

Current State of ontology in engineering systems

Henson Graves and Matthew West

This report is an expanded version of part of the Ontology Summit 2012 communicate as it relates to the current state of ontology use in engineering big systems. The Ontology Summit is an annual event of ONTOLOG, a virtual community of practice devoted to advancing the field of ontology, ontological engineering and semantic technology. The Summit topic for 2012 was ontology as it relates to big systems. Other topics were Big Data, Federation and Integration of Systems, and Ontology Quality. Engineering big systems includes both big systems engineered by an enterprise and the big enterprises that produce and use big systems. This paper gives an overview of the current state of ontology use in engineering systems.

Models and ontology in engineering

Engineers have always built models; models to describe physical systems, to specify products to be built, and to describe system interaction with the physical world. In common usage, a model is anything used to represent anything else. Scientific models are used to represent physical things and factual relationships. Engineering models represent a physical system or manufactured product. A system can be represented by many different models that may have different purposes. Engineering is becoming increasingly model centric in that models serve as the ground truth for design and analysis in that they are the authoritative source of information. The engineering community recognizes that modeling languages need a precise meaning to enable collaboration, standards, and reasoning. This is where ontology comes in. Ontological analysis is an enabler for good modeling. The analysis enables us to identify the good conceptualizations needed for modeling a domain and to establish the meaning of these conceptualizations.

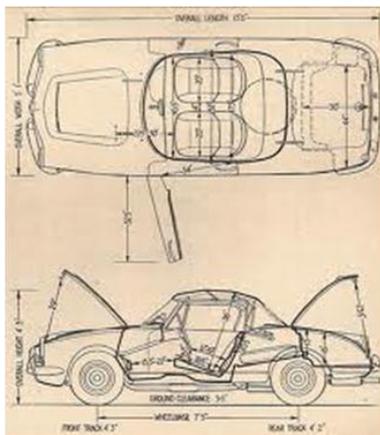


Figure 1 A design model for an automobile

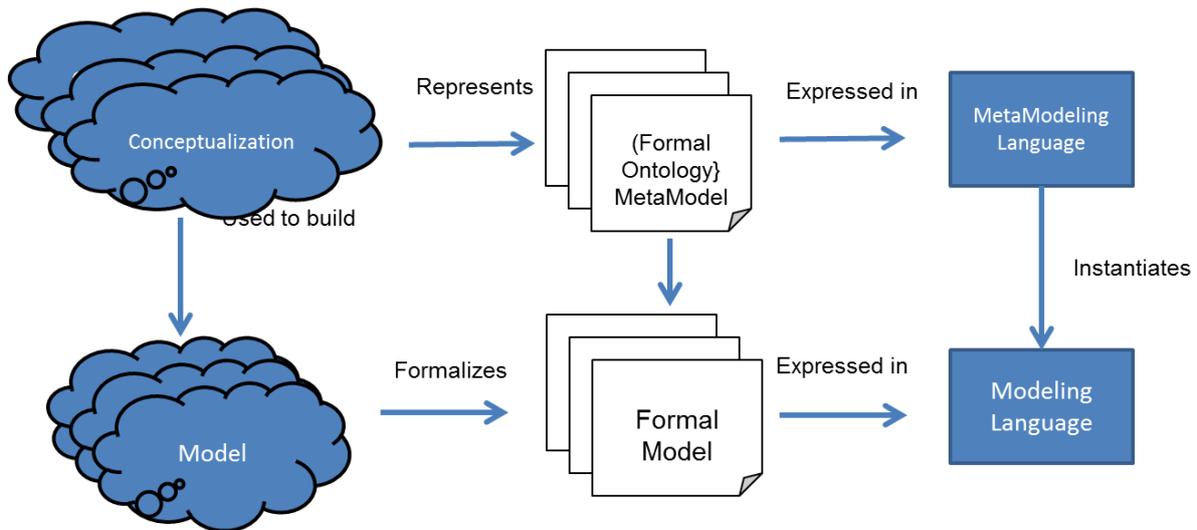
Ontology as a discipline dates back to Aristotle. For Aristotle, the main task of philosophy was to experience the empirical world and acquire knowledge about it (Metaphysics, Chapter 9). He created an ontology of substances. For Aristotle the general properties of things have to be found through cognitive processes. The modern usage of ontology is credited to the philosopher Edmund Husserl. Ontology as a philosophical discipline aims at developing a system of general categories, the relationships between them and the rules that govern them which together form a theory of reality.

