However, it quickly became clear that this was unworkable, because so many things like products, organizations, properties, and locations appeared in many of the process areas without them clearly belonging to one of them. This lead to duplication of concepts between schemas and the need for reconciliation between them.

This commonality of concepts between process areas led to this approach being abandoned in favour of one where:

- Subject Area schemas were developed for common concepts, responsibility for which was given to one data modeller,
- Data modellers were given responsibility for ensuring that requirements from their Process Area were met in the Subject Areas.

By project completion, almost the whole model was in Subject Areas – this proving an important factor in integrating requirements across the different Process Areas. The final set of Subject Areas is shown in Figure 2 – the more abstract and widely referred to Subject Areas are shown at the top, with the more Process Area specific schemas shown lower in the triangle. For reference, the numbers in brackets show the number of entity types in each schema (the total size of the DDM is in excess of 1700 entity types).

.

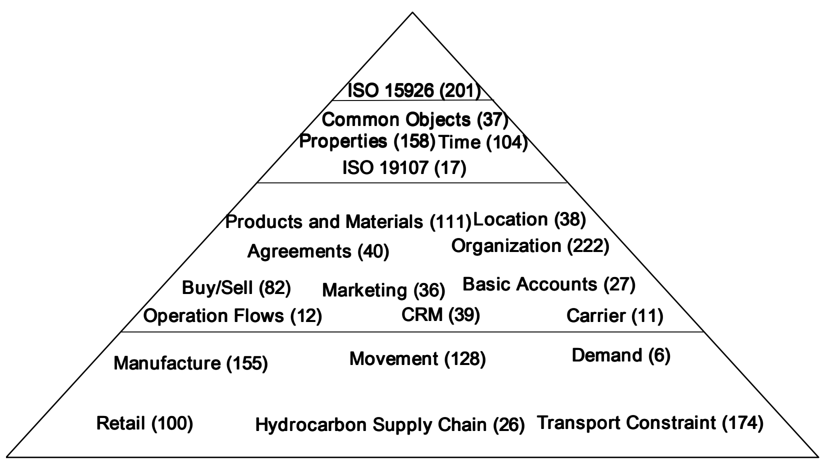

Figure 2. The set of subject areas for the DDM V2.0

Whilst we do not concentrate on process here, review workshops were held bi-weekly in order to (a) review and amend schemas and (b) to normalize the approach taken by the data modelers. Training materials were also produced and sessions given where deemed necessary. In addition, a mid-term external review was carried out by David Hay (of Hay, 1996). Whilst impressed by the overall quality of what had been delivered, observations made were that:

- A mid level of abstraction is missing in some places where useful patterns might be found, and
- Names of leaf entity types were sometimes not in business language.

These concerns were only partially addressed in the current version of the DDM - outstanding points will be further addressed in future work.

4. Key Concepts/Patterns as Outcomes

It was noted earlier that one of the key decisions of the DDM project was to reduce the large potential large scale/complexity of the model by identifying the general patterns that underlie (a) different domains alongside (b) different aspects of the same domain. Two example patterns arising are now described relating to (a) temporal dissectiveness and (b) power classes. Brief vignettes are presented to demonstrate how issues arising from modelling in line with the metaphysical choices led to pattern development. Note is also made of how the syntax imposed by the Visual EXPRESS toolkit was 'adapted' to meet the conceptual demands.

4.1. Dissective and Non-dissective States

It is plain that objects have properties. Since Aristotle, there has been a distinction between accidental properties (which may change) and essential ones that do not. The way in which the ontology deals with this issue in the DDM can be explained through 'spatial dissectiveness' and the distinction between mass and count nouns. Goodman (1977) describes this property of 'dissectiveness', where when something is divided the resulting parts are of the same type as the whole. For example, if a batch of oil (a mass noun) is physically divided in two, then there are two batches of oil. Conversely, if a car (a count noun) is physically divided in two, then there is two halves of the car. In Goodman's terms, batches of oil are thus dissective and cars are not.

The 4D paradigm treats time on a par with space. Applying this choice to dissectiveness reveals that there is both spatial and temporal dissectiveness – where the count-mass distinction is a form of spatial dissectiveness. Dissectiveness also applies temporally however. For example, if one considers the first and second half of a pump pumping, then these halves are also pumpings – pumping is temporally dissective. In contrast, if the first and second halves of a project programme are considered, then these are not project programmes – project programmes are not temporally dissective. Similarly cars and pump equipment are also not temporally dissective. Within the DDM, temporally dissective individuals were recognised explicitly through the use of the ISO 15926-2 entity type **whole_life_individual**. Hence entities such as **possible_organization**, that are temporally non-dissective, are regarded as instances of 'whole-life individuals'.

