Extending Upper Ontology Partitions to Address Contextually Determined Concepts

Michael G. Bennett[a1], Christof Hasse[b], Maxwell R Gillmore[c]

[a]*Hypercube Ltd.,* [b]*UBS.* [c]*Independent.*

Abstract. This paper describes the use and extension of a commonly used upper ontology partition, that relating to things that are typically defined in relation to a role or function, and applies this conceptualization to any subject matter that is contextually determined. An example is explored in detail in which client or customer data can be rationalized by the use of an ontology that is based on such partitions, showing how data in context-specific applications typically contains a mix of context-specific and non context-specific properties. More generally, the use of contextually determined concepts is used to make explicit the implicit contexts of data in a range of data sources, as a means to enable reusability of data across an organization.

Keywords: top level ontology, upper ontology, data integration, entity data, context, role, KR Lattice

1. Introduction

Most data models are highly contextual to the use case or requirement for which that data model was created. Such data models have one or more implicit contexts.

The motivation for this work is the need for data to be re-usable across multiple applications. Such re-use requires the use of a common ontology across the organization. This needs to be a purely conceptual ontology and it needs to be refined enough that the concepts reflected in different data models can co-exist within a common semantic structure. Such an ontology can be used to address the semantics of concepts across the organization, so that data elements in applications can be re-used in other business contexts or mapped to data elements in other applications, reporting systems and so on.

This points to the need for a well-defined conceptual framework. In this paper we consider one aspect of such a framework, specifically to support the range of context-specific concepts across the data assets of the organization.

---

[1] Corresponding Author: Michael Bennett; E-mail: mbennett@hypercube.co.uk

1.1 Addressing Context

There are several effects of context that are of relevance:
- Context may impact on the words used (vocabulary)
- Context may affect the way that someone conceptualizes the subject matter – e.g. the many conceptualizations of "Well" in the oil industry; Trade versus Deal in finance
- Concepts may themselves be contextual e.g. customers, actors, instruments and so on.

Terminological differences between different communities of users (also known as speech communities) cut across and often occlude distinctions of conceptualization. Different communities may well conceptualize the same subject matter in different, possibly incompatible ways. More often, they may conceptualize things differently but there is a way to distinguish between these concepts in the over-arching business ontology. For example, whether to consider an employee as a person who is employed by someone, or a person as the employee of a given company. In this paper we focus on concepts that are contextual by their nature.

2.1 Upper Ontology Support for Context

To support concepts that are contextual by their nature, we consider a common set of top level ontology partitions. These are given in Sowa's "Knowledge Representation Lattice" (KR Lattice) [1] as Independent, Relative and Mediating categories. Sowa defines these as follows:
- Independent categories are characterized by monadic predicates defined in terms of some entity x by itself (including its inherent parts and properties) and not in terms of anything external to x.
- Relative categories are characterized by dyadic predicates that relate an entity x to some external entity y that can exist independently of x.
- Mediating categories are characterized by triadic or higher predicates that show how an entity x mediates two or more entities ($y,z, . . .$) and thereby establishes new relationships among them.

This is the upper ontology adopted by the Enterprise Data Management Council [2] in the original development of the Financial Industry Business Ontology [3], although the partitions themselves are no longer retained in the published FIBO ontologies. The EDM Council has adapted the above definitions to give the sense of these partitioning concepts in a more business readable way, as follows:
- **Independent Thing:** a thing in its own right
- **Relative Thing:** A thing defined specifically and only in relation to some context.
- **Mediating Thing:** A thing which brings together two or more independent things into some relation, usually resulting in their being defined as Relative Things.

The examples given in this paper are based on that work.

In considering these matters we make a distinction between conceptualization and formalization. For example, what is formalized as a class of "Relative Thing" in the KR Lattice may be formalized in different ways in other ontologies such as the Basic Formal Ontology (BFO) [4] or OntoUML [5] but the concept remains that of some concept the definition of which is specific to some context. Many ontologies, including BFO, do not make the context explicit. In this work, the notion of Mediating Thing effectively is the context.

At its most general, the notion of a "Relative Thing" is anything, the definition of which is specific to some context. While these may often be understood in terms of party or role concepts, the range of things that are context-specific is much broader, encompassing actors, agents, tools, instruments, functional entities like banks or publishers and any other concepts that depends on a context for their semantics.

