Panel:
Extending UML from Software to Systems Engineering

Moderator: Stephanie White, Long Island University
Panelists:
Murray Cantor, Rational Software
Sanford Friedenthal, Lockheed Martin Corporation
Cris Kobryn, Telelogic
Byron Purves, Boeing

Abstract

"The Unified Modeling Language (UML) is a graphical language for visualizing, specifying, constructing, and documenting the artifacts of a software-intensive system" [1]. This panel of domain and tool development experts discusses the features needed to extend UML to support systems engineering, and together with the audience, investigates the advisability and feasibility of the extensions.

Audience participation is an important part of this session. The audience is invited and urged to contribute. While probably in agreement that consistent modeling methods between the two domains will lead to more effective and efficient system development, members of the audience may be concerned that an extended UML may not meet their needs and may be too complex. Since the audience has had diverse experience is engineering systems and using system-engineering methods, audience contributions can lead to an improved UML standard.

1. Position statements

Sanford Friedenthal
Lockheed Martin Corporation
Chair, OMG Systems Engineering Domain Special Interest Group

The primary objective for systems engineering is to develop cost effective system solutions, which satisfy customer and stakeholder needs. Over the past few years, several process standards have been introduced to help accomplish this objective, including ANSI/EIA 632, IEEE 1220, ISO 15288, and the CMMI. However, a critical missing element has been a standardized modeling language to specify, design, and verify complex systems. A standard modeling language for systems engineering, is intended to bridge the semantic gap between systems, software, and other engineering disciplines, and improve the ability to exchange systems engineering information amongst tools.

The Unified Modeling Language has become a de-facto standard amongst the software community for analyzing and designing software. The language is believed to be sufficiently robust to support extensions to address the needs of systems engineering. However, there are significant challenges for UML to provide a complete, unambiguous, and intuitive representation for modeling systems. This includes representing hierarchical behavior, continuous time systems, inputs and outputs, physical structure, performance, physical, and other system characteristics, such as reliability and safety, and parametric relationships.

In July 2001, the International Council on Systems Engineering (INCOSE) and the OMG initiated a collaboration to address this need. The Systems Engineering Domain Special Interest Group (SE DSIG) was formed, and held its kickoff meeting in September 2001, with the goal of extending UML for systems engineering. This effort is closely aligned with the ISO AP-233 effort to develop a data interchange standard for systems engineering, which is focused on tool interoperability. The SE DSIG is establishing a set of requirements for UML for systems engineering, which they expect OMG to issue in 2003 as part of an OMG RFP for a "UML Profile for Systems Engineering". In addition, the SE DSIG has been working closely with some of the UML 2.0 Submission teams to ensure these requirements are being addressed in the proposed update to UML V2.0.

Cris Kobryn
Telelogic
Co-Chair, OMG Analysis & Design Task Force

In the more than five years since the Object Management Group (OMG) adopted the Unified Modeling Language (UML) in 1997, it has established itself as the de facto standard for modeling software systems. During this time UML has been successfully

Proceedings of the 10 th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems (ECBS'03)
0-7695-1917-2/03 $17.00 © 2003 IEEE

applied to specify software solutions in a wide variety of domains, ranging from healthcare and financial services to telecom and aerospace.

Recently there has been keen interest among systems engineers to customize UML for their specialized domain. They recognize the advantages of a common modeling language, and realize that there is more commonality than difference between software and systems engineering. Consequently, UML experts at the OMG have been collaborating with systems engineering experts in the OMG Systems Engineering Domain Special Interest Group (OMG SE DSIG) and the International Council on Systems Engineering (INCOSE) to customize UML to meet the needs of systems engineers.

Early collaborations found that UML 1.x was inadequate to meet the needs of systems engineering because of its limitations regarding scalability and specifying crosscutting functionality (i.e., functionality that applies across structures and behaviors).

As a result, current collaborations are focused on adapting the evolving revision of UML, UML 2.0, which includes an enhanced Profiles mechanism for customization. Other features proposed for UML 2.0 that will interest systems engineers include the following:

- Architectural specifications: UML 2.0 will improve support for specifying large, complex architectures by supporting the hierarchical decomposition of structure and behavior. This includes the specification of structured classifiers (classes, components), subsystems, sequence diagram references, and activity diagram partitions.
- Crosscutting functionality: UML 2.0 will increase the integration of structure with behavior, allowing modelers to define functions that apply across structural and behavioral boundaries.
- Unification of actions and activities: UML 2.0 will reuse the same basic constructs for defining actions and activities, which will significantly increase the precision and efficiency for specifying process workflows and other types of behavior.
- Multiple views: UML 2.0 will allow different system stakeholders to view the system from different, complementary perspectives.

