NATIONAL DIGITAL TWIN PROGRAMME

The Approach to Develop the Foundation Data Model for the Information Management Framework

cdbb
Centre for Digital Built Britain

Executive Summary

Industries involved in the creation and management of built assets require effective, resilient and secure data and information sharing and aggregation. Further, consistent information is vital to draw data from multiple sectors and domains and to permit rapid evaluations, enhanced decision-taking and faster responses.

As a result, a formal mechanism to ensure that the right information can be made available at the right time, to the right people and that the quality of the information is known and understood, is required.

The Information Management Framework (IMF) is a common language by which consistent data and information can be communicated in a way that is effective, resilient and secure.

The technical part of the IMF comprises three main elements:

- A Foundation Data Model
- A Reference Data Library, and
- An Integration Architecture

This document identifies the pragmatic and technical requirements for the Foundation Data Model, considers whether any existing Top-Level Ontologies could be used as a suitable start-point, and determines the approach to be taken in its development.

In arriving at the technical requirements for the Foundation Data Model, we first considered the pragmatic requirements that needed to be met. These are:

1. It will need to reflect the world of science and engineering
2. It will need to be extensible.
3. It will need to be stable.
4. It will need to be capable of consistent extension by independent teams.

This resulted in the following technical requirements:

1. A principled and rigorous approach
2. Rooted in a science and engineering view of the world
3. A comprehensive scope
4. As simple as possible (but no simpler)

These in turn gave rise to more detailed technical requirements, which we used to classify the Top-Level Ontologies we have been able to identify.

There are four Top-Level Ontologies that meet all the technical requirements: BORO, IDEAS, HQDM and ISO 15926-2. They are distinct from the other Top-Level Ontologies in being 4-dimensionalist. They are otherwise also all closely related in that they all take BORO’s 4-dimensionalist foundation as their starting point, although they have been developed with different purposes in mind from there.

It is recommended that the Foundation Data Model seed is developed from the 4-dimensionalist Top-Level Ontologies: BORO, IDEAS, HQDM and ISO 15926-2 that best met the technical requirements we identified as relevant to the Foundation Data Model. This should be underpinned by rigorously establishing the foundations for the Top-Level Ontology developed.

Introduction

Industries involved in the creation and management of built assets require effective, resilient and secure data and information sharing and aggregation. Further, consistent information is vital to draw data from multiple sectors and domains and to permit rapid evaluations, enhanced decision-taking and faster responses.

As a result, a formal mechanism to ensure that the right information can be made available at the right time, to the right people and that the quality of the information is known and understood, is required.

The Information Management Framework (IMF) is a common language by which consistent data and information can be communicated in a way that is effective, resilient and secure.

The technical part of the IMF comprises three main elements:

- A Foundation Data Model
- A Reference Data Library, and
- An Integration Architecture.

The Foundation Data Model and Reference Data Library together provide a language, an inter-lingua, so that data can be shared consistently and used to support decisions without requiring any further "data wrangling". The Integration Architecture provides the transport mechanisms, together with authorisation and security protocols, to ensure that information can be accessed seamlessly, but only by those authorised to do so.

The division between the Foundation Data Model and Reference Data Library can be compared to the division between the sentence structures that can be used to say things, and the words used in the sentence structures. Together the Foundation Data Model and Reference Data Library make an ontology, a theory of what exists, and at the core of the Foundation Data Model is a Top-Level Ontology – the essential categories on which everything else is based.

Previous work discovered and characterized thirty-six Top-Level Ontologies[1], so that potential users can assess the suitability of those Top-Level Ontologies for their particular purpose.

The purpose of this document is to set out, at a relatively high level, the Top-Level Ontology requirements for our Foundation Data Model. This process has allowed us to establish which candidate Top-Level Ontologies best meet those requirements, and to determine the approach for how to proceed in adopting, adapting, or developing the Top-Level Ontology for our Foundation Data Model.

More detailed information on the requirements and the process followed is set out in the Top-Level Ontology Survey Document.

