Object-Process Methodology:
A Graphic-Textual Requirements Engineering Platform

Proposal for Tutorial
Requirements Engineering Conference (RE'03)
8th-12th September 2003, Monterey Bay, CA

Dov Dori

Technion, Israel Institute of Technology
and
Massachusetts Institute of Technology
dori@mit.edu

Background and Testimony

Object-Process Methodology is a unifying approach for developing, communicating, supporting and evolving systems of various domains, types, magnitudes and complexities. To support these activities, a clear and concise single-model formalism has been developed, which is expressed in two complementary modalities: formal, intuitive graphics and natural language text. Taught at MIT and the Technion, OPM has evolved as a significant extension of Object technology which caters equally well to systems' structure (through objects and relations) and behavior (through processes that transform OPM encompasses the entire lifecycle of a software system or product, from concept and initiation through development to deployment. Currently, the most prominent strength of OPM is its ability to support requirements engineering.

Mr. Mark Richer, Systems & Applications Architect at Pratt & Whitney Canada (mark.richer@pwc.ca) recently wrote the following about OPM:
"OPM is a methodology designed to facilitate the formal and rigorous description and analysis of systems, be they information systems or physical systems. The striking feature about OPM is that it recognises as peers the notion of Process and Object. This enables OPM to provide true unified modelling of the structural as well as dynamic aspects of a system. This is in contrast to traditional Object Oriented analysis, such as UML, which use a variety of different modelling formats and symbols to accomplish the same end. For this reason, OPM is being used at P&WC to perform requirements elicitation, systems analysis and the development of precise domain specific ontologies.

Another striking feature of OPM is that it combines a graphics based model, termed Object Process Diagram or OPD, with structured formal declarative statements that complement each interaction depicted on the diagram. These declarative statements are termed Object Process Language or OPL paragraphs. The added value of having both views, graphic and semantic, is that it facilitates non-specialist understanding of the resultant analysis. A short-coming of OPM at the present time is its lack of support for mechanical theorem proving, which would provide consistency checking of the system being developed. This becomes critical as the size of the project increases. At P&WC, we have used OPM to develop a axiomized ontology pertaining

to P&WC's unique set of Product Configuration Management concepts, which resulted in the generation of thousands of statements in OPL. We would have dearly enjoyed the benefits of mechanical consistency checking on this project.

The above mentioned Configuration Management project has now been expanded to include the participation of consultants from IBM Canada, Enovia Montreal and Enovia Labs in Charlotte N.C. IBM Canada and Enovia have agreed to use OPM as the means to formally and precisely model the concepts embodied by the Enovia LCA (Life Cycle Applications) product as part of an effort to establish the fits and gaps between this product and P&WC's needs.

OPM is also being used for requirements elicitation and systems analysis on a major P&WC/Engineering project to implement an Earned Value Management System (EVMS), which will bring the Engineering Change Request process together with Project Systems capabilities of SAP. This particular project is a full blown J2EE implementation."

In the preface of this course textbook, *Object-Process Methodology – A Holistic Systems Paradigm* (Dori, D., Springer Verlag, 2002), Prof. Edward Crawley, Head of MIT's Aero-Astro Department, who teaches OPM in his Systems Architecture course, wrote:

"Mature disciplines, such as mechanics, are well into the era of symbolic manipulation and prediction. Maturing disciplines, such as human genomics, are in the phase of symbolic representation. OPM is a parallel development in symbolic representation of systems. … I have used OPM in my System Architecture course at MIT. It has proved an invaluable tool to professional learners in developing models of complex technical systems, such as automobiles, spacecraft and software systems. It allows an explicit representation of the form/function duality, and provides an environment in which various architectural options can be examined. Incorporating OPM into my subject has added the degree of rigor of analysis necessary to move the study of technical system architecture towards that of an engineering discipline.

One can anticipate that there will be many academic applications of OPM. I would consider using it in intermediate or advanced subjects in system engineering, product development, engineering design and software engineering. It is ideal for courses that demonstrate how various disciplines come together to form a multi-disciplinary product."

Goals

The tutorial will present the underlying OPM ontology and its industrial applications as a with emphasis on Web and distributed applications as well as open reuse of software components. It is based in part on material I will teach at MIT Professional Institute in the course **Systems Development with UML and Object-Process Methodology** in July 2002 and will be given again in July 2003. See http://web.mit.edu/professional/summer/courses/computer/6.18s.html.

Potential Attendees and Background Knowledge

The target audience includes information technology professionals interested in modeling software systems. Specifically, database administrators and system

integrators, analysts, designers, modelers, executives, and project leaders will benefit from attending the tutorial and applying OPM for the purpose of developing better software systems faster and more reliably.
Background knowledge in object technology is helpful but not mandatory.

