Advanced Databases

Jerome Simeon
IBM T.J. Watson Research Center
(js1491@nyu.edu)
(simeon@us.ibm.com)

Spring 2014

1 Goals

In this course, we will study two critical areas of modern database technology: advanced aspects of query compilation, and data integration techniques.

2 Prerequisites

This class assumes the students have followed an undergraduate-level database system class, and a general programming class. It is recommended that students have taken a class on algorithms and data structures. Additional background in compilation is useful but is not required.

3 Overview

Query compilers are central to database systems, and modern databases go well beyond relational techniques usually taught in introductory database courses. Over the year, query compilers have grown to address the needs of new applications as well as adapt to new hardware. The purpose of the query compiler is to transform user data requests, often described in a declarative language such as SQL, into an efficient execution plan. We will study an array of advanced techniques, including some of the following: modern indices and data structures for storage, extensions to the relational algebra for nested queries, query compilation techniques for Web format such as XML or JSON, parallelization, handling of updates and transactions.

Data Integration is central to modern applications which often must process information stored in diverse data sources, as companies cannot rely on single databases to address their business needs. In the last twenty years, databases have developed techniques to address the needs of those applications. Due to the wide variety and heterogeneity of data sources usually involved, data integration raises an broad range of challenges. We will study modern techniques

and architectures for data integration that have been developed over the last twenty years by the database community, including some of the following: architectures for data integration, how to handle semantic heterogeneity, how to process queries over several data sources, how to handle discrepencies or uncertainty in the data, how to maintain information in an integrated database.

Query compilers and information integration are two very active areas of research, and they fundamentally rejoin in several areas: flexible data models such as RDF, XML or JSON which are used to handle heterogeneity in the data, distributed compilation techniques that are necessary to execute queries over multiple data stores. We will study those overlapping aspects toward the end of the course.

4 Class Material

"Database Management Systems", by Ramakrishnan and Gehrke, McGraw-Hill, 2002. This book is for background, but we will go beyond its content.

"Query Compilers" by Guido Moerkoette at: http://pi3.informatik.uni-mannheim.de/ moer/querycompiler.
A bible of query compilation techniques, notably covering techniques that go beyond relational stores.

"Principles of Data Integration" by Anhai Doan, Alon Halevy, Zachary Ives. Morgan Kaufman, 2012. A recent book on data integration that covers most technical aspects important to this area.

Note that none of those are required. They are recommended as they provide additional context and details, but students should be fine with the lectures slides.

Additional reading material relevant to each class will be provided over time, usually from the academic litterature, for further study.

5 Syllabus

This is a tentative schedule, and will be subject to revisions as the class progresses. The class will alternate between the two main topics (query compilation and data integration), usually two classes at a time.

- Relational query compilers. The first two classes will cover classic relational compilation, with a focus on indices and query optimization, and will serve as a refresher and stepping stone for the rest of the class. (2 weeks)
- Introduction to Data integration. We will cover the main principles underlying modern data integration systems, including general architecture aspects. (2 weeks)
- Modern storage and indices. We will review modern indices techniques, e.g., collumn stores, spatial indices. (2 weeks).

- Techniques for schema matching, wrapper creation and maintenance. Techniques for approximate information processing. (2 weeks)
- Query compilation and optimization techniques for nested queries. Indepth of nested relational algebras and query decorrelation (unnesting) techniques. Query compilation and optimization techniques for semistructured and unstructured data (XML or JSON for instance). Path analysis, tree-join algorithms etc. (2 weeks)
- Semantic Web for Information Integration. We will cover RDF and related technologies such as linked data and show how such techniques can be used for information integration. (2 weeks)
- Techniques for compilation and optimization in information integration context. Chase and backchase techniques. Query delegation to sources in an heterogeneous environment. (2 week)