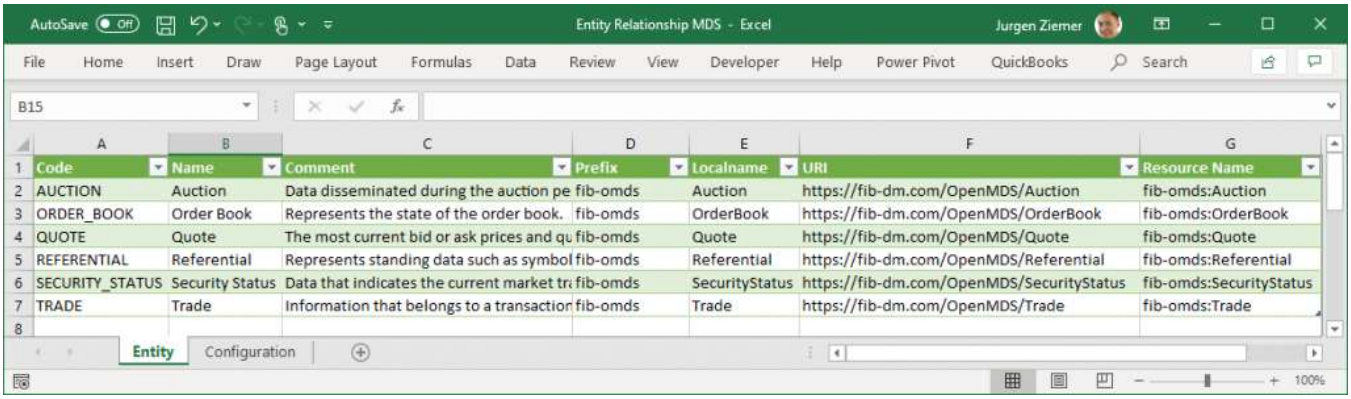
Entity-Relationship Metadata Sets

The generic Entity-Relationship Metadata Set is a bridge from modeling tool-specific to ontology metadata set. First, it replaces tool-specific metamodel names with common names. For example, PowerDesigner has Inheritances, Sparx EA has Generalizations, the E/R metadata set uses the term Subtype.



	Code	Name	Comment	Prefix	Localname	URI	Resource Name
2	AUCTION	Auction	Data disseminated during the auction pe	fib-omds	Auction	https://fib-dm.com/OpenMDS/Auction	fib-omds:Auction
3	ORDER_BOOK	Order Book	Represents the state of the order book.	fib-omds	OrderBook	https://fib-dm.com/OpenMDS/OrderBook	fib-omds:OrderBook
4	QUOTE	Quote	The most current bid or ask prices and qu	fib-omds	Quote	https://fib-dm.com/OpenMDS/Quote	fib-omds:Quote
5	REFERENTIAL	Referential	Represents standing data such as symbol	fib-omds	Referential	https://fib-dm.com/OpenMDS/Referential	fib-omds:Referential
6	SECURITY_STATUS	Security Status	Data that indicates the current market tr	fib-omds	SecurityStatus	https://fib-dm.com/OpenMDS/SecurityStatus	fib-omds:SecurityStatus
7	TRADE	Trade	Information that belongs to a transaction	fib-omds	Trade	https://fib-dm.com/OpenMDS/Trade	fib-omds:Trade

CODT Reverse-Engineered Entity Relationship Metadata Set

The E/R Entity Metadata Set add columns for Prefix, Localname, URI, and Resource Name. We must configure the base URI for our target ontology, here “https://fib-dm.com/OpenMDS/”, and its abbreviation, the Prefix, “fib-omds”. The Localname is a simple “UnCamel” string function of the logical data model entity name, and the Resource Name concatenates Prefix, a colon, and the Localname.

Ontology Metadata Sets

The Ontology Metadata Sets depicted in the LDM diagram (patent drawing 14) are an isomorphic representation of an RDF/OWL metamodel, but for reverse-engineering we only populate the common subset. For example, the LDM does not have a concept of Inverse Object Properties and Equivalent Classes.