1 Reference to Top-Level Ontology Survey Document

Requirements for the Top-Level Ontology of the Foundation Data Model

We start by setting out the pragmatic requirements we need our Foundation Data Model to meet, and then determine what technical requirements this means we seek to satisfy in a Top-Level Ontology.

Some pragmatic requirements for our Top-Level Ontology

We know several things about our Foundation Data Model that are particular to our purpose:

1. **It will need to be rooted in the world of science and engineering**

   Different Top-Level Ontologies are developed with different purposes. Some are developed to reflect science and engineering; others are developed with a linguistic or common sense approach reflecting how the world is perceived. In order to manage information relating to built assets, we will need a Top-Level Ontology rooted in the reality of science and engineering. Some philosophers call such Ontologies that seek to reflect reality as it is Foundational Ontologies[2].

2. **It will need to be extensible.**

   We cannot identify all the requirements for the Foundation Data Model at the outset: trying to identify them would be an exercise in boiling the ocean. So, we need a Foundation Data Model that is set up to be extensible at the outset so it can expand to meet additional requirements as they arise. Furthermore, we know that the scope is likely to broaden and deepen. More information on the coverage requirements, both general and specific, are identified in "The Pathway towards an Information Management Framework".

3. **It will need to be stable.**

   When requirements change, problems can arise because changes to your existing data model may be required as well as additions. This is expensive for those who are using the data model as they have to restructure their data, and as a result, in practice, some organizations do not use the latest version which means consistency is lost, defeating the object of the exercise. So, we need a Foundation Data Model that takes account of the broad scope so it can be stable as it expands to meet changes in requirements (i.e. append only).

2 Again, note that some use foundational for what we are calling ontological.

4. **It will need to be capable of consistent extension by independent teams.**

   The Foundation Data Model and Reference Data Library are part of one whole; however, it will not be practical for all this to be controlled by a "single mind". Sufficient principles and guidance are necessary so that teams of different people can work independently yet come up with results that are consistent.

Some technical requirements for our Top-Level Ontology

The pragmatic requirements set out above mean we can identify some technical requirements a Top-Level Ontology needs to meet. An overview of these is given here, more detail can be found in the full survey document.

A principled and rigorous approach

In order to assure extensibility, stability and independent development, the Top-Level Ontology must take a top-down approach that sets out the principles of development and consistency regarding what kinds of things exist. In philosophy and Knowledge Representation such Top-Level Ontologies are known as **ontological**[3]. Alternative approaches that are developed purely bottom-up without overall guiding principles are known as **generic**.

For Top-Level Ontologies that do commit to ontological principles, there are different levels of commitment. The more commitments made by the Top-Level Ontology, the less wriggle room there is for variability and misunderstanding. So, for the Foundation Data Model a **highly committed** Top-Level Ontology is preferred.

Rooted in a science and engineering view of the world

As stated above, our Top-Level Ontology must be rooted in science and engineering, rather than for example a common sense or linguistic view of the world. Some philosophers call such Ontologies that seek to reflect reality as it is **Foundational Ontologies**[4].

3 There are however also some that refer to these as foundational.
4 https://www.cdbb.cam.ac.uk/what-we-do/national-digital-twin-programme/resources-top-level-ontologies-and-industry-data-models

A comprehensive scope

The only way we can ensure that our unknown but expanding scope can be accommodated is to set our scope as "life, the universe and everything,"[5] meaning that it should support anything valid that can be said formally. Philosophers call such Top-Level Ontologies **categorical**. Sometimes claims of being categorical can be overly optimistic, so we have used the Pathway document coverage requirements as a pragmatic check on claims of being sufficiently categorical.

As simple as possible (but no simpler)

In addition to being able to say anything that is valid, we will want there to be (as far as possible) only one way to say things. More than one way to say things introduces the need to translate between them, which adds complexity. Further, we will want our ontology to be as compact and simple as possible.

"A Survey of Top-Level Ontologies - to inform the ontological choices for a Foundation Data Model"[6] sets out an approach to determining simplicity, which is summarised here.

