Answer Set Programming - a story of default negation, definitions and informal semantics

Abstract: After about two decades in existence, Answer Set Programming (ASP) has earned a prominent position as one of the most commonly used knowledge representation languages owing its appeal to intuitive, flexible and expressive syntax, a sound programming methodology, and an impressive effectiveness of software for processing programs. The source of its success lies in years of research in knowledge representation, logic programming, satisfiability, and constraint satisfaction. ASP has been able to build on accomplishments in each of these areas, reaching an elegant balance between expressivity, ease of use, and computational effectiveness. In this talk I will examine the main research themes that first led to formulation of ASP as a computational paradigm for declarative problem modeling and solving, and that still continue to this day to influence how ASP evolves. They concern the default negation and its close relative, negation as failure, with their connections to commonsense reasoning research, logic programming, and nonmonotonic logics of early 1980s; definitions, how they shaped some early research in logic programming and how they turned critical for the way programs in ASP are now commonly developed; and finally the all important matter of informal semantics with its power to validate formal concepts and offer guidance on how to represent formally atomic pieces of knowledge and how to combine them together.