We demonstrate the main results of the Esprit Project P6333 IDEA, lasted from June 1992 to March 1997, whose main contributors are five industrial partners from five European countries: Bull, the main contractor (France), TXT-Informatica (Italy), Sema (Spain), Dutch PTT (Holland), and Bim (Belgium), and four research institutes: ECRC in Munich, INRIA in Paris, the University of Bonn, and Politecnico di Milano. The main objective of the IDEA Project is to investigate the use and promote the spread of object-oriented and rule-based technologies in next-generation information systems.

One of the major results of the IDEA Project is the IDEA Methodology and its supporting environment. The IDEA Methodology addresses the analysis, design, prototyping, and implementation of modern database systems applications, taking advantage of modern approaches developed in the context of database design, but also in the broader area of object-oriented software engineering. The distinguishing feature of the IDEA Methodology is the emphasis on both deductive rules and active rules, which significantly enrich the semantics supported within database applications.

The motivation behind the IDEA Methodology is the perception that technology is ahead of design methodologies for database applications. Many commercial systems exhibit advanced features (such as objects and rules) which remain largely unused due to a lack of experience, culture, and design tools. The IDEA Methodology, backed by a rich collection of tools, aims at closing this gap, by focusing on the new features of modern database technology.

The distinguishing approach of the IDEA Methodology is the emphasis on knowledge independence; by this term, we indicate the ability of extracting semantic knowledge from applications, normally encoded in a procedural format (programs), and placing it into the database schema, encoded declaratively in the form of objects and rules; in this way, knowledge is system-enforced, shared by all applications, and may be maintained and evolve more easily.

Knowledge independence takes leverage from Chimera, the conceptual specification language of the IDEA Methodology, which is based on an object oriented data model and offers active and deductive functionalities. The language draws its name from the Chimera, an imaginary creature of Greek mythology with a lion’s head, a goat’s body, and a serpent’s tail, which well represents the integration of three different paradigms, object-orientation, active rules and deductive rules, into a single powerful body.

The IDEA Methodology is surrounded by a number of tools for supporting the developer throughout all the phases of the application life-cycle: schema design, active rule generation, rule analysis, application prototyping, debugging, and browsing. Support is also given in mapping active rule applications from the IDEA design language Chimera into Oracle.

The IDEA Tool Environment consists of five tools: Iade, Arachne, Argonaut, Pandora, and the Algres Testbed. The tools are clustered in two groups. The "Design Support Environment" assists analysis, design, and implementation and includes Iade, Arachne, Argonaut, and Pandora. The "Prototyping Environment" assists application rapid prototyping and includes the Algres Testbed, a sophisticated execution environment consisting of six components (the Schema Compiler, the Trigger and Operation Compiler, the Transaction Compiler, the Debugger, the Form Manager, and the Browser).

At ICDE we demonstrate a Java implementation of the tools (the IDEA Web Lab) and the Web site of the IDEA Methodology.