Philosophers consider ontological simplicity to have three elements:

1. **Parsimony**

   Parsimony is usually taken to be defined by Ockham's razor – *entities are not to be multiplied beyond necessity*. This can be accounted for, to some extent by counting the entities in the ontology. However, if fundamental entities can be distinguished from derived entities, then it is sufficient that *fundamental entities are not to be multiplied beyond necessity*. In this case, we account for (count) fundamental entities only - derived entities are a 'free' (or significantly less costly) lunch.[7]

2. **Explanatory Sufficiency**

   In regard to minimizing entities Kant said: *The variety of entities should not be rashly diminished*. So the Top-Level Ontology must have sufficient entities to cover what we are interested in, which we have already said is everything. This means we need a Top-Level Ontology that is **categorical**.

3. **Fruitfulness**

   While we want to be parsimonious in fundamental types, it is a good thing when derivative entities arise from the fundamental entities, and this is called being fruitful. It shows that these fundamental entities can be used to produce something useful. However, you do not want to over generate: when the additional entities are profligate or promiscuous, serving no useful purpose.

5 "The Hitchhikers' Guide to the Galaxy" Douglas Adams
6 https://www.cdbb.cam.ac.uk/what-we-do/national-digital-twin-programme/resources-top-level-ontologies-and-industry-data-models
7 In this we follow Schaffer who suggests that fundamental parsimony is where the priority should lie, and that an abundant view of what there is with a restrictive view of what is fundamental is where simplicity and fruitfulness lies. Related to this is a preference for plenitude: for not placing unnecessary constraints on what can exist – if it is possible for something to exist, then it does. As he notes, both classical mereology and (impure) set theory exhibit this.

A particular kind of ontological choice results in **horizontal stratification**, meaning that there is a choice that can be made to divide one fundamental type into two. Horizontal stratification arises if the choice is made. This will tend to multiply fundamental entities, as well as relationships between them. From an accounting perspective, one needs to justify the increase in fundamental entities.

A particular case of horizontal stratification is whether to make physical objects and activities distinct (non-overlapping) or not. An overlapping strategy says that the same thing can be both an activity and a physical object, making a distinction means that there must be two things, an activity and a physical object, and then a further relationship between them.

A practical example of the consequences is an abnormal load on the highway. In the UK, an 'abnormal load' is a vehicle that has any of the following:

- a weight of more than 44,000kg
- an axle load of more than 10,000kg for a single non-driving axle and 11,500kg for a single driving axle
- a width of more than 2.9 metres
- a rigid length of more than 18.65 metres

If you have an abnormal load, then there are a whole series of actions you need to take including providing a plan to authorities of what route you are taking and when the movement will take place.

What we need to do now is tell a story for a particular abnormal load so we can see what we get.

The implied definition of "vehicle" above includes any load the vehicle is carrying, rather than treating the vehicle as one thing and the load as another. With that in mind, here is a story...