Relationship between Models, Ontologies, and Modeling Languages

How does ontology relate to the practice of building models in an engineering modeling language such as SysML or ISO15926? Models are used to describe systems that exist or might exist in the real world. To describe those systems one needs a language to talk about the world. An example of what

a language might include is the notion of a system as an individual that consists of components, where the components are connected in some way such that the system as a whole has emergent behavior.

When a Model is *expressed* in a modeling language the first issue is that the model be a syntactically correct model of the language. A language description for a modeling language such a SysML includes the syntactical rules which define syntactically correct sentences. A language specification determines all possible *grammatically valid* models that can be constructed using that language. In addition if one can describe the pattern for a *system* then one can use software to check that one has a system model, not just an arbitrary model.



In OMG terminology, a template for *System* would be called a metamodel.

Ontology Support for Systems Engineering

Two areas where ontology is being used and has much more potential for use are System Modeling and Enterprise Modeling In particular; ontologies have been used to model system specifications and enterprise architectures. An ontology of systems is a pattern for what constitutes a system: the parts and connections, identity, dependence, unity And, for engineered systems, replaceable system components. Key relations like classification, specialization, and whole-part are well understood in the realm of ontology, and see major usage in systems engineering.



A key element of *systems* that is generally not well supported in ontology is system components. The example of the design for an automobile provides an example of the kind of ontological analysis that is needed for system modeling. The design for a manufacturer's automobile model identifies each component such as the left headlight as well as the specification for the headlights. Each automobile built will have a specific left headlight component.

Figure 3 Automobile With Missing Headlight

These system components do not behave in the same way as ordinary physical objects. Nicola Guarino presented material from an as yet unpublished paper on system components with some things one might say about a headlamp:

1. The left headlamp of my car is not working
2. The left headlamp of my car is missing
3. This cable connects the battery to the left headlamp
4. The left headlamp of my car has been replaced 2 times

If a headlight is smashed does the automobile still have a headlight component? When the left headlight of an automobile is removed, a technician still says that an electrical cable connects to the left headlight. What do engineers actually speak of when they talk about connections to the missing left headlight and what do they have in mind when they speak this way? Do they have in mind objects whose existence conditions are determined by specifications? This example generated a lot of discussion regarding the identity criteria for an implementation of the automobile design. Can components be replaced within a unit without the identity of the unit changing?

During the Summit, Matthew West illustrated the issue by telling the (simplified) life story of a system component: the pump, P101, at the bottom of a distillation column, an example which was used in the development of ISO 15926, a standard for the integration of data for process plant.

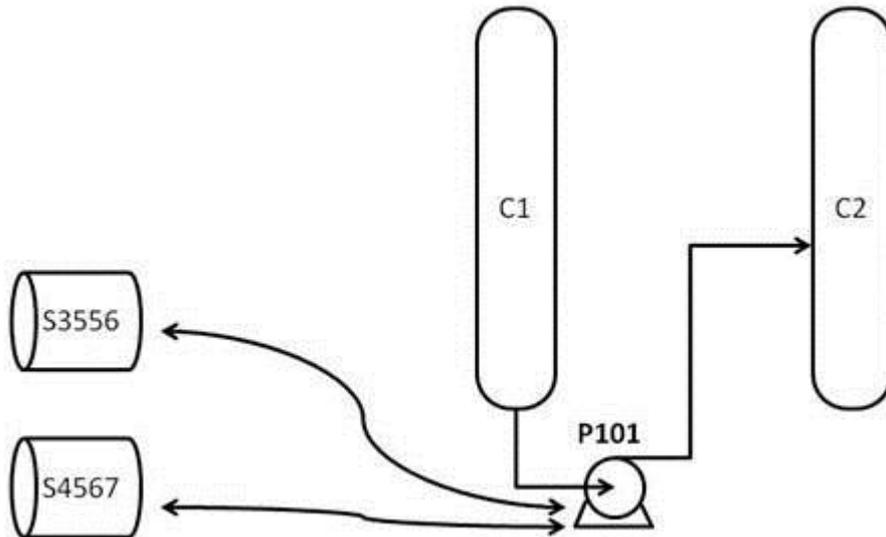


Figure of Distillation Unit and replacement of system component.

The designer creates a drawing of the distillation column including at the bottom of the column a pump to pump away the column bottoms. He labels it P101, decides that one pump will be sufficient, and gives the specification for the pump in terms of Net Positive Suction Head, differential head, flow rate, materials of construction, and many other things.

The construction engineer picks up the drawing and specification and notices he has to install a pump as P101. Fortunately, he has a pump in stock from a previous project, that has been in stores unused for 5 years which exactly meets the specification. On it is stamped Serial No S3556.