The importance of dissectiveness is largely tied up with counting. Consider three objects:

- John,
- John minus a hair on his head,
- John minus a toenail.

John is a person. If person was spatially dissective, then ̃ 'John minus a hair on his head' and 'John minus a toenail' would also be persons. When considering the question "How many people are identified above?" we want to give the answer one but if person is dissective we have to give the answer three. By recognising that person is non-dissective, we are able to say that unless you have the maximal extent of a person, or a whole person in ordinary language, it is not a person, i.e. spatial parts of a person are not a person.

Now in a 4D ontology the same problem arises temporally as well as spatially. So if we consider the three objects:

- John's life,
- John's life minus the first day,
- John's life minus 31st July 2006,

we want to say that there is only one life here and not three. So life is non-dissective temporally (Strictly speaking, in the 4D paradigm, John and John's life are the same object as they have the same spatio-temporal extent).

When counting a mass (e.g. bits of oil) there is a need to specify a count-like container (e.g., a batch or a puddle). If this is not done, both the whole batch and its parts end up being counted. When counting things that are temporally dissective, such as pumpings, there is a need to specify that it is a pumping episode that occurs (the maximal connected pumping for a pump).

Recognising that there are both temporally dissective and non-dissective individuals had implications for the DDM. In a 4D ontology, accidental properties (in the Aristotelian sense) are temporal states of the 4D whole. Where the whole is temporally non-dissective (a **whole_life_individual**) these properties (temporal states) are potentially not of the same type as the whole – and so cannot be subsumed into the same type hierarchy. In the DDM, they are instead subsumed under a hierarchy, where if **X** is the whole life entity type, then **state_of_X** is the hierarchy of all states of **X**, dissective and non-dissective.

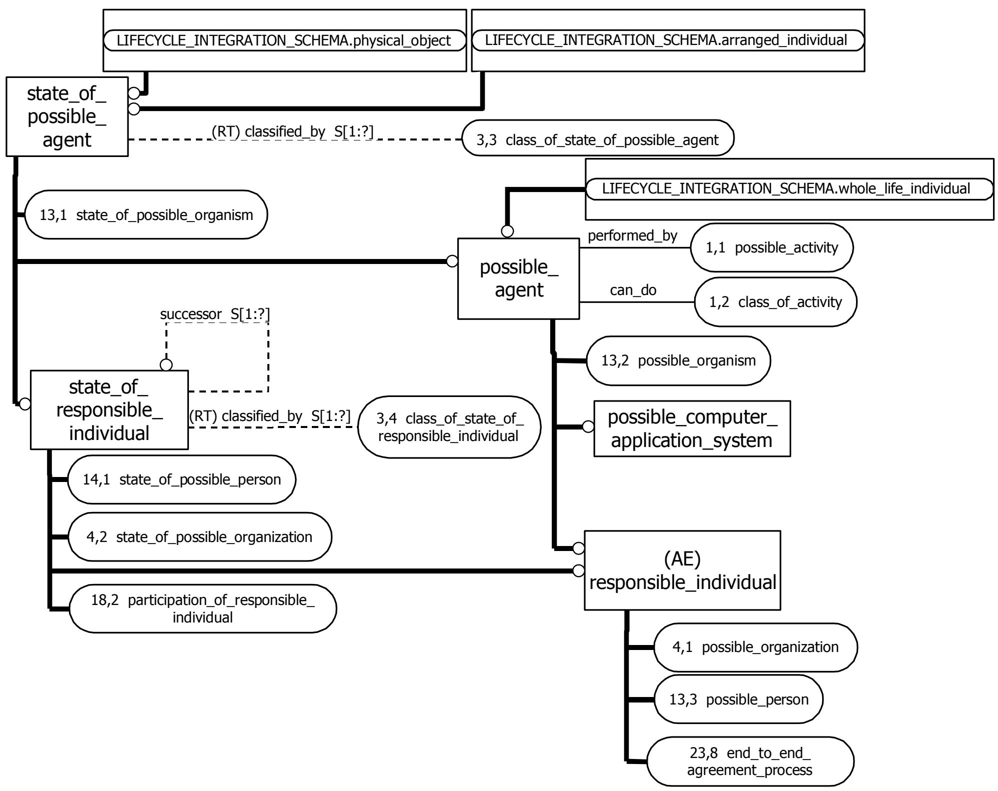

Figure 3: An example from Shell's DDM of dissective and non-dissective entity types.

