The talk is about a 2-part introduction to research on generating query plans from proofs. The setting is answering queries over sources that are connected to one another by semantic relationships (that is, integrity constraints). The sources may have overlap, may have multiple means of access with different requirements (e.g., number of required arguments), and the means of access within and across relations may differ radically in cost. This work presents a method for generating the best query plan, exploiting the integrity constraints. A solution to this problem can be applied in many database contexts such as answering queries using materialized views, optimizing queries using constraints, and querying on top of web services or web forms. The approach factors the problem into 3 components including 1) proving that the query can be answered, 2) generating a plan from the proof, 3) estimating the cost of a plan.

In the first talk, I will give background on the applications, some examples of generating plans from proofs, and start in on the underlying theory behind the proof/plan correspondence for different integrity constraint languages, neglecting the issue of the efficiency of plan generation and the cost of executing the generated plans. In the second talk, I will finish the theory and then discuss a system that can generate plans efficiently, taking cost into account.

About the Speaker:

Michael Benedikt is a Professor at Oxford University's computer science department, and a fellow of University College Oxford. He came to Oxford after a decade in US industrial research laboratories, including a position as Distinguished Member of Technical Staff at Bell Laboratories. He has worked extensively in mathematical logic, finite model theory, verification, database theory, and database systems, and has served as chair of the ACM's main database theory conference, Principles of Database Systems. The current focus of his research is Web data management, with recent projects including querying of the deep Web, querying and integration of annotated data, and querying of web services.