The designer and the Operator comes to see the pump be installed, and once the connections are made, he gives the pump a friendly kick and says to the construction engineer "It's good to see P101 realized at last". The construction engineer says in return "Yes, and it's good to get S3556 off my hands at last." He turns to the operator and says "Why don't we change your drawings to show S3556 instead of P101?" The operator says "No, don't do that, it's a replaceable part, and one day another pump will be put there, and I don't want to have to change all the drawings and other documentation that refers to P101 each time it is replaced, as far as I am concerned it's the same pump whatever is installed there."

Sometime later the pump breaks down and needs to be taken back to the workshop. The maintenance engineer says to the operator "Hi, can I take S3556 installed as P101 back to the workshop?" The operator replies "Sure, but what am I supposed to do without my P101? If it does not exist I cannot operate my distillation column." The maintenance engineer responds, "I understand. We have another pump S4567, that meets the same specification as P101. We'll replace S3556 with it and you will only be without P101 for a few hours. I don't understand how you can continue to call it P101 though when all the parts have changed at once." The operator replies "I don't care about that. What I care about is what is connected in my system to pump the liquid from the bottom of the column. As long as it does that, it is P101 to me."

Later the distillation column is demolished. The operator says, "A sad end, I was very fond of P101, but it is no more." The demolition engineer says, "Yes indeed. Fortunately, we can take S4567 and use it on another plant."

It's probably worth summarizing the key characteristics of a system component:

- It comes into existence the first time it is installed.
- It is identical to the equipment items installed, whilst they are installed (but not before or after).
- It can survive complete replacement of all its parts at once.
- It can survive periods of non-existence.
- It ceases to exist when the system it is a component of ceases to exist.

This is clearly rather different from the life of ordinary physical objects. However, relatively few ontologies recognize that such things exist. Many try to fob system components off as being classes, or abstract individuals, though these clearly do not have the required characteristics.

Systems Engineering Needs From Ontology

There are many engineering and enterprise tasks where ontology is definitely applicable, but is not yet in wide use. Systems engineering is all about assembly from components and understanding the whole and the relationships between the parts and support for the use of the same parts in different

systems. This calls for ontologies which can themselves be components of other ontologies and be assembled to contribute to an ontology of the whole. Yet in general ontology developments are one-off with it being rare for ontologies to be reused or be reusable. For ontology to be useful for the engineering of big systems, reusable ontologies to support reusable engineering models will be important.

Big systems have a long life and usually change over that life. This means that the ontologies that describe a system need to be able to change, but in a way that means that the history is not lost. This requires a sophisticated approach to change management in model and ontology creation and maintenance. Big systems tend to interact with their environment and change state as a result of interaction. This means those conceptualizations are needed to model state change and system evolution throughout its lifecycle.

For some time information models have been used to model enterprise information. Foundation ontologies contain conceptualizations needed for enterprise modeling. These include processes, events, descriptions, plans, physical quantities, individuals, types etc. Further ontologies provide relationships between the concepts which can be exploited to relate data needed to determine program status. Some enterprises have recognized that ontologies generalize these information models and provide better access and organization than traditional data models.

Increasingly the importance of social, legal, and value-related qualities of big systems, both enterprises that make products and their big system products, are being realized. These qualities must be taken into account during the design and evaluation of systems. Again this means that conceptualizations are needed to model these system qualities. Other examples of potential ontology use are principles of how to construct good quality reusable models, the management of ontologies of and for large systems and the challenges in developing and maintaining them.

Impediments to ontology use

Different not necessarily compatible conceptualizations used in modeling big systems by different stakeholders have to be integrated to achieve the semantic interoperability for stakeholders to collaborate. Many of the challenges in the application of ontology to big systems are that (1) the modelling of federated systems requires a broader collection of conceptualizations, (2) the ability to take data from one lifecycle stage and repurposed it for use in later lifecycle stages, (3) and integrating the results of models using multiple modeling languages.

One of the biggest impediments in modeling a domain is the use of an incorrect conceptualizations or ontologies. One common case in engineering is process models which have no relationship to any engineering process that could ever possibly work. When modeling languages make ontological commitment prematurely before the conceptualizations of the domain have been understood, they run the risk of failure to be adopted. As a result the use of ontologies in engineering demands the development of ontology validation methods, possibly using techniques used to validate models in engineering.

Ontology Tools and Training for Systems Engineers

The discipline of ontology has work and methodology for ontology engineering which is methodology for constructing ontologies. This may prove of use for engineering. One way to show

engineers the importance of ontology is to show examples of how poor ontologies (or the lack of them) impede the work to make and use a system. Then they are better prepared to develop the kind of ontologies needed.