EOS
Andrea Omicini
2011/01/29 00:59
Table of Contents

Extensible Object Systems (EOS) ................................................................................................................................. 3
Partners ........................................................................................................................................................................... 3
Description ..................................................................................................................................................................... 3
Extensible Object Systems (EOS)

- **Funding Body:** MIUR (ex 40% - PRIN 2004)
- **Project Acronym:** EOS
- **Project Full Title:** "Extensible Object Systems"
- **Official Site:** http://bart.disi.unige.it/EOS/
- **Years:** 2004-2006

**Partners**

- Alma Mater Studiorum—DEIS
  - Mirko Virolì (UNIBO Coordinator)
  - Alessandro Ricci
  - Enrico Denti
  - Andrea Omicini
  - Antonio Natali
- Università degli Studi di Genova
  - Elena Zucca (Project Coordinator)
- Università degli Studi di Firenze
- Università degli Studi Piemonte Orientale

**Description**

The objective of this project is to advance design and foundations of modern extensible object systems, where by "extensible" we mean object systems which go beyond the traditional ingredients of the object paradigm, aiming at incorporating more and more powerful and flexible features. Note that this "extensibility" can be obtained at two complementary levels: The language level, by language constructs and/or more refined type systems. The metalevel, by more powerful operations for manipulating code fragments considered as black boxes. Both levels, often in an interdependent way, will be considered in this project. The research will consist in the development of formal models/calculi giving rigorous characterizations of the behaviour of the object systems under consideration, and in the application of these results to prevalent languages/environments such as Java, C++, C-sharp, .NET. We will develop prototypes and/or extensions to such languages/environments. The research plan will be based upon four complementary capabilities which are highly desirable for extensible object systems, orthogonal to the distinction between language level and metalevel in the sense that each one can work at either level: Compositional analysis Interoperability Adaptability Structuring and reconfiguration. The first is the capability of analyzing (e.g., typechecking) single fragments and then combining them on the basis of the results of the analysis, without any need of reinspection. The second is the capability of supporting coexistence and cooperation between fragments expressed in heterogeneous languages, or extensions/versions of a language. The third is the capability of adapting a system's behaviour to different, unexpected requests from the external contexts. The last is the availability of flexible mechanisms for composing fragments, which can take place either statically, before execution has started (what we call "structuring") or dynamically, when an application is already running (what we call "reconfiguration").