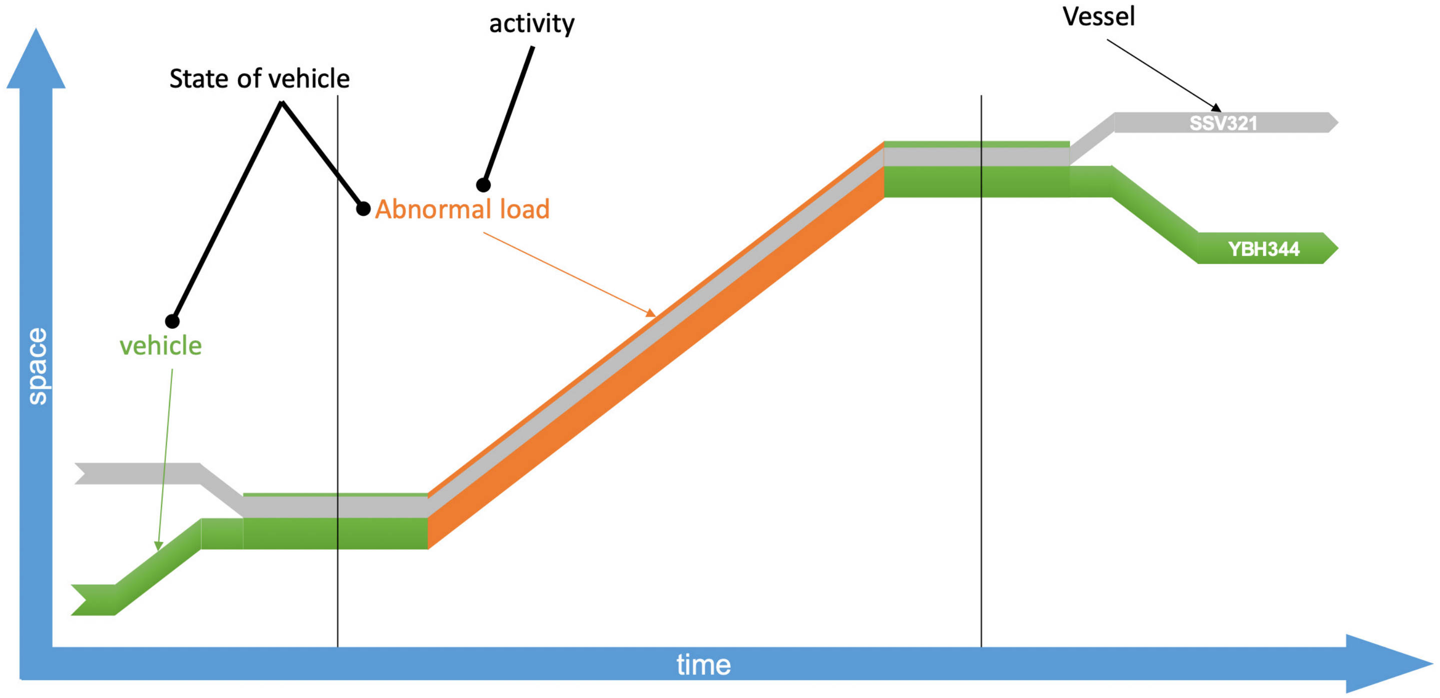

Figure 1: Space-time diagram for an abnormal load.

A vehicle, reg no YBH344, sets off from its depot without a load, so it is a “normal load” and can travel freely. It travels (activity) to the place where the load is picked up, a factory on private land. The load is a stainless steel vessel with manufacturers serial number SSV321. The vehicle is loaded at the end of the day and left on site overnight.

The vehicle has undergone a change in physical state, a part (the load, SSV321) has been added to the vehicle and it now has different weight and dimensions, when measured they meet one or more of the conditions for an abnormal load. So, in the morning when it enters the highway system, it becomes an abnormal load and abnormal load regulations apply. When it arrives at its destination, it leaves the highway and ceases to be an abnormal load. The vehicle is unloaded the next day which again changes the physical state of the vehicle so that it is no longer an abnormal load when on the highway.

A Top-Level Ontology that unifies entities so that activities are overlapping with physical objects, will find that there is one thing that is an abnormal load that is both an activity and a state of vehicle. A Top-Level Ontology that stratifies physical objects and activities will require that there are two objects, an abnormal load as a state or property of vehicle, and an abnormal load activity for the movement of the vehicle. In addition it will need a relationship that says the abnormal load state of the vehicle is a participant in the abnormal load movement activity. This clearly has more entities.

The point is that a decision to stratify the physical object from its activities forces us to separate information that can be kept together. In this case, we want to know when the vehicle is an abnormal load, but in the stratified version, we also need a different object (the activity) for some of the information. In the non-stratified version, the temporal part of the vehicle is also the activity, so all of the information is kept together.

“An Initial Review of Top-Level Ontologies” identifies which Top-Level Ontologies exhibit horizontal stratification, and those with no horizontal stratification are preferred.