When we say that a given category of Relative Thing 'is' some category of Independent Thing (for example that Customer 'is' a person), this 'is' relation is not a type relation ('is-a' or 'sub class of') but is represented as a property of the Relative Thing, the range of which is the kind of Independent Thing that may perform that role or function. The categories of Independent Thing and of Relative Thing are effectively disjoint, and individuals of the latter are not for example persons, but are occurrences of a person (or whatever is the Independent Thing) performing that role or function.

To illustrate this, we take a simple kind of relative thing – the customer or client of an entity. This reflects a common data management problem, which is that the entity (for example a bank) has data about individual people, corporations and so on, in a wide range of different databases, in different formats and at different states of completeness, accuracy and so on. In order to unify the data about people or organizations or other kinds of legal entity, we will show that the data held in each data resource must be understood as being a mix of properties about the thing in and of itself, and properties of the thing in the specific context of that application or service.

2. Applying This to Customer Data

Consider a bank (B) and a legal person (P). The legal person may be an individual person or it may be an organization.

Over time the person and the bank have entered into a number of relationships, such as for different lending products, credit products and so on. There may also be cross-product relationships such as for support or for marketing of new products and services. In the case of a person that is itself a formal organization, there may also be relationships such as being the counterparty to various over-the-counter financial contracts, relationships in which some under-writing or guarantee is given, and so on. Figure 1 illustrates the points of interaction between the bank and the customer in relation to different products offered by the bank and taken up by the customer.
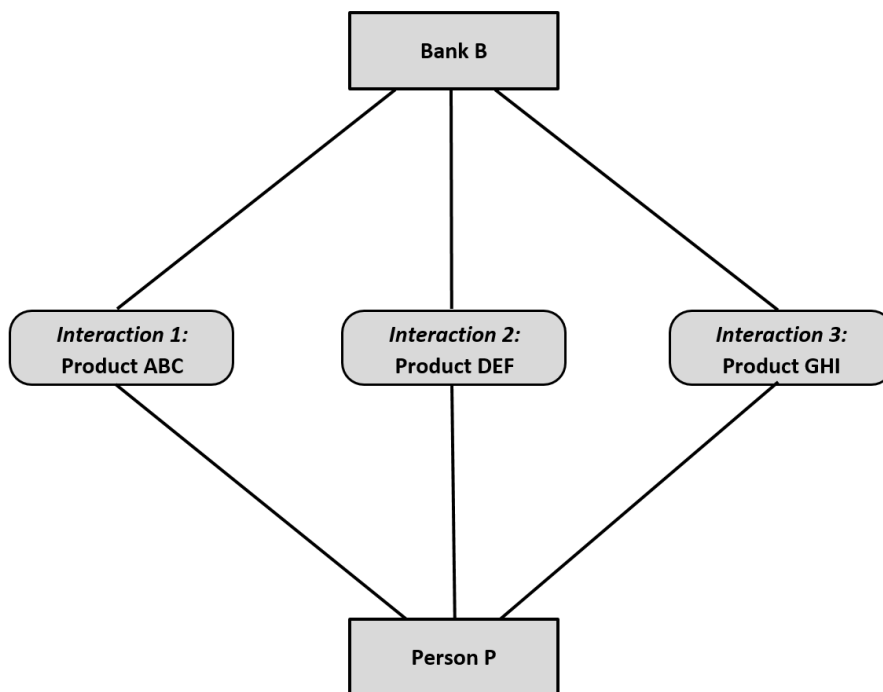
Figure 1: Interactions between Bank and Person

To keep this simple we will consider a person that is also an individual human being. This pattern can be extended to accommodate legally constituted organizations of various kinds, but that parts of the data will differ, since the properties intrinsic to business organizations are not all the same as those for humans; for example date of birth does not apply to an organization, while date of incorporation does not apply to a human – though a common parent property for both of these would be applicable globally.

Having considered the individual human being, we will see that some parts of this product relationships model will range across both humans and companies, for example both can be in the role of a borrower.

