System Modeling Concepts

System Modeling Concepts

The printable version is no longer supported and may have rendering errors. Please update your browser bookmarks and please use the default browser print function instead.

Lead Author: *Sanford Friedenthal,* ***Contributing Authors:*** *Dov Dori, Yaniv Mordecai*

A system model represents aspects of a system and its environment. There are many different types of models, as there are a variety of purposes for which they are built. It is useful to have a common way to talk about the concepts underlying the many different types of models (e.g., many modeling techniques enable the understanding of system behavior, while others enable the understanding of system structure). This article highlights several concepts used for modeling systems.

□

Contents

Abstraction

Perhaps the most fundamental concept in systems modeling is abstraction, which concerns hiding unimportant details in order to focus on essential characteristics. Systems that are worth modeling have too many details for all of them to reasonably be modeled. Apart from the sheer size and structural complexity that a system may possess, a system may be behaviorally complex as well, with emergent properties, non-deterministic behavior, and other difficult-to-characterize properties. Consequently, models must focus on a few vital characteristics in order to be computationally and intellectually tractable. Modeling techniques address this complexity through various forms of abstraction. For example, a model may assume that structural characteristics of many individual components of a particular type are all the same, ignoring the small order differences between individuals in instances that occur in real life. In that case, those differences are assumed to be unimportant to modeling the structural integrity of those components. Of course, if that assumption is wrong, then the model could lead to false confidence in that structural integrity. There are two key concepts that are applied in regard to modeling different levels of abstraction, which are: view and viewpoint, and black-box and white-box modeling, which are described below. Although these two modeling methods are the most widely recognized, different modeling languages and tools employ other techniques as well.

View and Viewpoint

IEEE 1471, a standard for architecture modeling, defines "view" and "viewpoint" as follows:

- View - A representation of a whole system from the perspective of a related set of concerns.
- Viewpoint - A specification of the conventions necessary for constructing and using a view; a pattern or template from which to develop individual views by establishing the purposes and audience for a view and the techniques for its creation and analysis.

Even though IEEE 1471 is focused on architecture models, the concepts of view and viewpoint are general and could apply to models for other purposes as well (IEEE 2000). The viewpoint addresses the concerns of the stakeholders and provides the necessary conventions for constructing a view to address those concerns; therefore, the view represents aspects of the system that address the concerns of the stakeholder. Models can be created to represent the different views of the system. A systems model should be able to represent multiple views of the system to address a range of stakeholder concerns. Standard views may include requirements, functional, structural, and parametric views, as well as a multitude of discipline-specific views to address system reliability, safety, security, and other quality characteristics.

Black-Box and White-Box Models

A very common abstraction technique is to model the system as a black-box, which only exposes the features of the system that are visible from an external observer and hides the internal details of the design. This includes externally visible behavior and other physical characteristics, such as the system's mass or weight. A white-box model of a system, on the other hand, shows the internal structure and displays the behavior of the system. Black-box and white-box modeling can be applied to the next level of design decomposition in order to create a black-box and white-box model of each system component.

Conceptual Model

A conceptual model is the set of concepts within a system and the relationships among those concepts (e.g., view and viewpoint). A system conceptual model describes, using one diagram type (such as in Object-Process Methodology (OPM)) or several diagram types (such as in Systems Modeling Language (SysML)), the various aspects of the system. The conceptual model might include its requirements, behavior, structure, and properties. In addition, a system conceptual model is accompanied by a set of definitions for each concept. Sometimes, system concept models are defined using an entity relationship diagram, an object-process diagram (OPD), or a Unified Modeling Language (UML) class diagram.

A preliminary conceptual (or concept) model for systems engineering (Systems Engineering Concept Model) was developed in support of the integration efforts directed toward the development of the Object Management Group (OMG) SysML and the International Organization for Standardization (ISO) AP233 *Data Exchange Standard for Systems Engineering* (ISO 2010). The concept model was originally captured in an informal manner; however, the model and associated concepts were rigorously reviewed by a broad representation of the systems engineering community, including members from the International Council on Systems Engineering

(INCOSE), AP233, and SysML development teams.

A fragment from the top level systems engineering concept model is included in Figure 1. This model provides concepts for requirements, behavior, structure and properties of the system, as well as other concepts common to systems engineering and project management, such as the stakeholder. The concept model is augmented by a well-defined glossary of terms called the semantic dictionary. The concept model and the semantic dictionary contributed greatly to the requirements for the OMG Systems Modeling Language written in the UML for Systems Engineering Request for Proposal.