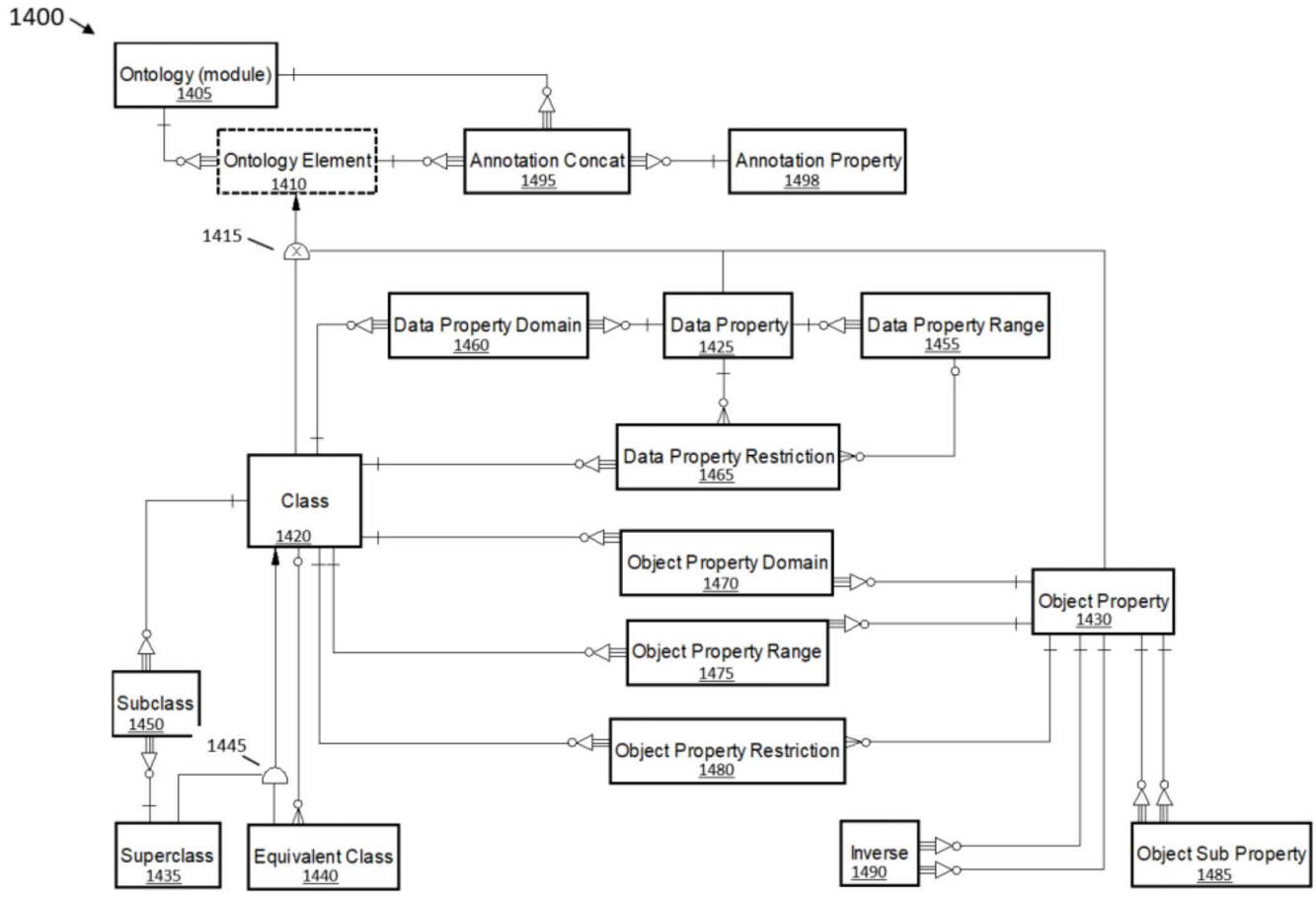
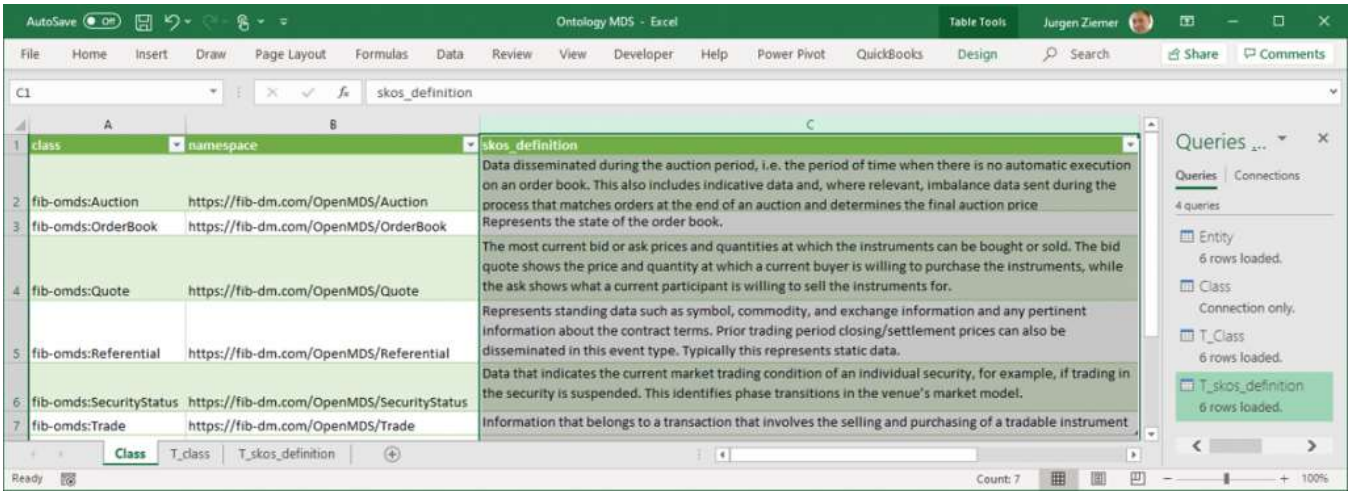