Although the improvements described above will make UML 2.0 significantly more useful to systems engineers "out of the box," much additional work will be required to customize the language so that it meets the special needs of systems engineers (e.g., defining physical systems and continuous time properties). Fortunately, the OMG SE DSIG has already started defining the requirements for defining a UML 2.0 Profile for Systems Engineering, and plans to issue a Request for Proposals in early 2003. We anticipate that this evolving profile will play an important role in addressing the modeling needs of the systems engineering community.

Murray Cantor
Rational Software
Technical Liaison, OMG Systems Engineering Domain Special Interest Group

Over the last four years, Rational has experienced increasing demand for applying UML semantics and associated architecture processes to full systems (hardware, software, workers …) development. This presentation is based on what we have learned from a series of ongoing engagements at various sites of Lockheed-Martin, Boeing, Raytheon, and others.

With the demand for more integration and the continuing evolution of technology, both the problem space and solutions spaces facing the system architect are growing. New, UML-based modeling techniques have proven useful in addressing the resulting complexity, enabling cross-functional communications in the development teams.

I discuss some of the UML semantic elements' strengths and weaknesses for modeling systems. Further, I introduce and motivate the Rational Unified Process (RUP) SE system model framework for modeling systems specifications, architecture and design.

2. Panelist biographies

Dr. Murray Cantor is a Principal Consultant in the Strategic Services Organization of Rational Software. His areas of expertise include software and system engineering processes, and system development management and leadership. Cantor leads the development of RUP SE, the extension of the Rational Unified Process to system engineering. In addition, he serves as Rational's technical liaison to Object Management Group's System Engineering Domain Special Interest Group.

Cantor learned the trade of system development by taking leadership roles in a variety of projects at IBM and TASC. Cantor has been a system architect, team lead, project manager, development product manager, architecture manager, and program manager. He is the author of the books *Object-Oriented Project Management with UML*, published by John Wiley in 1998, and *Software Leadership*, published by Addison-Wesley in October 2001. You can email him at mcantor@rational.com.

Proceedings of the 10 th IEEE International Conference and Workshop on the Engineering of Computer-Based Systems (ECBS'03)
0-7695-1917-2/03 $17.00 © 2003 IEEE

Sanford Friedenthal's experience includes the full system life cycle from conceptual design, through development and production on a broad range of systems including missile systems, electro-optical navigation and targeting systems, and information systems. Mr. Friedenthal has been a manager for systems engineering at Lockheed Martin responsible for ensuring systems engineering processes are implemented on the programs, and enhancing overall systems engineering capability. He has been a lead developer of advanced systems engineering processes and methods including the Lockheed Martin Integrated Engineering Process, the Software Productivity Consortium's Integrated Systems and Software Engineering Process, and the Object-Oriented Systems Engineering Method (OOSEM). Mr. Friedenthal is the liaison between INCOSE and OMG, and chairs the OMG Systems Engineering Domain Special Interest Group (SE DSIG) to support development of a UML profile for System Engineering. You can email him at sanford.friedenthal@lmco.com.

Cris Kobryn is the Chief Technologist at Telelogic, where he is responsible for standards leadership, strategic planning and technology evangelism. Cris has applied advanced technologies to solve a wide range of business and scientific problems, and is an expert in distributed software architectures, component-based development and software modeling. As an Object Management Group representative, he has been a major contributor to the Unified Modeling Language (UML) specification and serves as the co-chair of both the UML Revision Task Force and the Analysis & Design Task Force. Cris is a member of ACM, IEEE, INCOSE and AAAI. You can email him at Cris.Kobryn@telelogic.com.

Byron Purves is a Technical Fellow in System Engineering with The Boeing Company in Huntsville, Alabama. He holds a B.Sc. in Physics and Mathematics and a Ph.D. in Physics. Over a thirty-year career he has published papers in automatic speech recognition, aircraft noise control, electric utility control systems, space automation and robotics and system verification. He holds five instrumentation patents. His experience includes time spent in both software engineering and system engineering management. His present interests include integration of system and software engineering. You can email him at byron.purves@boeing.com.

Stephanie Farbman White is a full professor in the College of Information and Computer Science on the C.W. Post Campus of Long Island University. She is also president of System World, which develops systems-engineering technologies. Her research interests include systems engineering for software-intensive systems, requirements engineering, process modeling, and model-based analysis and design. Prior to teaching at Long Island University, Dr. White was a manager and principal engineer at Northrop Grumman, where she worked on Air and Space programs. She received a PhD in computer science from Polytechnic University in Brooklyn, and is a member of ACM, IEEE, and INCOSE. You can email her at Stephanie.White@liu.edu.

3. References

[1] G. Booch, J. Rumbaugh, and I Jacobson, *The Unified Modeling Language User Guide*, Addison Wesley, MA, 1999.