Why the Tutorial is Important

A coherent software system modeling methodology is essential for requirements engineering. OPM is such an approach. It is founded on well-defined ontology with solid infrastructure; it has clear formal semantics expressed via graphics and natural language, which enable fast and reliable system modeling; and it caters to domain experts who are not IT professionals and therefore enables them to actively engage in the RE process.

Detailed Outline

OPM ontology: Objects, processes, states, and how they relate. Object-Process Diagrams and Object-Process Language, transformations and state transitions, enablers, transformees, fundamental structural relations: aggregation, characterization, generalization and Inheritance, classification-instantiation, complexity management, abstraction and refinement mechanisms, system lifecycle.

OPM models for Web-based systems and code mobility over the Internet: client-server, code-on-demand, remote procedure invocation, software agents, "Inheritance on Steroids": open reuse of OPM components with applications to accelerated multi searching over distributed data and knowledge sources.

Reading

The main textbook for the course is: Dori, D. *Object-Process Methodology: A Holistic Systems Paradigm*. Springer Verlag, Berlin, Heidelberg, New York, 2002. The book includes a CD with three CASE tools that support OPM and generate natural language from the diagrams. These tools will serve for demonstration of OPM principles and applications during the tutorial.

About the Presenter

Dov Dori is Associate Professor at the Faculty of Industrial Engineering and Management, Technion, Israel Institute of Technology, and Research Affiliate at MIT. Between 1999-2001 he was Visiting Associate Professor at MIT's Engineering Systems Division and Visiting Scholar at Sloan School of Management.

Professor Dori received his B.Sc. in Industrial Engineering and Management from the Technion in 1975, M.Sc. in Operations Research from Tel Aviv University in 1981, and Ph.D. in Computer Science from Weizmann Institute of Science, Israel, in 1988. Between 1978 and 1984 he was Chief Industrial Engineer of MERKAVA Tank Production Plant, and between 1996 and 1998 he was Head of Technion's Area of Information Systems Engineering. His research interests include Systems

Development Methodologies, Information Systems Engineering, Computer-Aided Software Engineering and Document Analysis and Recognition. Dov Dori has developed the Machine Drawing Understanding System (MDUS) and Object-Process Methodology (OPM), for which he won several prizes.

Between 1999 - 2001 Prof. Dori was Associate Editor of IEEE Transaction on Pattern Analysis and Machine Intelligence (T-PAMI). He is Associate Editor of International Journal of Document Analysis and Recognition and is on the Editorial Board of the International Journal of Pattern Recognition and Artificial Intelligence. He is author/co-editor of four books and author of over 130 publications. He is Fellow of the International Association for Pattern Recognition (IAPR) and Senior Member of IEEE. He has been consultant and invited lecturer for companies, including Pratt and Whitney Canada, Ford Motor Company, FAA, NASA, The MITRE Corporation, Xerox, Kodak, and others.

Pertinent References

1. Dov Dori, Object-Process Methodology - A Holistic Systems Paradigm, Springer Verlag, Heidelberg, New York, 2002 (ISBN 3-540-65471-2; Hardcover). http://130.113.156.29/home/trade/booklists/e.acgi?A3-540-65471-2amazon.de/exec/obidos/ASIN/3540654712/booksnewasin/302-6670724-1913628
2. Dov Dori, Object-Process Analysis: Maintaining the Balance between System Structure and Behavior. *Journal of Logic and Computation*, 5, 2, pp. 227-249, 1995.
3. Dov Dori, Representing Pattern Recognition-Embedded Systems through Object-Process Diagrams: the Case of the Machine Drawing Understanding System. *Pattern Recognition Letters*, 16, 4, pp. 377-384, 1995.
4. Dov Dori, Vector-Based Arc Segmentation in the Machine Drawing Understanding Environment. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 17, 11, pp. 1057-1068, 1995. Abstract: http://www.computer.org/tpami/tp1995/i1057abs.htm?SMSESSION=NO
5. Dov Dori, Avigdor Gal and Opher Etzion, A Temporal Database with Data Dependencies: a Key to Computer Integrated Manufacturing. *International Journal of Computer Integrated Manufacturing*, 9, 2, pp. 89-104, 1996. Abstract: http://citeseer.nj.nec.com/8784.html
6. Dov Dori and Moshe Goodman, On Bridging the Analysis-Design and Structure-Behavior Grand Canyons with Object Paradigms. *Report on Object Analysis and Design*, 2, 5, pp. 25-35, 1996.
7. Dov Dori, Expressing Structural Relations among Dimension-set Components Using the Object-Process Methodology. *Report on Object Analysis and Design*, 2, 6, pp. 20-24, 1996.
8. Dov Dori, Object-Process Analysis of Computer Integrated Manufacturing Documentation and Inspection. *International Journal of Computer Integrated Manufacturing*, 9, 5, pp. 339-353, 1996.
9. Dov Dori, Analysis and Representation of the Image Understanding Environment Using the Object-Process Methodology. *Journal of Object*

Oriented Programming, 9, 4, pp. 30-38, 1996.

