
Houman Younessi
Director, Rensselaer Initiative in Software Engineering (RISE)
Associate Professor of Computer Science
Rensselaer Polytechnic Institute
Rensselaer at Hartford
Email: houman@rh.edu

Brian Henderson-Sellers
Director, Centre for Object Technology Applications and Research
Professor of Information Systems
Faculty of Information Technology
University of Technology, Sydney
Email: brian@it.uts.edu.au

Object technology is an approach that is increasingly being adopted for the development of quality software and software-intensive systems. Recent experience has demonstrated that it provides a sophisticated environment to support high quality software engineering practice. However, the use of object technology should not be restricted to languages but should encompass analysis, design, V&V and many such aspects belonging to the methodological dimension of the development life-cycle. Furthermore, it should also cover other issues belonging to the technological and contextual aspects.

It is now possible to support the full lifecycle by use of object-oriented (OO) development processes. Despite this, however, integrated process-focused OO approaches to software development, whilst vital to successful object-oriented project management, have not been a strong feature of extant approaches popularly practiced. Until recently, object-oriented development has been either ad-hoc, concentrating on language issues, or at best in accordance to a prescriptive methodology. OPEN [1] has been developed primarily to improve this situation.

In this presentation, the case for process is put forth. However, a prescriptive process can never suffice for the myriad of situations occurring in real software-intensive systems development. Three options are considered. Of these three, the idea of ensuring flexibility by creating a metamodel-based process framework is selected for further examination. One example of such an approach is OPEN (Object-oriented Process, Environment and Notation). The metamodel underpinning for OPEN is examined in detail and then it is shown how this metamodel is used for constructing a family of individually configured processes in order that each process meets the specific constraints of the organizational culture and/or the demands of the particular project.
A second objective of this presentation is to show the relationship between modeling; that is, the use of a notation language such as UML [2], and the essence of a process such as OPEN. We show that whilst it is possible to depict a model of a process such as OPEN using a language such as UML, it is in fact the modeling language that is circumscribed by the process. To achieve this end, and using OPEN as the background process, the UML (Unified Modeling Language) is introduced through a number of case study examples. Modeling and notational issues of the UML are discussed at an introductory level, set in the relevant context of OPEN's modeling tasks and techniques.

References.