Implicit entities

Ontologies are typically built through the careful manual addition of references to entities. However, some Top-Level Ontologies include algorithms for automatically adding new references to entities that are implied by existing ones. This is known as **formal generation** and can open up possibilities for fruitfulness.

Two cases of formal generation are mereological fusion of two particulars and the complement in set theory. As Figure 2 shows, the formal generation produces both the object and its hierarchical relation(s).

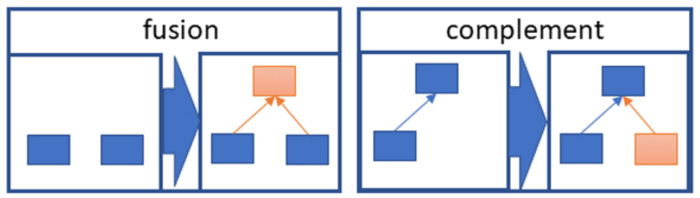

Figure 2 – Two kinds of formal generation

Adopting formal generation can be seen as an example of plenitude; all possible applications of the rule are automatically allowed. For example, both classical mereology and set theory automatically allow formally generated entities.

Together these technical requirements give us a practical basis for assessing the Top-Level Ontologies for our purpose as represented in the pragmatic requirements given above.

Characterization and assessment of available Top-Level Ontologies

The list of Top-Level Ontologies discovered by the survey are given in Appendix A. The survey also identified, where possible, the characteristics of the Top-Level Ontologies so they could be classified as to whether they met particular technical requirements or not.

Figure 3 shows a Venn diagram of these classified by the technical requirements outlined above that are important for our purposes, such that those that meet the most requirements are in the most categories.

There are four Top-Level Ontologies that meet all the technical requirements: BORO, IDEAS, HQDM and ISO 15926-2. They are distinct from the other Top-Level Ontologies in being 4-dimensional. They are otherwise also all closely related in that they all take the BORO 4-dimensional foundation as their starting point but have been developed with different purposes in mind from there.