This is illustrated in Figure 3 above. The entity type **state_of_responsible_individual** includes both non-dissective subtypes, such as **responsible_individual** and dissective subtypes such as **state_of_possible_organization** and **participation_of_responsible_ individual**. Figure 4 below shows that **participation_of_responsible_individual** is the supertype for a number of roles that may be played by **responsible_individual**s.

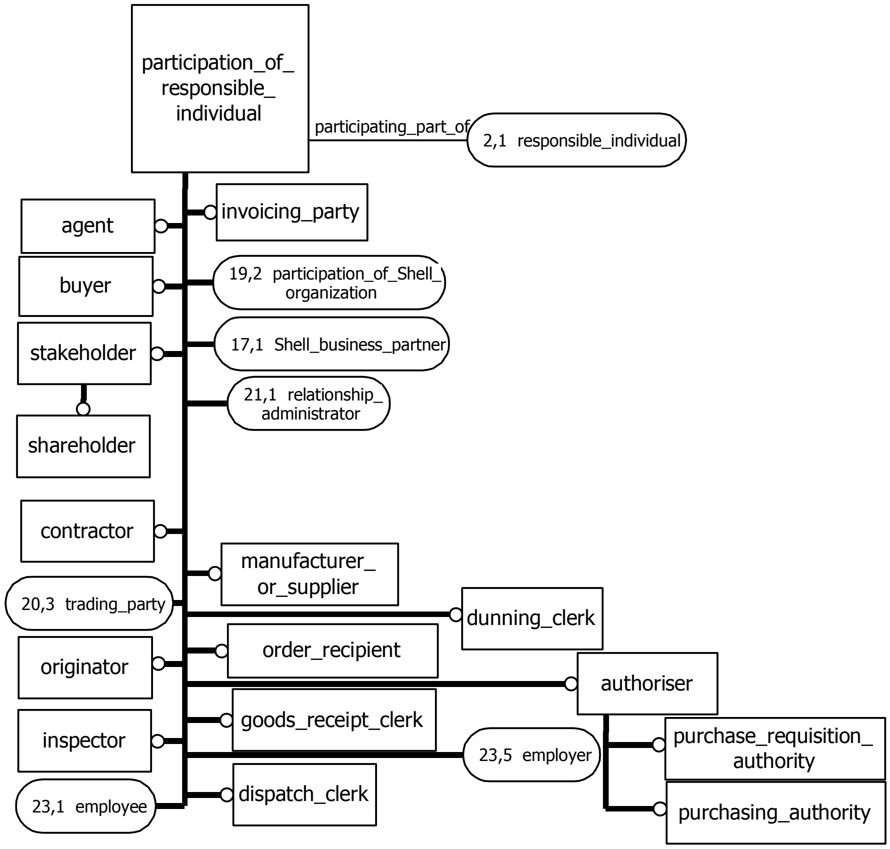

Figure 4: An excerpt from the DDM that shows roles as subtypes.

4.2. Powerclass

The powerclass of a class C is defined as the class of all possible subclasses of C. So for the class:

{a,b,c}

the powerclass is:

{{a},{b},{c},{a,b},{a,c},{b,c},{a,b,c}}.

In ISO 15926 there are a significant number (around a third) of entities with the name **class_of_x** and quite a few with the name **class_of_class_of_x**. While similar names can be false friends, they also often indicate an underlying similarity. The initial DDM analysis also came across a significant number of entities that were naturally named **class_of_x**. More detailed analysis of the 4D extensions of these entities revealed them to be (in most cases) the powerclasses of **X**. A common purpose for these entities was to be a basis for different classifications of things. So, for example, the different classifications of products (e.g. brand formulation classifications) were subtyped under class of product (in other words, product powerclass).

The methodological analysis led to a clarification of the nature of these entities and the understanding that they were examples of a general powerclass pattern. This understanding was documented explicitly in the DDM and is illustrated in Figure 5 below. Here **class_of_offering** is the powerclass of all offers to sell something. On the other hand, **intentional_class_of_offering** is just those classes of offerings that are those that were intentionally made (and not any arbitrary or accidental classes).