2.1 What the Data Looks Like

Bearing in mind that the data for each interaction will reside in one or more databases that are specific to that product, we can anticipate that each of these will contain a mix of properties that are intrinsic to the person, and properties that are specific to the relationship implied by the business context. Figure 2 gives some simplified examples of properties that one might find in the data.
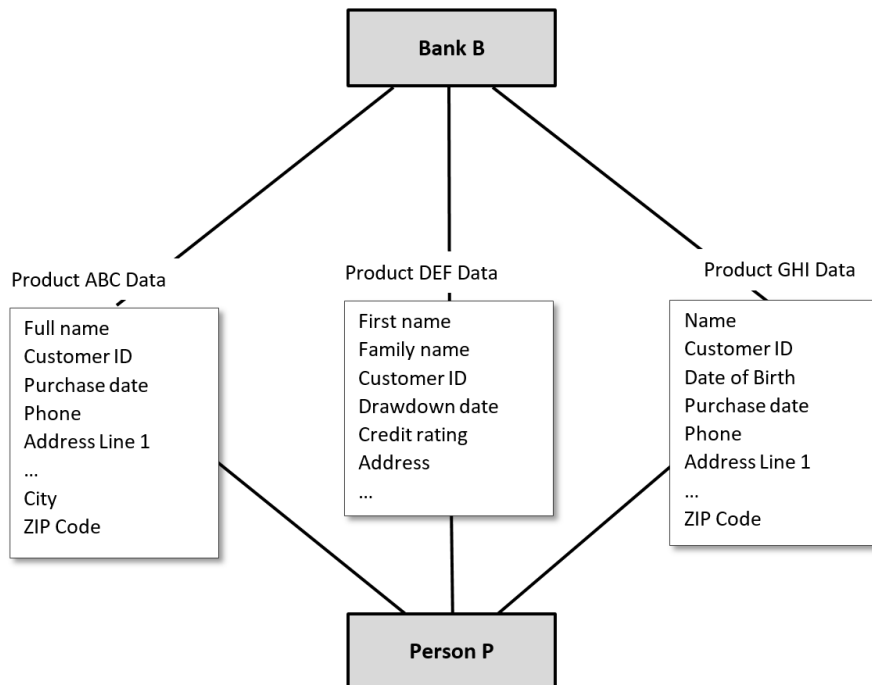
**Figure 2: Data for Bank and Person Relationships**

The first challenge for the over-arching conceptual ontology is that for each property in each data source, we need to segregate these two kinds of property.

On examination of these properties, it is clear that some of them are intrinsic to the person as a person (the Independent Thing), such as date of birth, while others are specific to the customer relationship or product (Relative Thing), such as product identifiers and customer relationship history.

The over-arching conceptual ontology would start by defining each of these product or customer relationships as kinds of Relative Thing. These could be referred to by the label 'Account' for one meaning of that word, this being the sense of an account relationship with the Person P, on the part of the Bank B. The corresponding Mediating Thing is the account relationship that the notion of "customer" represents.

Alternatively the label "Account" might be reserved for the Mediating Thing: an ontological representation of the relationship as seen in the round, with both the supplier (bank) and the customer being parties in roles within the context of that account. Note both of the above concepts may use the word 'account' in the sense used in 'the Smith account', which is distinct from the sense of 'account' in the context of general ledger accounts. It describes an account relationship.

Adapting Figure 1 to show the basics of the conceptual ontology gives us Figure 3.
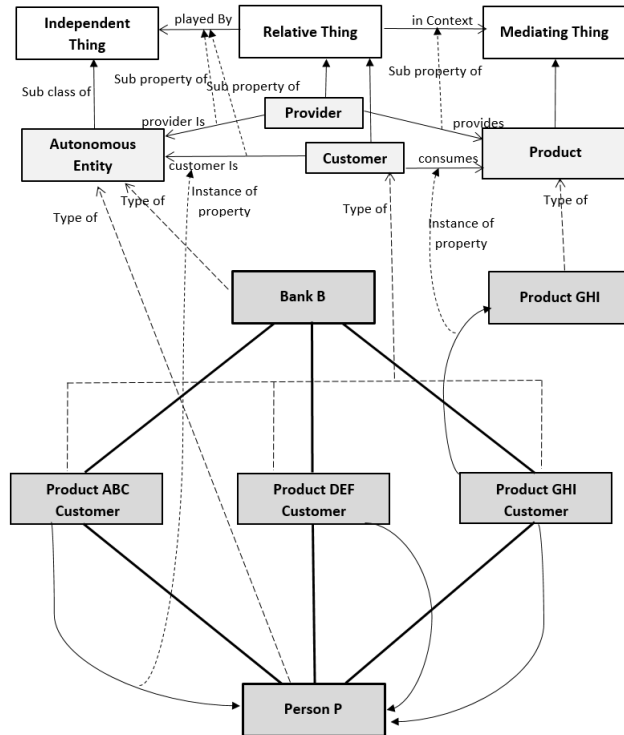
Figure 3: Ontology for the Individuals in Figure 1