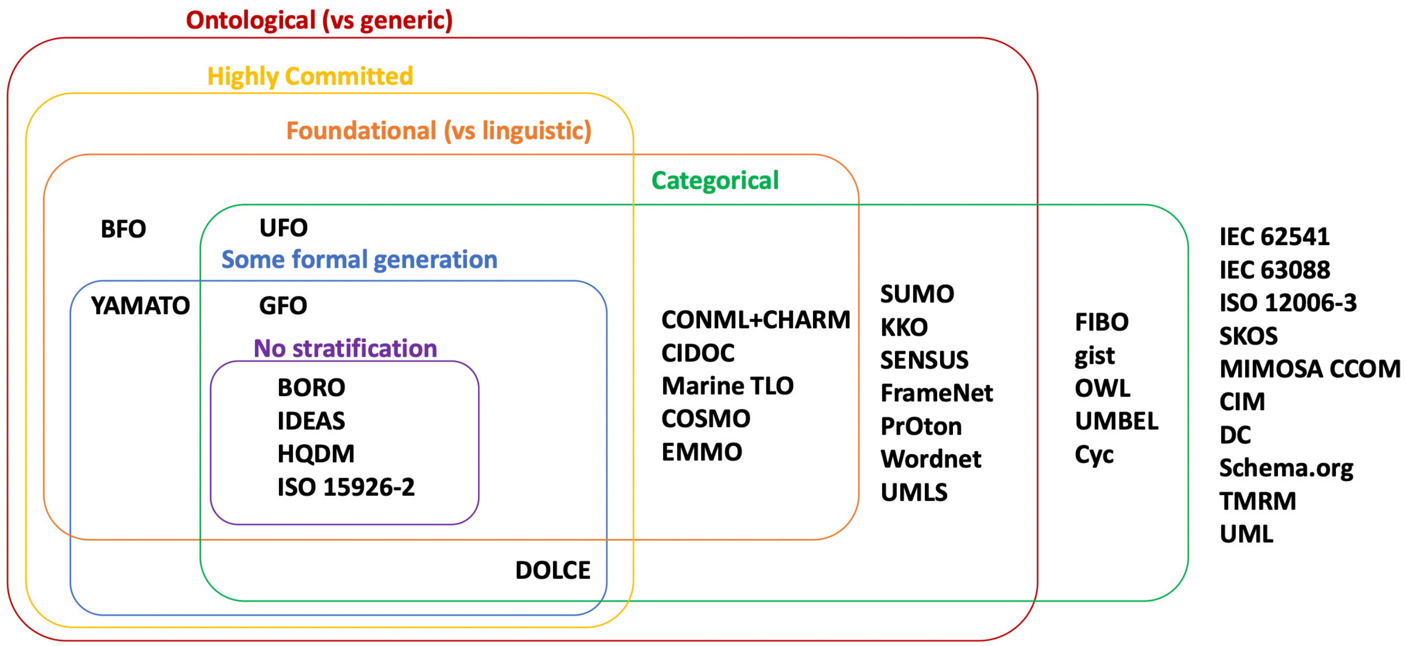

Figure 3: A Venn diagram showing classifications of Top-Level Ontologies.

Approach

The Foundation Data Model seed is to be developed from the four 4-dimensionalist Top-Level Ontologies: BORO, IDEAS, HQDM and ISO 15926-2 that met all the technical requirements identified as relevant to the Foundation Data Model.

Further, the approach is to increase the rigour of the Top-Level Ontology by developing a formal constructional core ontology to maximise the entities formally generated relative to the entities taken as given.

Appendix A Candidate source top-level ontologies – longlist

A list (in alphabetic order) of all the candidate source top-level ontologies.

Acronym	Initial release	Links
BFO	2002	http://basic-formal-ontology.org/, https://en.wikipedia.org/wiki/Basic_Formal_Ontology, https://en.wikipedia.org/wiki/Upper_ontology#Basic_Formal_Ontology_(BFO)
BORO	late 1980s	https://www.borosolutions.net/, https://en.wikipedia.org/wiki/BORO, https://en.wikipedia.org/wiki/Upper_ontology#BORO
CIDOC (ISO 21127:2014)	1999	http://www.cidoc-crm.org/, https://en.wikipedia.org/wiki/CIDOC_Conceptual_Reference_Model, https://en.wikipedia.org/wiki/Upper_ontology#CIDOC_Conceptual_Reference_Model
CIM	1999	https://www.dmtf.org/standards/cim, https://en.wikipedia.org/wiki/Common_Information_Model_(computing)
ConML+CHARM	2011	http://www.conml.org/, http://www.conml.org/Resources/TechSpec.aspx, http://www.charminfo.org/
COSMO	not known – pre-2006	http://www.micra.com/, https://en.wikipedia.org/wiki/Upper_ontology#COSMO
Cyc	1984	https://www.cyc.com/the-cyc-platform, https://en.wikipedia.org/wiki/Cyc, https://en.wikipedia.org/wiki/Upper_ontology#Cyc
DC	1995	http://dublincore.org/, https://en.wikipedia.org/wiki/Dublin_Core