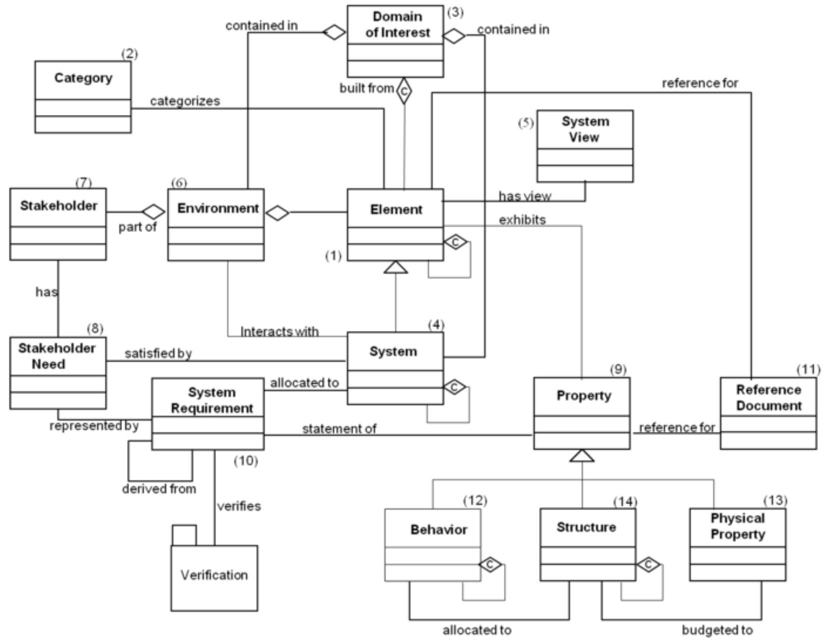

Figure 1. Fragment of the Object Management Group System Concept Model (Oliver 2003, Slide 3). Permission granted by David Oliver on behalf of INCOSE MDSD Working Group. All other rights are reserved by the copyright owner.

A concept model is sometimes referred to as a meta-model, domain meta-model, or schema, and can be used to specify the abstract syntax of a modeling language (refer to the Model Driven Architecture (MDA®) Foundation Model (OMG 2010)). Several other systems engineering concept models have been developed but not standardized. Future standardization efforts should establish a standard systems engineering concept model. The model can then evolve over time as the systems engineering community continues to formalize and advance the practice of systems engineering.

References

Works Cited

IEEE. 2000. *Recommended Practice for Architectural Description for Software-Intensive Systems*. New York, NY, USA: Institute of Electrical and Electronics Engineers (IEEE), IEEE 1471-2000.

ISO. 2010. *OMG System Modeling Language (OMG SysML),* version 1.2. Needham, MA, USA: Object Management Group.

OMG. 2010. *MDA Foundation Model*. Needham, MA, USA: Object Management Group. Document number ORMSC/2010-09-06.

Primary References

ANSI/IEEE. 2000. *Recommended Practice for Architectural Description for Software-Intensive Systems*. New York, NY, USA: American National Standards Institute (ANSI)/Institute of Electrical and Electronics Engineers (IEEE), ANSI/IEEE 1471-2000.

Dori, D. 2002. *Object-Process Methodology – A Holistic Systems Paradigm*. New York, NY, USA: Springer-Verlag.

Guizzardi, G. 2007. "On ontology, ontologies, conceptualizations, modeling languages, and (meta)models," Proceedings of the 2007 Conference on Databases and Information Systems IV. Available at: http://portal.acm.org/citation.cfm?id=1565425.

IEEE. 2000. *Recommended Practice for Architectural Description for Software-Intensive Systems*. New York, NY, USA: Institute of Electrical and Electronics Engineers (IEEE), IEEE 1471-2000.

INCOSE. 2003. *Systems Engineering Concept Model. Draft 12 Baseline.* Seattle, WA, USA: International Council on Systems Engineering. Available at: http://syseng.omg.org/SE_Conceptual%20Model/SE_Conceptual_Model.htm.

OMG. 2003. *UML for Systems Engineering Request for Proposal*. Needham, MA, USA: Object Management Group. OMG document number ad/2003-3-41. Available at: http://www.omg.org/cgi-bin/doc?ad/2003-3-41.

Additional References

None.

< Previous Article | Parent Article | Next Article >

SEBoK v. 2.10, released 06 May 2024