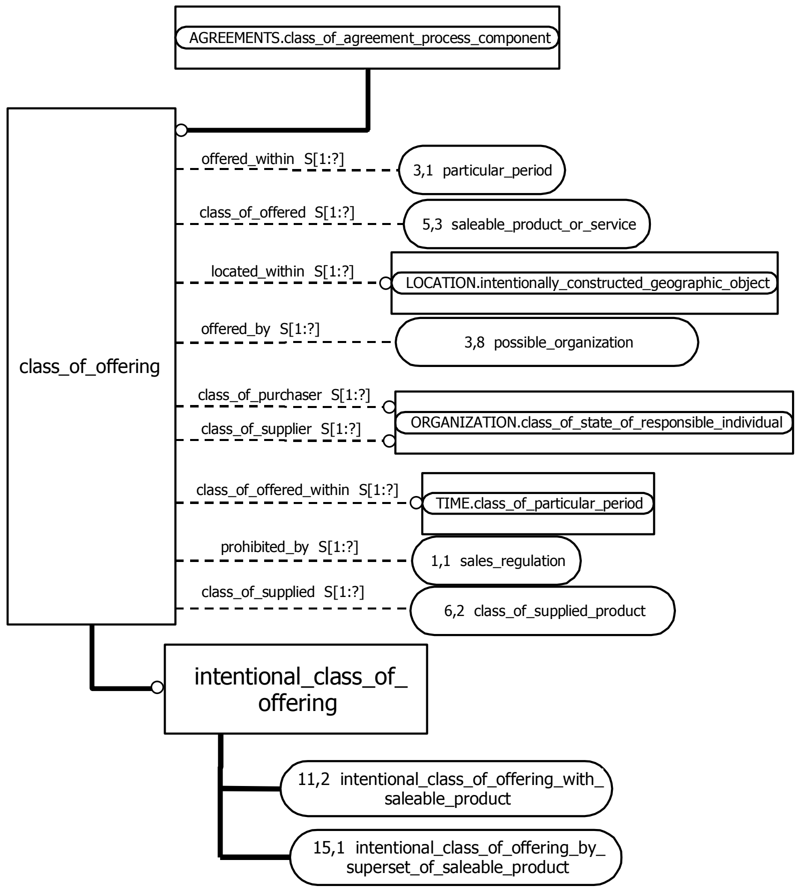

Figure 5: An excerpt from the DDM showing a powerclass and some non-powerclass subtypes.

5. Summary and Conclusions

This paper has sought to provide insight into the concrete industrial problem of developing a conceptual model that defines Shell's information requirements. The solution itself takes the form of the Downstream Data Model (DDM), which has its roots in a framework based on the notion of ontological commitment. Within the confines of the paper, we have sought to provide useful insights into the key strategic aspects relevant to in the creation and deployment of the DDM – primarily that related to the metaphysical choice made in relation to the use of 4D (extensionalism). Building on the explanation of the 4D approach, the work has provided examples of the impact of that choice in relation to two patterns that emerged from modelling Shell's information requirements. The first pattern relates to dealing with spatial and temporal dissectiveness in the model to more accurately express entities that are, or are not, dissective in space and time and to allow accurate analysis of those entities (for example though counting). The second pattern relates to the use of powerclasses as a means of providing an enriched classification mechanism – where the general explains the specific in a manner that reduces the number of entities required for example.

The value of the work lies in the fact that cursory analysis of systems within Shell demonstrates a mixed 'ontological bag' implicitly embedded in them. We believe that this implicit nature comes from (a) unawareness of the implications of making certain choices and (b) adherence to particular paradigms embedded in support technology (e.g. entity-attribute). The result, however, is a limited semantic understanding that costs the organization in integration terms. As an outcome, we believe that ontological understanding needs to be separated from both the 'epistemological gloss' and the 'implementational gloss' (though both are important, the former needs to be a foundation for the latter two). Emphasis on philosophical ontology is necessary to move the state-of-the-art in conceptual modelling forward. In conclusion, we do not seek to propose that any one set of ontological commitments are 'better' than others (though, clearly, we have a view); rather that an enhanced understanding of ontology is required in order that commitments can be laid bare and examined. Without such examination, conceptual models will continue to be 'point driven' and the drive for systems integration will be hampered by a limited understanding of semantics and their impact on systems development and integration. Our experience here is that collaboration between those involved in conceptual modelling on the information systems side of the fence and those involved in philosophical ontology is potentially fruitful. While we primarily work in the former camp, we note that this view is echoed on the other side of the fence

References

[1] Chengalur-Smith, I. N., Ballou, D. P. and Pazer, H. L. (1999) The Impact of Data Quality Information on Decision Making: An Exploratory Analysis, *IEEE Transactions on Knowledge and Data Engineering,* **11,** 853-864.