FIG. 14

CODT Patent drawing FIG 14 – Ontology Logical Data Model

The Ontologist may add these higher semantics after importing the base ontology.

The Class Metadata Set below populates from the E/R Entity Metadata Set. Again, reverse-engineered Metadata Sets may have fewer columns than those extracted from the ontology. For example, the average data model may only have a single documentation item, per default transformed into SKOS Definition.



CODT Reverse-Engineered Ontology Metadata Set

In order to Load the ontology metadata set, we must break it down into triple of subject, predicate, and object.

The T-tabs above are Power Queries sourcing the Ontology Metadata Set. The T_Class data set below is the raw data to CONSTRUCT or bulk-load triples for classes.

```
CONSTRUCT {
fib-omds:Auction rdf:type owl:Class .
}
WHERE {}
```

Or we can bulk-insert a CSV file below.

subject	predicate	object
fib-omds:Auction	rdf:type	owl:Class
fib-omds:OrderBook	rdf:type	owl:Class
fib-omds:Quote	rdf:type	owl:Class
fib-omds:SecurityStatus	rdf:type	owl:Class
fib-omds:Trade	rdf:type	owl:Class

CONSTRUCT triples for Classes

Note that the CONSTRUCT triples match the joins in the SPARQL SELECT queries, extracting metadata from the ontology.

```
# Owl Classes.rq
SELECT ?class ?qname ?namespace ?skos_definition
WHERE {
  ?class a owl:Class .
# ...
OPTIONAL {
  ?class skos:definition ?skos_definition}
}
```

Likewise, we break down triples for the Class Definition annotation property.

subject	predicate	object
fib-omds:Auction	skos:definition	Data disseminated during the auction period, i.e. the period of time when there is no automatic execution on an order book. This also includes indicative data and, where relevant, imbalance data sent during the process that matches orders at the end of an auction and determines the final auction price
fib-omds:OrderBook	skos:definition	Represents the state of the order book.
fib-omds:Quote	skos:definition	The most current bid or ask prices and quantities at which the instruments can be bought or sold. The bid quote shows the price and quantity at which a current buyer is willing to purchase the instruments, while the ask shows what a current participant is willing to sell the instruments for.
fib-omds:SecurityStatus	skos:definition	Data that indicates the current market trading condition of an individual security, for example, if trading in the security is suspended. This identifies phase transitions in the venue's market model.

subject	predicate	object
fib-omds:Trade	skos:definition	Information that belongs to a transaction that involves the selling and purchasing of a tradable instrument