We can then start to migrate the relevant kinds of data to where they belong in the ontology. In the case of data about the individual person (Independent Thing) there would be one set of properties defined in the ontology, so as to capture all of the kinds of things that are said about a person in any data held about them by the bank. This data would be re-factored so as to capture all of the kinds of information held by the bank about specific kinds of person, i.e. natural persons, organizations and so on.

To do this, we would look at each kind of data held in customer relationship date for each product. Where the information is always true of the person regardless of the specific product, this is moved to a set of data elements about persons, for example date of birth. Where the information is specific to the customer, this remains with the data about the customer in the context of that product relationship, for example customer identifiers for that product.

This is illustrated in Figure 4.
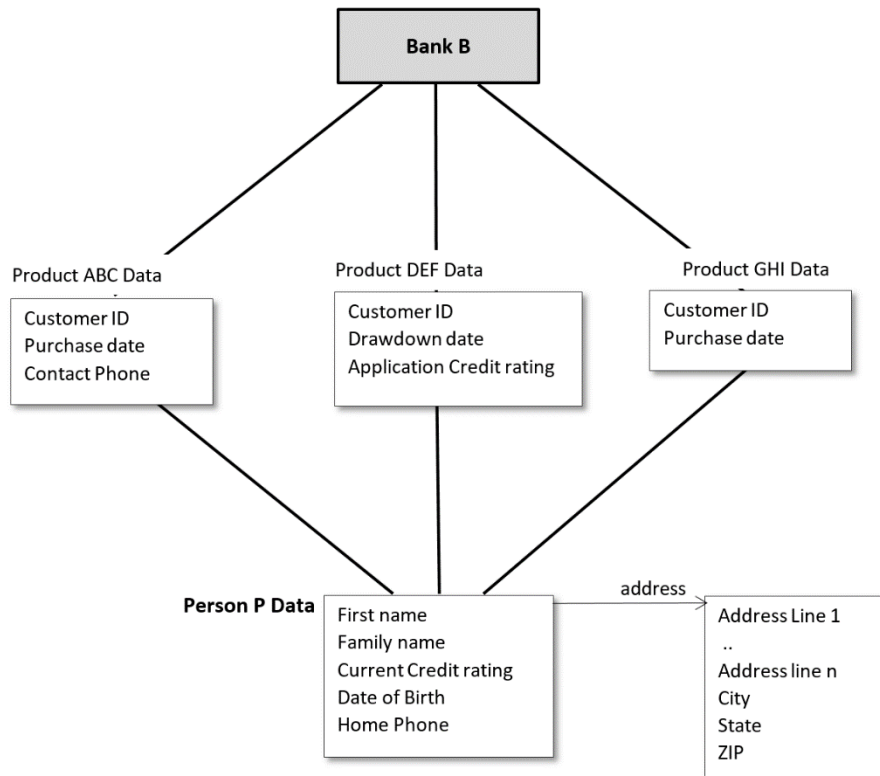
Figure 4: Refactoring Data for Bank and Person Relationships

Note how some properties introduced a question that had to be resolved in the above example. For example, if the customer gave the bank a phone number when they opened an account for a particular product, should the bank assume it is their main or only phone number, or that it was the one they wanted the bank to use as a contact for that product? This becomes important when the kind of person is an organization since different relationships typically have different contact points. Similarly, assuming that the credit rating for Product DEF was the rating at the time the product relationship was entered into, this likely won't be their current credit rating. The former is part of the history of the customer relationship for that product whereas the latter is something that can be said about the individual and at any given time should be the most recently available rating value at that time.

In this way we have done three things with the data that was available at each of the product specific application data structures:

1. Identified what things are always true about the entity in and of itself – and moved those to the entity database
2. Identified data elements that are specific to the customer account relationship

3. Identified data elements that, while specific to the account relationship, will need to be maintained going forward within the single set of data that is about the legal person in and of itself.