[2] Fisher, C. W. and Kingma, B. R. (2001) Criticality of Data Quality as Exemplified in Two Disasters, Information & Management, 39, 109-116.

[3] Goodman, N. (1977) The Structure of Appearance, R. Reidel Pub. Co., Dordrecht, Holland ; Boston.

[4] Gruber, T. R. (1993) A Translation Approach to Portable Ontologies, Knowledge Acquisition, 5, 199-220.

[5] Hay, D. C. (1996) Data Model Patterns: Conventions of Thought, Dorset House Publishing, New York.

[6] Heller, M. (1990) The Ontology of Physical Objects. Four Dimensional Hunks of Matter, Cambridge University Press.

[7] Hirschheim, R. A., Klein, H. K. and Lyytinen, K. (1995) Information Systems Development and Data Modeling: Conceptual and Philosophical Foundations, Cambridge University Press, Cambridge.

[8] ISO 10303-11 (1994) Industrial automation systems and integration – Product data representation and exchange – Part 11: Description methods: The EXPRESS language reference manual

[9] ISO 15926-2 (2003) Integration of lifecycle data for process plant including oil and gas production facilities: Part 2 – Data model

[10] March, S., Hevner, A. and Ram, S. (2000) Research Commentary: An Agenda for Information Technology Research in Heterogeneous and Distributed Environments, Information Systems Research, 11, 327-341.

[11] Moody, D. L. (1998) In The 17th International Conference on Conceptual Modeling, Vol. 1507 (Eds, Ling, T. W., Ram, S. and Lee, M. L.) Springer, Singapore, pp. 211-225.

[12] Moody, D. L. (2005) Theoretical and Practical Issues in Evaluating the Quality of Conceptual Models: Current State and Future Directions, Data & Knowledge Engineering, 55, 243-276.

[13] Nelson, J., Poels, G., Genero, M. and Piattini, M. (2005) Quality in Conceptual Modeling: Five Examples of the State-of-the-Art (Guest Editorial), Data & Knowledge Engineering, 55, 237-242.

[14] Orr, K. (1998) Data Quality and Systems Theory, Communications of the ACM, 41, 66-71.

[15] Park, J. and Ram, S. (2004) Information Systems Interoperability: What Lies Beneath?, ACM Transactions on Information Systems, 22, 595-632.

[16] Partridge, C. (1996) Business Objects: Re-Engineering for Re-Use, Butterworth-Heinemann, London.

[17] Partridge, C. (2002) The BORO Program, LADSEB CNR, Italy, Padova.

[18] Price, R. and Shanks, G. (2005) A Semiotic Information Quality Framework: Development and Comparative Analysis, Journal of Information Technology, 20, 88-102.

[19] Redman, T. C. (1995) Improve Data Quality for Competitive Advantage, Sloan Management Review, 36, 99-107.

[20] Redman, T. C. (1998) The Impact of Poor Data Quality on the Typical Enterprise, Communications of the ACM, 41, 79-82.

[21] Sider, T. (2001) Four-Dimensionalism : An Ontology of Persistence and Time, Clarendon Press ; Oxford University Press, Oxford New York.

[22] Simons, P. M. (1987) Parts : A Study in Ontology, Clarendon Press ; Oxford University Press, Oxford New York.

[23] Strong, D. M., Lee, Y. W. and Wang, R. Y. (1997) Data Quality in Context, Communications of the ACM, 40, 103-110.

[24] Strong, D. M. and Volkoff, O. (2004) A Roadmap for Enterprise Systems Implementation, IEEE Computer, 37, 22-29.
[25] Strong, D. M. and Volkoff, O. (2005) In 10th International Conference on Information QualityMIT, Cambridge, MA.
[26] Wand, Y. and Wang, R. Y. (1996) Anchoring Data Quality Dimensions in Ontological Foundations, *Communications of the ACM,* **39,** 86-95.

Acknowledgements

Developing a data model is a highly skilled job and to successfully undertake the development of an enterprise data model requires an exceptional team. We were fortunate to have access to a number of people with the necessary data modelling and project management skills, many with a background in developing International Standard data models. The members of that team are Ian Bailey, Tim King, Mark Lycett, Jenny Meader, Chris Partridge, David Price, Carol Rene, and Jan Sullivan. The team was lead by Matthew West and David Hay has participated as an external reviewer. It is duly noted that all team members have contributed directly or indirectly in one way or another to this paper and our thanks are extended to them.