10. Dov Dori, Unifying System Structure and Behavior through Object-Process Analysis. *Journal of Object-Oriented Programming*, 9, 4, pp. 66-73, 1996.
11. Dov Dori and Moshe Goodman, From Object-Process Analysis to Object-Process Design. *Annals of Software Engineering*, 2, pp. 25-40.1996. Abstract: http://www.baltzer.nl/ansoft/ase2.html#ase008
12. Doron Myersdorf and Dov Dori, The R&D Universe and Its Feedback Cycles: an Object-Process Analysis. *R&D Management*, 27, 4, pp. 333-344, 1997.
13. Liu Wenyin and Dov Dori, A Protocol for Performance Evaluation of Line Detection Algorithms. *Machine Vision and Applications*, 9, pp. 240-250, 1997. http://link.springer.de/link/service/journals/00138/papers/7009005/70090240.pdf
14. Mor Peleg and Dov Dori, Representing Control Flow Constructs in Object-Process Diagrams. *Journal of Object-Oriented Programming*, 11, 3, pp. 58-71, 1998.
15. Liu Wenyin and Dov Dori, A Generic Integrated Line Detection Algorithm and its Object-Process Specification. *Computer Vision - Image Understanding (CVIU)*, 70, 3, pp. 420-437, 1998.
16. Liu Wenyin and Dov Dori, An Incremental Arc Segmentation Algorithm and its Evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 20, 4, pp. 424-431, 1998. http://www.computer.org/tpami/tp1998/i0424abs.htm
17. Mor Peleg and Dov Dori, Extending the Object-Process Methodology to Handle Real-Time Systems. *Journal of Object-Oriented Programming*, 11, 8, pp. 53-58, 1999.
18. Liu Wenyin and Dov Dori, Object-Process Diagrams as an Explicit Algorithm Specification Tool. *Journal of Object-Oriented Programming*, 12, 2, pp. 52-59, 1999. Abstract: http://www.joopmag.com/archive/9905/html/from_pages/index.shtml
19. Liu Wenyin and Dov Dori, Object-Process Based Graphics Recognition Class Library: Principles and Applications. *Software - Practice and Experience*, 29, 15, pp. 1355-1378, 1999.
20. Mor Peleg and Dov Dori, The Model Multiplicity Problem: Experimenting with Real-Time Specification Methods. *IEEE Transaction on Software Engineering*, 26, 8, pp. 742-759, 2000. http://iew3.technion.ac.il:8080/Home/Users/dori/Model_Multiplicity_Paper.pdf
21. Dov Dori, Document Analysis Systems Development and Representation through the Object-Process Methodology. Document Analysis Systems, S.W. Lee and Y. Nakano (Eds.) Lecture Notes in Computer Science, Vol. 1655, Springer, pp. 271-282 1999.
22. Dov Dori, Syntactic and Semantic Graphics Recognition: The Role of the Object-Process Methodology. In A. Chhabra and D. Dori (Eds.), International Workshop on Graphics Recognition, Lecture Notes in

Computer Science (1941), 2000.
http://link.springer.de/link/service/series/0558/papers/1941/19410277.pdf

23. Dov Dori, Object-Process Methodology Applied to Modeling Credit Card Transactions. *Journal of Database Management*, 12, 1, pp. 2-12, 2001. http://iew3.technion.ac.il:8080/Home/Users/dori/JDM-Dori-OPM.pdf
24. Pnina Soffer, Boaz Golany, Dov Dori and Yair Wand, Modeling Off-the-Shelf Information Systems Requirements: An Ontological Approach. *Requirements Engineering* 6, pp. 183-199, 2001. http://link.springer.de/link/service/journals/00766/papers/1006003/10060183.pdf
25. Hafedh Mili, Mohammed Fayad, Davide Burgali, David Hamu and Dov Dori, Enterprise Frameworks: Issues and Research Directions. *Software Practice and Experience*, 32, pp. 1-32, 2002.
26. Iris Reinhartz-Berger, Dov Dori, and Shmuel Katz, OPM/Web – Object-Process Methodology for Developing Web Applications. *Annals of Software Engineering* (to appear).

* A brief description of the tutorial, including: goals, detailed outline, justification of why the tutorial is important, background knowledge required and potential attendees.

Audio-visual/Technical Requirements

Data projector

Description of Supporting Material to be Distributed

Slide handouts and access to OPM-supporting CASE tools.
The book may be sold at a special reduced rate for the tutorial attendees (to be finalized with the publisher).

Tutorial speaker information: Dov Dori, Engineering Systems Division, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139-4307, dori@mit.edu tel. 781 858-2867 fax 617-258-7733.