Acronym	Initial release	Links
DOLCE	2019	http://www.loa.istc.cnr.it/dolce/overview.html, https://en.wikipedia.org/wiki/Upper_ontology#DOLCE
EMMO	2019 (?)	https://github.com/emmo-repo/EMMO, https://materialsmodelling.com/2019/06/14/european-materials-modelling-ontology-emmo-release/
FIBO	2010 (?)	https://spec.edmcouncil.org/fibo/
FrameNet	2000 (?)	https://framenet.icsi.berkeley.edu/fndrupal/, https://en.wikipedia.org/wiki/FrameNet
GFO	2006	https://www.onto-med.de/ontologies/gfo, https://en.wikipedia.org/wiki/General_formal_ontology, https://en.wikipedia.org/wiki/Upper_ontology#General_Formal_Ontology_(GFO)
gist	2007	https://www.semanticarts.com/gist/, https://en.wikipedia.org/wiki/Upper_ontology#gist
HQDM	2011	http://www.informationjunction.co.uk/hqdm_framework/
IDEAS	2006	https://en.wikipedia.org/wiki/IDEAS_Group, https://en.wikipedia.org/wiki/Upper_ontology#IDEAS
IEC 62541	2006	https://opcfoundation.org/developer-tools/specifications-unified-architecture, https://en.wikipedia.org/wiki/OPC_Unified_Architecture
IEC 63088	2017	https://webstore.iec.ch/publication/30082
ISO 12006-3	2007	https://www.iso.org/standard/38706.html, https://en.wikipedia.org/wiki/ISO_12006
ISO 15926-2	2003	https://www.iso.org/standard/29557.html, https://en.wikipedia.org/wiki/ISO_15926, https://en.wikipedia.org/wiki/Upper_ontology#ISO_15926
KKO	not known	https://kbpedia.org/docs/kko-upper-structure/
KR Ontology	1999	http://www.jfsowa.com/ontology/toplevel.htm
MarineTLO	2013 (?)	https://projects.ics.forth.gr/isl/MarineTLO/, https://en.wikipedia.org/wiki/Upper_ontology#MarineTLO
MIMOSA CCOM	not known	https://www.mimosa.org/mimosa-ccom/, https://en.wikipedia.org/wiki/OpenO%26M
OWL	2004	https://www.w3.org/OWL/, https://en.wikipedia.org/wiki/Web_Ontology_Language
PROTON	2005 (?)	https://ontotext.com/documents/proton/Proton-Ver3.0B.pdf, https://en.wikipedia.org/wiki/Upper_ontology#PROTON
Schema.org	2011	https://schema.org/, https://en.wikipedia.org/wiki/Schema.org
SENSUS	2001	https://www.isi.edu/natural-language/projects/ONTOLOGIES.html
SKOS	2009	https://www.w3.org/2004/02/skos/, https://en.wikipedia.org/wiki/Simple_Knowledge_Organization_System

Acronym	Initial release	Links
SUMO	2000	http://www.adampease.org/OP/, https://en.wikipedia.org/wiki/Suggested_Upper_Merged_Ontology, https://en.wikipedia.org/wiki/Upper_ontology#SUMO_(Suggested_Upper_Merged_Ontology)
TMRM	late 1990s	https://www.isotopicmaps.org/tmrm/, https://en.wikipedia.org/wiki/Topic_map
UFO	2005	https://nemo.inf.ufes.br/en/projetos/ufo/, https://en.wikipedia.org/wiki/OntoUML, https://en.wikipedia.org/wiki/Upper_ontology#UFO_(Unified_Foundational_Ontology)
UMBEL	2008	https://en.wikipedia.org/wiki/UMBEL, https://en.wikipedia.org/wiki/Upper_ontology#UMBEL
UML	1994	http://uml.org/, https://en.wikipedia.org/wiki/Unified_Modeling_Language
UMLS	1986	https://www.nlm.nih.gov/research/umls/index.html, https://en.wikipedia.org/wiki/Unified_Medical_Language_System
WordNet	1985	https://wordnet.princeton.edu/, https://en.wikipedia.org/wiki/WordNet, https://en.wikipedia.org/wiki/Upper_ontology#WordNet
YAMATO	1999	https://en.wikipedia.org/wiki/Upper_ontology#YAMATO_(Yet_Another_More_Advanced_Top_Ontology)

(?) is used to indicate the data is uncertain.

Acknowledgements

Lead Author:
Matthew West

Contributors:
Al Cook
David Leal
Andrew Mitchcell
Chris Partridge
Jan Sullivan

This research forms part of the Centre for Digital Built Britain's (CDBB) work at the University of Cambridge within the Construction Innovation Hub (the Hub). The Hub is funded by UK Research and Innovation through the Industrial Strategy Fund.

www.cdbb.cam.ac.uk