We also separated address information; a more complete ontology would allow us to use the same notion of "Relative Thing" to distinguish between for example delivery address, mailing address, company headquarters address, address for correspondence (which may remain as account-specific data) and so on. All of these are the kind of Independent Thing that is an address (some information structure the purpose of which is to refer to some real or virtual location, as modeled in FIBO). This arrangement is not shown in Figure 4.

3. Discussion

The pattern for "Customer" represents one narrow class of "Relative Thing" namely parties in formal contractual roles. These are endemic to much of finance and commerce, and include for example contract parties, securities issuers (these being specialized contract parties), transaction participants, lenders, underwriters and so on.

Other similar categories of relative thing include actors in activities or processes, and agents in some agentive context (for example calculation agents for derivatives calculations). These are not parties in roles but they are relative things. That is, they are relative to something (activity or agency in these cases), rather than being relative to some formal contractual role as in the case for parties in roles.

The real power of Relative Thing is that it defines anything that is contextually determined. Actors are just one example of a relative thing that is not a party to a formal business or contractual role. An asset is also a relative thing (relative to the context of ownership) for example. So is the underlyer for a derivative contract.

It is possible to take any context-dependent aspects of data in source material and make their context explicit (as a Mediating Thing), while re-framing those properties of a thing that are contextually determined, as being properties of the Relative Thing. As with the example of the customer data, there will also be properties of the independent thing that fulfils that role, function or other contextual matter, where those properties are true of the thing regardless of context.

There may also be properties of a relative thing that are defined within a broader context than that of the individual application or product but are still contextual. This points to the need for a hierarchy of kinds of Mediating Thing, from the more general to the more specific. For example a "Pilot" (Relative Thing) may be a pilot in the broad context of aviation, a commercial pilot, or the pilot of a given flight on a given day. Each narrower concept in which the pilot is defined, itself defines a narrower meaning of the term "Pilot" within that context. This suggests that there should be a hierarchy of kinds of context, from the broadest to the narrowest, each of them being the context in which some specific relative concept is defined.

Different business units may conceptualize the same subject matter differently. While some of the properties in a context-specific data model may not be intrinsic to the thing performing the role or function, but are properties of the thing in that context

or in a broader context, others are intrinsic to the independent thing itself. A third category to consider is things which are technically relative, such as a social security number or a pilot's license, where these things are relative to some jurisdiction or issuing authority, but where the widest range of applications for the data will not need to address this relativity. In such cases, the organization may have a conceptualization whereby the social security number, pilot's license etc. are treated as if they are intrinsic to the person or other independent thing. That is, the context is implicit in all applications of the data, rather than being expressed explicitly in the ontology. This approach should be used with caution – for example if an organization uses a social security number as a unique identifier for people, but wishes to expand into non-US markets where some other unique identification such as a passport number would be more appropriate.

One can think in terms of someone being an employee of a given company, or they may simply be employed. In this case, these are two possible conceptualizations of the same subject matter, and different data sources may reflect one or other conceptualization. By having all of these formalisms available in the unifying conceptual ontology it becomes possible to represent subject matter in each of the ways that the source data and stand-alone applications have elected to conceptualize and then formalize that subject matter. Adding explicit contexts to the representations of data structures that may previously have been considered as independent things, means that the organization will be able to unify what were previously incompatible micro-theories implicit in those data structures. Further exploration of context theory would broaden the range of ways in which this thinking can be applied.

The supporting top level ontology should have enough basic concepts that these contextual distinctions can be explicitly modeled. To the extent that the ontological nature of each contextualization can be reflected by some formalization that is situated within the upper ontology matter, it should be possible to represent separate, incompatible conceptualizations within the same model. This is in preference to having localized "micro theories". In this way data about related items may be re-used where appropriate across these disciplines or functional areas.

References

1. Sowa, J.F.: Knowledge Representation, Logical, Philosophical and Computational Foundations. Brooks/Cole, Pacific Grove, California (2000)
2. The Enterprise Data Management Council. Available at www.edmcouncil.org
3. The Financial Industry Business Ontology (FIBO) Available from spec.edmcouncil.org/fibo
4. Institute for Formal Ontology and Medical Information Science (IFOMIS): The Basic Formal Ontology (BFO) Available at http://ifomis.uni-saarland.de/bfo/
5. Menthor: OntoUML, available at http://www.menthor.net/ontouml.html