CONSTRUCT triples for Class Definitions

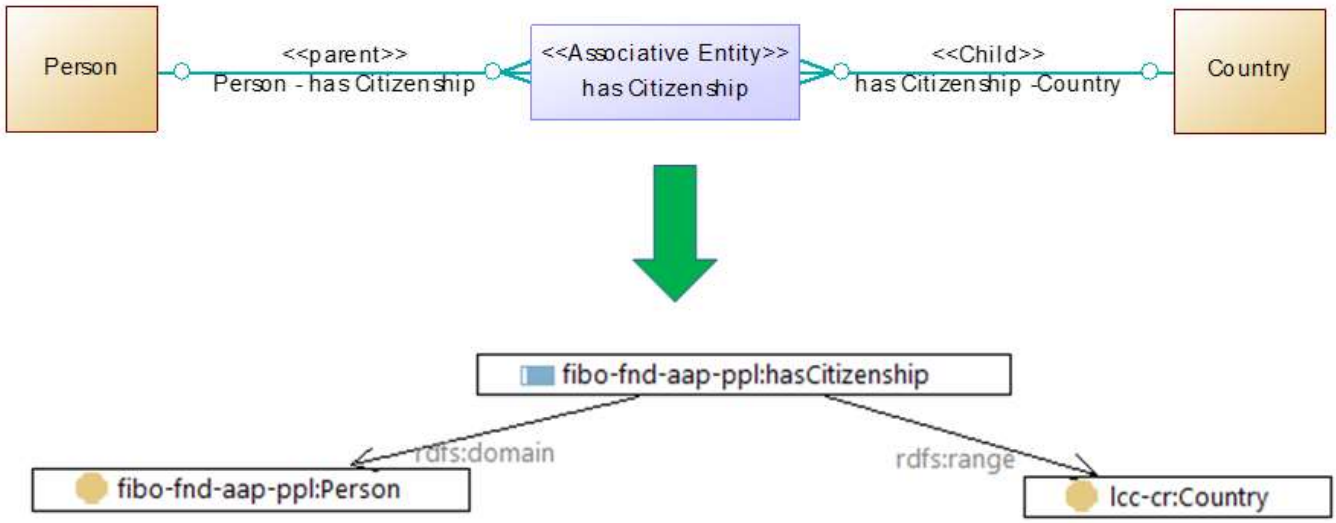
Just like PowerDesigner import and list report proved the existence isomorphic transformation, so does SPARQL: If we extract metadata with SELECT data, we can also load the same metadata using CONSTRUCT.
Unfortunately, ontology editors lack the powerful comparison report functionality of the data modeling tools.

Associative Entities are Object Properties

My third article made a detailed case that ontology Object Properties are Associative Entities – not Relationships. How about the reverse transformation?

The conventional wisdom derives object properties from data model relationships or database foreign keys, and **all** Entities/Tables transform into ontology classes. Many data models don't have named relationships, and modeling tool generated names, like "Relationship_1," don't provide meaningful Object Property Localnames.

Per default configuration, CODT transforms object properties into associative entities. For the isomorphism to hold, CODT must reverse-engineer these associative entities back into object properties.



CODT Associative Entity to Object Property Transformation

The easiest configuration examines the entity stereotype. CODT generated entities have a stereotype <<Associative Entity>>, and for those, we populate the Object Property metadata set and derive class restrictions from the relationships of the Associative Entity. Data Modelers extending a CODT derived data model, like FIB-DM, should set the stereotype for new Associative Entities.

Modelers should configure the Stereotype on Associative Entities for Data Models; they want to onboard onto the Knowledge Graph or Enterprise Ontology.

Can CODT detect Associative Entities? The standard Associative Entity resolves a many-to-many data model relationship. In this case, the associative entity participates as a child entity in exactly two identifying relationships.

However, some Associative Entities may have more than two related base entities, for example, the classic Project, Resource, Role pattern.

Criteria may stipulate that Associative Entities don't have attributes. If they do, then we need a class to hold them as data properties.

Every modeling standard and every data model is different. Data Architects and Ontologist should analyze the source model to determine the CODT optimal configuration. It fine to experiment and test different ways to reverse engineer the data model.

Conclusion

Logical and Conceptual Data Models are a better source to reverse-engineer ontologies than database schemas. We examined the use cases for operational-, enterprise-ontologies, and knowledge graphs. The isomorphism between the common metamodel subset of ontology and data model extends to their metadata set representations. Metadata Sets are bi-directional, and hence CODT can forward and reverse engineer. Associative Entities should transform into object properties. However, it remains a challenge to identify them.

The CODT transformation is the best way to transform data models into ontologies.

References

The canonical version of this article: https://fib-dm.com/data-model-reverse-engineered-into-ontology/

Join the discussion on LinkedIn: https://www.linkedin.com/pulse/data-model-transformed-ontology-jurgen-ziemer/

FIB-DM articles

1. Finance Domain ontology transformed into an Enterprise Data Model
2. Ontology Class- and Data Model Entity-hierarchy, are they the same?
3. Ontology Object Properties are Data Model Associative Entities – not Relationships

PowerPoints and videos

Semantics for extra-large banks, the CODT POC class/tutorial

Bank Call Report in FIBO, 2017 Enterprise Data World Conference